NAAC GRADE A+
ACCREDITED UNIVERSITY

REVA UNIVERSITY
Bengaluru, India

HANDBOOK

B. Tech.

Artificial Intelligence and Data Science

Academic Year 2024-25

RUKMINI EDUCATIONAL
Charitable Trust

www.reva.edu.in

Chancellor's Message

"Education is the most powerful weapon which you can use to change the world."

- Nelson Mandela.

At REVA University, we are firm believers in this truth.
The power of education lies not only in the acquisition of knowledge but in its ability to transform lives, communities, and nations. As the world rapidly evolves, driven by technological advancements and global interconnectedness, education remains the catalyst for progress, innovation, and positive change.

Today, we live in an era where knowledge is no longer confined to books or classrooms. Information is accessible at the touch of a button, and opportunities for learning abound in every corner of the world. Yet, the true essence of education lies beyond the mere accumulation of information. It is about cultivating critical thinking, fostering creativity, and empowering individuals to contribute meaningfully to society. At REVA University, we are committed to nurturing seekers of knowledge who aspire to make a difference in the world.

Guided by our founding philosophy of "Knowledge is Power," we strive to create an environment where intellectual curiosity is encouraged, and dreams are transformed into reality. India is a land of immense talent and potential, and it is our duty as educators to provide the spark that ignites this potential. Through the transformative power of education, we aim to shape future leaders who possess not only technical proficiency but also strong ethical values and a commitment to social responsibility.

A university is more than just a place of learning; it is a place of growth, exploration, and transformation. Our faculty, with their expertise and dedication, are at the heart of this transformation. They are more than teachers; they are mentors who guide students on their journey of self-discovery and academic excellence. Our student-centric, transformational approach ensures that every learner is given the opportunity to explore their full potential and exceed their own expectations.

At REVA University, we take great pride in our state-of-the-art infrastructure and facilities, designed to provide an inspiring and conducive environment for both academic and extracurricular pursuits. Our campus is a vibrant space where students are encouraged to challenge their minds, develop their skills, and grow as individuals.

As we move forward, I am reminded of the words of Benjamin Disraeli: "A university should be a place of light, of liberty, and of learning." This vision continues to inspire us at REVA University, where we work as a team to create a brighter future for our students and our society.

I invite you to join us on this journey of enlightenment, growth, and transformation. Together, let us lay the foundation for a future built on values, wisdom, and knowledge.

Dr. P. Shyama Raju
Chancellor
REVA
University

Pro Chancellor's Message

REVA University has emerged as a premier destination for higher education across diverse fields such as engineering, science, commerce, management, architecture, law, arts, and humanities. Our commitment to excellence in education is reinforced by the adoption of cutting-edge technologies and innovative teaching methods that ensure our students are equipped for the future.

The integration of modern tools and ICT-based technologies is at the core of our academic philosophy. We focus on digital learning, project-based learning, and personalized learning experiences that cater to individual student needs. By harnessing the power of advanced technologies such as AI-powered learning platforms, data analytics, and virtual/augmented reality, we are able to offer dynamic, interactive educational experiences that transcend traditional classroom boundaries. This technological transformation enables us to deliver STEM education more effectively while providing our faculty with ongoing professional development to stay at the forefront of teaching innovations.

Our programs are meticulously designed after a thorough analysis of current industry needs and trends, with a focus on knowledge assimilation, practical application, and global employability. We recognize the importance of preparing students not just for today's job market but for a rapidly changing future where automation, artificial intelligence, and data-driven decision-making will play pivotal roles. To meet these demands, we emphasize hands-on learning, skill development, and innovation, ensuring that our students are well-prepared to thrive in their respective fields.

At REVA University, we have implemented the Choice Based Credit System and Continuous Assessment Grading Pattern (CBCS – CAGP) in all our programs. This system provides students with the flexibility to choose subjects that align with their interests while developing essential skills. CBCS courses integrate knowledge on local, regional, national, and global issues, fostering a comprehensive understanding of the world and enabling students to become entrepreneurial and employable in a competitive global marketplace. Furthermore, students are offered a variety of value-added courses to enhance their skillsets, ensuring they are equipped to navigate the evolving demands of their chosen careers.

The future of the engineering profession, and indeed many other fields, will be shaped by dramatic technological and societal changes. The rise of automation, sustainability concerns, and increased globalization will create both opportunities and challenges for the next generation of professionals. REVA University is fully prepared to meet these challenges head- on. Through state-of-the-art laboratories, research centers, and partnerships with premier industries and academic institutions, we are committed to creating talented professionals and leaders who can navigate these future transformations.

Our growth and success have been built on a foundation of excellence in curriculum design, student-centric teaching methods, and hands-on learning practices. I extend my gratitude to our students, parents, faculty, staff, and well-wishers for their contributions in helping REVA University become a next-generation, globally recognized education hub.

Mr. Umesh S. Raju
Pro Chancellor
REVA University

Vice Chancellor's Message

The last two decades have seen remarkable growth in higher education in India and across the globe. The move towards interdisciplinary studies and interactive learning has opened up several options as well as created multiple challenges.

A strong believer and practitioner of the dictum "Knowledge is Power", REVA University has been on the path of delivering quality education by developing young human resources on the foundation of ethical and moral values while boosting their leadership qualities, research culture and innovative skills.

Built over 50 acres of green campus, this 'temple of learning' has excellent and state-of-the-art infrastructure facilities conducive to a higher teaching-learning environment and research. The main objective of the University is to provide higher education of global standards and hence, all the programmes are designed to meet international standards. Highly experienced and qualified faculty members, continuously engaged in the maintenance and enhancement of a student-centric learning environment through innovative pedagogy, form the backbone of the University. All the programmes offered by REVA University follow the Choice Based Credit System (CBCS) with Outcome Based Approach.

The flexibility in the curriculum has been designed with industry-specific goals in mind and the educator enjoys complete freedom to appropriate the syllabus by incorporating the latest knowledge and stimulating the creative minds of the students. Benchmarked with the course of studies of various institutions of repute, our curriculum is extremely contemporary and is a culmination of efforts of great think tanks - a large number of faculty members, experts from industries and research-level organizations.

At REVA University, research, consultancy and innovation are regarded as our pillars of success. Most of the faculty members of the University are involved in research by attracting funded projects from various research-level organizations like STI HUB, ISRO, DST, VGST, DBT, DRDO, AICTE and industries. The outcome of the research is passed on to students through live projects from industries. The entrepreneurial zeal of the students is encouraged and nurtured through EDPs and EACs. With firm faith in the saying, "Intelligence plus character –that is the goal of education" (Martin Luther King, Jr.), I strongly believe REVA University is marching ahead in the right direction, providing a holistic education to the future generation and playing a positive role in nation building. We reiterate our commitment to providing premium quality education accessible to all and an environment for the growth of overall personality development leading to generating "GLOBAL PROFESSIONALS".

Dr. N. Ramesh

Vice Chancellor (I/C), REVA University

Director's message

Congratulations and welcome all new students to the prestigious school of Computer Science and Engineering. The school has a rich blend of experienced and energetic faculty who are well-qualified in various aspects of Electronics, and Computer Science. The school possesses numerous state-of-the- art digital classrooms and laboratories having contemporary computing equipment, including cloud- based systems. The school offers B.Tech. in Computer Science and Engineering, B.Tech in Artificial Intelligence and Data Science, B. Tech in Computer Science and Engineering (Internet of Things and Cyber Security including Block Chain Technology), M.Tech in Computer Science and Engineering (Full Time) and M.Tech in Computer Science and Engineering (Part-Time). In addition, the school has a research center in which a student can conduct cutting-edge research leading to a Ph.D. degree. The faculties pursueresearch in areas like Data Mining, Healthcare Systems, Blockchain, Wireless Networks, Cloud Computing, Image Processing, Software Architecture, and Machine Learning Applications.

Curricula of both undergraduate and postgraduate programmes have been designed through a collaboration of alumni, academic, research, and industry experts to bridge the gap between industry and academia and inculcate innovation and leadership qualities. This makes the programme highly practical and industry-oriented. The B.Techprogramme aims to create quality human resources to play leading roles in the contemporary, competitive industrial and corporate world. The Master's Degree focus on quality research and design in the core and applicationareas to foster a sustainable world and enhance the global quality of life by adopting enhanced design techniques and applications. This thought is reflected in the various courses offered in the masters' programmes. Research degree programmes aim to design and develop solutions to contemporary computer and engineering technologies oriented towards human development.

Welcome to the school of Computer Science and Engineering at REVA University for better learning and becoming future leaders for the nation's socio-economic growth and the world.

Dr. Ashwinkumar U Motagi

Director, School of CSE

Rukmini Educational Charitable Trust

In October 2005, the long-standing dream of Smt. Rukmini Shyama Raju to educate the youth became a reality with the establishment of Rukmini Educational Charitable Trust (RECT). The RECT is a Charitable Trust to promote, establish, and conduct developmental activities in the fields of Arts, Commerce, Engineering, Environmental Science, Management, Science & Technology, Law, Architecture, Performing Arts, and others.

To meet this aim, the Trust has set up the REVA Group of Educational Institutions comprising REVA Institute of Technology and Management, REVA Institute of Science and Management, REVA Institute of Management Studies, REVA Institute of Education, REVA First Grade College, REVA Independent PU College Kattigenahalli, REVA Independent PU College-Ganganagar, and REVA Independent PU College Sanjaynagar, and REVA University. Through these institutions, the Trust seeks to fulfil its vision of providing world-class education and creating abundant opportunities for the youth to excel in academics.

Every great human enterprise is powered by the vision of extraordinary individuals and is sustained by the people who derive their motivation from the founders. Dr. P. Shyama Raju, the Chairman of the Trust, is a developer and builder of repute, a captain of the industry in his own right, and the Chairman & Managing Director of the DivyaSree Group of companies. The other trustees are Shri Bhaskar N Raju, and Smt Arathi B. Raju, Shri. Umesh. S. Raju and Smt. Tanisha U. Raju.

The Rukmini Educational Charitable Trust is driven by the purpose of nurturing students who are in pursuit of quality education and excellence. REVA, today, is a family of 10 institutions providing education from pre-university to post-graduation and providing research and resource support to PhD degree aspirants.

About REVA University

REVA University is a State Private University established in Karnataka State under the Government of Karnataka Act No. 13 in the year 2012 in Bengaluru, the IT capital of India. REVA University, recognised by the University Grants Commission (UGC) and approved by the All India Council for Technical Education (AICTE), has an A+ grade from NAAC. The University has a sprawling green campus spread over 43 acres of land.

The University has a DIAMOND Band ranking from QS I Gauge. As per QS Asian University Rankings, it is ranked 47th among all the private Universities of India and 6th among all private universities of Karnataka. In less than a decade, REVA University, Bengaluru, has established itself as a Global University in education by earning recognition as a forward-thinking institution across all disciplines.

The University currently offers 38 full-time undergraduate Programmes, 33 full-time postgraduate programmes, 20 Ph.D. programmes, and certification and diploma programmes. The University offers programmes under the faculty of Engineering, Architecture, Science and Technology, Commerce, Management Studies, Law, Arts & Humanities, and Performing Arts & Indic Studies. REVA offers some of the trending fields of study as undergraduate courses in Sports Science (B. Sc.), Agricultural Engineering (B. Tech), and Aerospace Engineering (B. Tech) which are full-time application-based programmes with a unique blend of theory and practical components.

With state-of-the-art infrastructure, the University has created a vibrant academic environment conducive to higher learning and research. This includes 200 smart classrooms that support blended learning, real-industry-like labs that foster on-the-job learning in students, a tech-enabled library with over 1 lakh collections of books, and most importantly, modern pedagogy. REVA currently has numerous students on campus from around the country. The campus has exclusive Halls of Residence which provides comfortable accommodation for boys and girls, apart from catering to the needs of all cuisine and ensuring adequate amenities are provided to make their stay an extended home.

In its mission to become a social impact university, REVA University has initiated several Corporate Social Responsibility initiatives. Jagruti, Abhivriddhi, Vanamahotsava, Education on Wheels, and Pragna are a few of the several projects in REVA. REVA has now moved on to become a Social Impact University and has aligned with the United Nations Sustainable Development Goals. Through these initiatives, REVA aspires to become an innovative University by developing a social connection with leadership qualities, ethical and moral values, research culture, and innovative skills through higher education of

global standards.

Vision

- REVA University aspires to become an innovative university by developing excellent human resources with leadership qualities, ethical and moral values, research culture and innovative skills through higher education of global standard.

Mission

- To create excellent infrastructure facilities and state-of-the-art laboratories and incubation centers
- To provide student-centric learning environment through innovative pedagogy and education reforms
- To encourage research and entrepreneurship through collaborations and extension activities
- To promote industry-institute partnerships and share knowledge for innovation and development
- To organize society development programs for knowledge enhancement in thrust areas
- To enhance leadership qualities among the youth and enrich personality traits, promote patriotism and moral values.

Objectives

- Creation, preservation and dissemination of knowledge and attainment of excellence in different disciplines
- Smooth transition from teacher - centric focus to learner - centric processes and activities
- Performing all the functions of interest to its major constituents like faculty, staff, students and the society to reach leadership position
- Developing a sense of ethics in the University and Community, making it conscious of its obligations to the society and the nation
- Accepting the challenges of globalization to offer high quality education and other services in a competitive manner

REVA UNIVERSITY
Bengaluru, India

ACADEMIC REGULATIONS

B. Tech., Degree Programs
(Applicable for the programs offered from 2024-25)

(Framed as per the provisions under Section 35 (ii), Section 7 (x) and Section 8 (xvi) & (xxi) of the REVA University Act, 2012)

THESE ACADEMIC REGULATIONS ARE UNDER CHOICE BASED CREDIT SYSTEM AND CONTINUOUS ASSESSMENT GRADING PATTERN (CBCS-CAGP)

1. Title and Commencement:

1.1 These Regulations shall be called **"REVA University Academic Regulations – B. Tech., Degree Program for the batch of students admitted for AY 2024-25 subject to amendments from time to time by the Academic Council on recommendation of respective Board of Studies and approval of Board of Management"**

1.2 These Regulations shall come into force from the date of assent of the Chancellor.

2. The Programs:

These regulations cover the following B. Tech., Degree programs of REVA University offered for the admitted batch during AY 2024-25 under respective schools.

SL No.	Name of the School	Name of the Program
1	School of Civil Engineering	B Tech in Civil Engineering
2	School of Computing and Information Technology	B Tech in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
		B Tech in Computer Science and Information Technology
		B Tech in Information Science and Engineering
		B Tech in Computer Science and Systems Engineering
3	School of Computer Science and Engineering	B Tech in Computer Science and Engineering
		B Tech in Artificial Intelligence and Data Science
		B Tech in Computer Science and Engineering (Internet of Things and Cyber Security including Blockchain Technology)
4	School of Electrical and Electronics Engineering	B Tech in Electrical and Electronics Engineering
5	School of Electronics and Communication Engineering	B Tech in Electronics and Communication Engineering
		B Tech in Electronics and Computer Engineering
		B Tech in Robotics and Artificial Intelligence
6	School of Mechanical Engineering	B Tech in Mechanical Engineering
		B Tech in Mechatronics Engineering
		B.Tech in Aerospace Engineering
7	Department of Agricultural Engg.	B Tech in Agricultural Engineering

3. Duration and Medium of Instructions:

3.1 Duration: The duration of the B Tech degree program shall be **FOUR** years comprising of **EIGHT** Semesters. A candidate can avail a maximum of 16 semesters - 8 years as per double duration norm, in one stretch to complete B. Tech degree, including blank semesters, if any. Whenever a candidate opts for blank semester, he/she has to study the prevailing courses offered by the School when he/she resumes his/her studies.

3.2 The medium of instruction shall be English.

4. Definitions:

4.1 Course: "Course" means a subject, either theory or practical or both and project, listed under a program; Example: "Fluid Mechanics" in B. Tech Civil Engineering program, "Signals and Systems" in B. Tech., Electronics and Communication Engg are the examples of courses to be studied under respective programs.

Every course offered will have three components associated with the teaching-learning process of the course, namely, L, T and P, where,

L stands for **Lecture** session consisting of classroom instruction.

T stands for **Tutorial** session, an Interactive session including discussions, self-study, desk work, seminar presentations, and other methods to deepen understanding.

P stands for **Practice** session and it consists of hands-on experience such as laboratory experiments, field studies, case studies, project based learning or course end projects and self- study courses that equip students to acquire the required skill component.

4.2 Classification of Courses

Courses offered are classified as follows:

4.2.1 Foundation Course (FC): The foundation Course is basic course which should be completed successfully as a part of graduate degree program irrespective of the program of study.

4.2.2 Professional Core Course (also known as Hard Core(HC) Course): The **Professional Core** is a core course in the main branch of study and related branch(es) of study, if any, that the candidates have to complete compulsorily.

4.2.3 Professional Elective Course (also known as Soft Core (SC) Course): Professional Elective course is a choice or an option for the candidate to choose a course from a pool of courses from the main branch of study or from a sister/related branch of study which supports the main branch of study.

4.2.4 Open Elective Course (OE): An elective course chosen generally from other discipline / subject, with an intention to seek exposure to the basics of subjects other than the main discipline the student is studying is called an **Open Elective Course.**

4.2.5 Audit Course (also known as Non-Credit Course /Mandatory Course(MC)): These courses are mandatory for students joining B.Tech. Program and students have to successfully complete these courses before the completion of degree.

4.2.6 Project Work / Dissertation: Project work / Dissertation work is a special course involving application of knowledge in solving / analyzing /exploring a real life situation
/ difficult problems to solve a multivariable or complex engineering problems. The project will be conducted in two phases, phase-I, consists of literature survey, problem identification, formulation and methodology. In Phase-II, student should complete the project work by designing or creating an innovative process or development of product as an outcome. A project work is carried out as minor project in 3rd year and major project in 4th year with appropriate credits allocated.

4.2.7 Skill Development Course (SDC): It is a practice based course introduced in first year, second year and third year that lead to advanced job skills as per current industry/societal requirements to enhance high employability index of graduates. It may also lead to a certificate, diploma and advanced diploma, etc.

4.2.8 Emerging Technology Course (ETC): This course is introduced in the first year, focusing on the latest advancements and innovations in technology. It aims to equip students with cutting-edge knowledge and skills relevant to emerging fields.

4.3 "Program" means the academic program leading to a Degree, Post Graduate Degree, Post Graduate Diploma Degree or such other degrees instituted and introduced in REVA University.

5. Eligibility for Admission:

5.1. The eligibility criteria for admission to B Tech Program of 4 years (8 Semesters) is given below:

Sl. No.	Program	Duration	Eligibility
1	Bachelor of Technology (B. Tech)	4 Years (8 Semesters)	A. FOR GENERAL MERIT CANDIDATES: A candidate who has Passed in 2nd PUC / 12 Std / Equivalent Examination with English as one of the Language and obtained a Minimum of 45% of Marks in aggregate in Physics and Mathematics as compulsory subjects along with Chemistry / Bio Technology / Biology / Computer Science / Electronics as optional subjects in the qualifying examination is eligible to pursue in under graduate programs (BE). B. FOR SC/ST & OBC (Cat-I, 2A, 2B, 3A 3B) CATEGORY CANDIDATES: A candidate who has Passed in 2nd PUC / 12 Std / Equivalent Examination with English as one of

			the Language and obtained a Minimum of 40% of Marks in aggregate in Physics and Mathematics as compulsory

Sl. No.	Program	Duration	Eligibility
			subjects along with Chemistry / Bio Technology / Biology / Computer Science / Electronics as optional subjects in the qualifying examination is eligible to pursue in under graduate pogroms (BE). C. The marks obtained by the candidate in Biotechnology/Biology/Computer Science / Electronics in the qualifying examination will be considered in the place of Chemistry in case the marks obtained in Chemistry is less for the required aggregate percentage for the pursue of determination of eligibility.
2	Bachelor of Technology (B Tech-Lateral Entry program)	3 Years (6 Semesters)	A. candidate who has passed Qualifying Examination i.e. any diploma examination or equivalent examination and obtained an aggregate minimum of 45 % (for General Merit Candidates) of marks taken together in all the subjects of the final year (i.e.,fifth and six semesters) diploma examination (Q E) is eligible for admission to B E- Courses and 40% of marks in Q. E. in case of SC, ST and Backward Classes of Karnataka Candidates. Passed B.Sc Degree from a recognized University as defined by UGC, with at least 45% marks (40% in case of candidates belonging to SC/ST category) and passed XII standard with mathematics as a subject. B. Provided that in case of students belonging to B.Sc. Stream, shall clear the subjects of Engineering Graphics / Engineering Drawing and Engineering Mechanics of the first year Engineering program along with the second year subjects. C. Provided further that, the students belonging to B. Sc. Stream shall be considered only after filling the seats in this category with students belonging to the Diploma stream. D. Provided further that student, who have passed Diploma in Engineering & Technology from an AICTE approved Institution or B. Sc., Degree from a recognized University as defined by UGC, shall also be eligible for admission to the first year Engineering Degree courses subject to vacancies in the first year class in case the vacancies at lateral entry are exhausted. However the admissions shall be based strictly on

			the eligibility criteria as mentioned in A, B, D, and E above.

Sl. No.	Program	Duration	Eligibility
			E. Passed D.Voc. Stream in the same or allied sector. (The Universities will offer suitable bridge courses such as Mathematics, Physics, Engineering drawing, etc., for the students coming from diverse backgrounds to achieve desired learning outcomes of the program)
3	Bachelor of Technology (B Tech)		Any candidate with genuine reason from any University / Institution in the country upon credit transfer could be considered for lateral admission to the respective semester in the concerned branch of study, provided he/she fulfils the University requirements.

5.2 Provided further that the eligibility criteria are subject to revision by the Government Statutory Bodies, such as AICTE, UGC from time to time.

6. Courses of Study and Credits

6.1 Each course of study is assigned with certain credit value

6.2 Each semester is for a total duration of 20 weeks out of which 16 weeks dedicated for teaching and learning, CIA and the remaining 4 weeks for SEE, evaluation and result declaration.

6.3 The credit hours defined as below:

In terms of credits, L refers to lecture hour (theory) credit per week, that indicate every one hour lecture per week of L amounts to 1 credit per Semester; T and P refer to tutorial hours and practice hours credit per week, that indicate every two hours of T and P per week amounts to 1 credit per Semester or a three hour session of T / P amounts to 2 credits per semester.

The total duration of a semester is 20 weeks inclusive of semester-end examination.

The following table describes credit pattern

Table -2: Credit Pattern

Lectures (L)	Tutorials (T)	Practice (P)	Credits (L:T:P)	Total Credits	Total Contact Hours
4	2	0	4:1:0	5	6
3	2	0	3:1:0	4	5
3	0	2	3:0:1	4	5
2	2	2	2:1:1	4	6
0	0	6	0:0:3	3	6
4	0	0	4:0:0	4	4
2	0	0	2:0:0	2	2
0	0	2	0:0:1	1	2

a. The concerned BoS will choose the convenient Credit Pattern for every course based on size and nature of the course.

7. Different Courses of Study:

Different **Courses of Study** are labeled as follows:

a. Foundation Course (FC)

b. Professional Core Course (Hard Core(HC))

c. Professional Elective Course (Soft Core(SC))

d. Open Elective Course (OE)

e. Skill Development Course (SDC)

f. Audit Course (Non-credit Course/ Mandatory Course) (MC)

g. Project Work / Dissertation: A project work is carried out as minor project in 3rd year and major project in 4th year with appropriate credits allocated. These are defined under Section 4.2.6 of this regulation.

h. Emerging Technology Course(ETC)

8. Credits and Credit Distribution

A candidate must earn 160 credits for successful completion of B Tech degree with the distribution of credits for different courses with the credit distribution given in the scheme of study.

8.1 The concerned BOS based on the credits distribution shall prescribe the credits to various types of courses listed in section 4.2 and shall assign title to every course thereon.

8.2 Every course including project work, practical work, field work, self-study elective should be entitled as per the list declared in section 4.2. However, as per AICTE, the credit distribution for various category of courses is given below in the table.

Sl. No.	Course Category	Abbreviation (AICTE)	Suggested breakup of credits (AICTE)	Credit breakup (REVA)
1	Humanities and Social Sciences including Management courses (HSMC)	HSMC	12	9
2	Basic Science Courses	BSC	25	19
3	Engineering Science courses including workshop, drawing, basics of electrical/mechanical/computer etc	ESC	24	23
4	Program Core Courses	PCC	48	61
5	Program Elective courses relevant to chosen specialization/branch	PEC	18	18
6	Open subjects – Electives from other technical and /or emerging subjects	OE	18	6
7	Project work, seminar and internship in industry or elsewhere	PROJ	15	18
8	Audit Courses (Mandatory Course)	MC	-	-
9	Skill Development Courses (SDC)/ETC/AEC	ETC/AEC/SDC		6
			160	160

8.3 The concerned BOS shall specify the desired Program Educational Objectives, Program Outcomes, Program Specific Outcomes and Course Outcomes while preparing the curriculum of a particular program. A candidate can enroll for a maximum of 26 credits

and a minimum of 16 credits per Semester. However, he / she may not successfully earn a maximum of 26 credits per semester. This maximum of 26 credits does not include the credits of courses carried forward by a candidate.

8.4 Only such full-time candidates who register for a minimum prescribed number of credits in each semester from I semester to VIII semester and complete successfully 160 credits in 8 successive semesters shall be considered for declaration of Ranks, Medals, Prizes and are eligible to apply for Student Fellowship, Scholarship, Free ships, and such other rewards / advantages which could be applicable for all full time students.

8.5 Minor degree/ Honor Degree:

To acquire Add on Proficiency Diploma/ Minor degree/ Honor Degree: a candidate can opt to complete a minimum of 18 extra credits either in the same discipline /subject or in different discipline / subject in excess to 160 credits for the B Tech Degree program. The Add on Proficiency Certification / Diploma/ Minor degree/ Honor Degree: so issued to the candidate contains the courses studied and grades earned.

9 Assessment and Evaluation

9.1 The Scheme of Assessment will have two parts, namely;

i. Continuous Internal Assessment (CIA); and
ii. Semester End Examination (SEE)

9.2 Assessment and Evaluation of each Course shall be for 100 marks. The CIA and SEE of UG Engineering programs shall carry 50:50 marks respectively (i.e., 50 marks CIA; 50 marks for SEE).

9.3 The 50 marks of CIA shall comprise of:

Internal Assessment Test (IA Test)	40 marks
Assignments / Seminars / Model Making / Integrated Lab / Project Based Learning / Quizzes, etc.	10 marks

9.4 There shall be **two Internal Assessment Tests** (IA tests) are conducted as per the schedule announced below. **The Students shall attend both the Tests compulsorily.**

- 1st test is conducted for 20 marks during **8th week** of the Semester;
- 2nd test is conducted for 20 marks during **15th week** of the of the Semester;

9.5 The coverage of syllabus for the said tests shall be as under:

- The question paper of the **1st test should be based on first 50% of the total syllabus**.
- The question paper of the **2nd test should be based on remaining 50 % of**

the total syllabus.

- An assignments must be designed to cover the entire syllabus.

9.6 There shall be two Assignments / Project Based Learning / Field Visit / Quiz test carrying 10 marks covering the entire syllabus.

9.7 SEE for 50 marks practical exam shall be held during 16th and 17th week of the semester.

9.8 SEE for 50 marks theory exam shall be held during 18th 19th and 20th week of the semester and it should cover entire syllabus.

9.9 IA test paper is set for a maximum of 40 marks to be answered in 1.5 hours duration (for 1 credit course, exam is conducted for 25 marks with a duration of 1 hour). A test paper can have 5 main questions. Each main question is set for 10 marks. The main question can have 2-3 sub questions all totaling 10 marks. Students are required to answer any 4 main questions. Each question is set using Bloom's action verbs. The questions must be set to assess the course outcomes described in the course document with the choice is given in questions.

9.10 The question papers for IAs shall be set by the internal faculty who have taught the course. If the course is taught by more than one faculty all the them together shall devise a common question paper(s). However, these question papers shall be scrutinized by the Question Paper Scrutiny Committee (internal BoE members) to bring the quality and uniformity in the question paper.

9.11 The evaluation of the answer scripts shall be done by the internal faculty who have taught the course and set the test paper. After evaluation of answer booklets faculty must distribute to students. If any corrections are noticed by students, faculty have to justify the award of marks. The final marks are to be declared only after such corrections. school/Department has to take signature for the marks obtained.

9.12 Assignment/seminar/Project based learning/simulation based problem solving/field work should be set in such a way, students be able to apply the concepts learnt to a real life situation and students should be able to do some amount of self-study and creative thinking. While setting assignment care should be taken such that the students will not be able to plagiarize the answer any other resources. Course faculty at his/her discretion can design the questions as a small group exercise or individual exercise. This should encourage collaborative learning and team learning and self-study.

9.13 CIA marks must be decided well before the commencement of SEE.

9.14 The SEE theory question paper is designed to comprehensively evaluate student's understanding and mastery of the course content. A total of 8 main questions will be crafted to cover the entire syllabus, out of which students are required to answer any 5 questions in full. Each main question will carry a maximum of 20 marks and may include 3 to 4 sub-questions. The maximum marks for each course is 100, and the duration is 3 hours. All questions must be formulated using Bloom's action verbs to ensure alignment with cognitive learning objectives. Additionally, the questions must be meticulously crafted to assess the student outcomes detailed in the course document, ensuring thorough coverage of all the course outcomes.

and earn the required credits.

9.15 There shall be minimum three sets of question papers for the SEE, of which one set along with scheme and solution of examination shall be set by the external examiners and two sets along with scheme of examination shall be set by the internal examiners. All the question papers shall be scrutinized by the Board of Examiners (BoE). It shall be responsibility of the BOE particularly Chairman of the BOE to maintain the quality and standard of the question papers and as well the coverage of the entire syllabus of the course.

9.16 There shall be single evaluation by the examiners for each paper. However, there shall be moderation by one of the senior examiners, either internal or external.

9.17 Board of Examiners, question paper setters and any member of the staff connected with the examination are required to maintain integrity of the examination system and the quality of the question papers.

9.18 There shall also be an **Program Assessment Committee (PAC)** comprising at-least 3 faculty members having subject expertise who shall after completion of examination process and declaration of results review the results sheets, assess the performance level of the students, measure the attainment of course outcomes, program outcomes and assess whether the program educational objectives are achieved and report to the Director of the School.

9.19 The report provided by the PAC shall be the input to the Board of Studies to review and revise the scheme of instruction and curriculum of respective program.

9.20 During unforeseen situation like the Covid-19, the tests and examination schedules, pattern of question papers and weightage distribution may be designed as per the convenience and suggestions of the board of examiners in consultation with Controller of Examination and Vice chancellor.

9.21 University may decide to use available modern technologies for writing the IAs and SEE by the students instead of traditional pen and paper.

9.22 Any deviations required to the above guidelines can be made with the written consent of the Vice Chancellor.

9.23 Online courses may be offered as per UGC norms.
For online course assessment guidelines would be as follows:

a. If the assessment is done by the course provider, then the School can accept the marks awarded by the course provider and assign the grade as per REVA University norms.

b. If the assessment is not done by the course provider then the assessment is organized by the concerned school and the procedure explained in the regulation will apply.

c. In case a student fails in an online course, s/he may be allowed to repeat the course

IAs for online courses could be avoided and will remain at the discretion of the School.

9.24 The online platforms identified could be SWAYAM, NPTEL, Coursera, Edx.org, Udemy, Udacity and any other internationally recognized platforms like MIT online, Harvard online etc.

9.25 Mapping of one or two credit online courses would be:

4 week online course – 1 credit

8 week online course / MOOC – 2 credits

12 week online course / MOOC – 3 credits

9.26 **Summary of Internal Assessment, Semester End Examination and Evaluation** Schedule is provided in the table given below (for theory courses having Credits >=2).

Summary of Internal Assessment and Evaluation Schedule

Sl. No.	Type of Assessment	When	Syllabus Covered	Max Marks	Scaled down to	Date by which the process must be completed
1	Test-1	During 8th week	First 50%	40	20	9th week
2	Test -2	During 15th Week	Remaining 50%	40	20	16th Week
3	Assignment 1/ Quiz - 1	Every week till Test-1	First 50%	10	05	9th Week
4	Assignment 2 / Quiz - 2	Every week during Test-1 and Test-2	Remaining 50%	10	05	16th Week
5	SEE	18th to 20th Week	100%	100	50	20th Week

9.27 **Summary of Internal Assessment, Semester End Examination and Evaluation** Schedule is provided in the table given below (for theory courses having Credit 1).

Summary of Internal Assessment and Evaluation Schedule

Sl. No.	Type of Assessment	When	Syllabus Covered	Max Marks	Reduced to	Date by which the process must be completed
1	Test-1	During 8th week	First 50%	25	12.5	8th week
2	Test -2	During 15th Week	Remaining 50%	25	12.5	15th Week
5	SEE	18th to 20th Week	100%	50	25	20th Week

School Board.

10 Assessment of Students Performance in Practical Courses

Lab courses are of two types: integrated labs and separate labs.

The performance in the practice tasks / experiments shall be assessed on the basis of:

a. Knowledge of relevant processes;
b. Skills and operations involved;
c. Results / products including calculation and reporting

10.1 Assessment of lab courses

10.1.1 Assessment of Separate lab course

The 50 marks of CIA is based on the performance of students in each lab experiment for a lab course that shall further be allocated as under:

i	Conduction of regular practical / experiments throughout the semester(Continuous evaluation)	20 marks
ii	Maintenance of lab records	10 marks
iii	Performance of internal lab test to be conducted after completion of all the experiments before last working day of the semester	20 marks
	Total	**50 marks**

10.2 Assessment of integrated lab course

The 10 marks meant for CIA is based on the performance of students in each lab experiment for integrated lab course that shall further be allocated as under.

Integrated lab is evaluated and awarded marks should meet the requirement of assignment/quiz/field work component of respective theory course having integrated lab component. No separate assignment/quiz/filed work is assessed for such courses.

i	Conduction of regular practical / experiments throughout the semester (continuous evaluation)	05 marks
ii	Maintenance of lab records and performance of internal lab test to be conducted after completion of all the experiments before last working day of the semester	05 marks
	Total	**10 marks**

10.3 The 50 marks meant for SEE in case of separate lab course shall be allocated as under:

i	Conduction of practical (experiment)	30 marks
ii	Write up about the experiment/tabulation/results/inference	10 marks
iii	Viva Voce	10 marks
	Total	**50 marks**

Note: No Separate SEE for integrated lab course

10.4 The duration for semester-end practical examination shall be decided by the concerned

10.5 For MOOC and Online Courses assessment shall be decided by the BOS of the School.

For >= 2 credit courses

i	IA-I	25 marks
ii	IA-2	25 marks
iii	Semester end examination by the concern school board (demo, test, viva voice etc.)	50 marks
	Total	**100 marks**

For 1 credit courses

i	IA (Performance of internal test to be conducted after completion of entire syllabus)	25 marks
iii	Semester end examination by the concern school board (demo, test, viva voice etc.)	25 marks
	Total	**50 marks**

11. Evaluation of Minor Project / Major Project / Dissertation >4 credit:

Right from the initial stage of defining the problem, the candidate must submit the progress reports periodically and present his/her progress in the form of seminars in addition to the regular discussion with the supervisor. At the end of the semester, the candidate must submit final report of the project / dissertation for final evaluation. The components of evaluation are as follows:

Component – I	Periodic Progress and Progress Reports (25%)
Component – II	Demonstration and Presentation of work (25%)
Component – III	Evaluation of Report (50%)

i	Phase-1 presentation	50 marks
ii	Phase-II presentation	50 marks
III	SEE	100 marks
	Total	**200 marks**

12. Evaluation of mandatory courses: Students should maintain minimum of 75% attendance to appear for SEE of Mandatory course. The SEE should be conducted in MCQ pattern and students should get minimum pass grade to obtain the degree. There is no internal assessment.

13. Evaluation of **Skill Development Courses:** The concerned BoS shall recommend to conduct test/demo/viva-voce/MCQ to test the student knowledge.

14. Requirements to Pass a Course:

A candidate's performance from CIA and SEE will be in terms of scores, and the sum of CIA and SEE scores will be for a maximum of 100 marks (CIA = 50 , SEE = 50) and have to secure a

minimum of 40% to declare pass in the course. However, a candidate must secure a minimum of 30% (15 marks) out of 50 marks in SEE, which is compulsory.

The Grade and the Grade Point: The Grade and the Grade Point earned by the candidate in the subject will be as given below:

SGPA/CGPA	Grade Point	Letter Grade	Performance	FGP Qualitative Index
9>=CGPA<=10	10	O	Outstanding	First Class with Distinction
8>=CGPA<=9	9	A+	Excellent	
7>=CGPA<=8	8	A	Very Good	First Class
6>=CGPA<=7	7	B+	Good	
5.5>=CGPA<=6	6	B	Above Average	Second Class
5>=CGPA<=5.5	5.5	C+	Average	
4>=CGPA<=5	5	C	Satisfactory	Pass
<4CGPA	0	F	Not Satisfactory	Unsuccessful

a. Computation of SGPA and CGPA

The Following procedure to compute the Semester Grade Point Average (SGPA).

The SGPA is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses undergone by a student in a given semester, i.e : **SGPA (Si) = ∑(Ci x Gi) / ∑Ci** where Ci is the number of credits of the ith course and Gi is the grade point scored by the student in the ith course.

Illustration for Computation of SGPA and CGPA

Illustration No. 1

Course	Credit	Grade Letter	Grade Point	Credit Point (Credit x Grade)
Course 1	3	A+	9	3X9=27
Course 2	3	A	8	3X8=24
Course 3	3	B+	7	3X7=21
Course 4	4	O	10	4X10=40
Course 5	1	C	5	1X5=5
Course 6	2	B	6	2X6=12
Course 7	3	O	10	3X10=30
	19			159

Thus, **SGPA = 159 ÷ 19 = 8.37**

Illustration No. 2

Course	Credit	Grade letter	Grade Point	Credit Point (Credit x Grade point)
Course 1	4	A	8	4X8=32
Course 2	4	B+	7	4X7=28
Course 3	3	A+	9	3X9=27
Course 4	3	B+	7	3X7=21
Course 5	3	B	6	3X6=18
Course 6	3	C	5	3X5=15
Course 7	2	B+	7	2X7=21
Course 8	2	O	10	2X10=20
	24			182

Thus, SGPA = 182 ÷ 24 = 7.58

Illustration No.3

Course	**Credit**	**Grade Letter**	**Grade Point**	**Credit Point (Credit x Grade point)**
Course 1	4	O	10	4 x 10 = 40
Course 2	4	A+	9	4 x 9 = 36
Course 3	3	B+	7	3 x 7 = 21
Course 4	3	B	6	3 x 6 = 18
Course 5	3	A+	9	3 x 9 = 27
Course 6	3	B+	7	3 x 7 = 21
Course 7	2	A+	9	2 x 9 = 18
Course 8	2	A+	9	2 x 9 = 18
	24			**199**

Thus**, SGPA = 199 ÷ 24 = 8.29**

b. Cumulative Grade Point Average (CGPA):

Overall Cumulative Grade Point Average (CGPA) of a candidate after successful completion of the required number of credits (168) for B. Tech degree in Engineering & Technology is calculated taking into account all the courses undergone by a student over all the semesters of a program, i. e : **CGPA = ∑(Ci x Si) / ∑Ci**

Where Si is the SGPA of the ith semester and Ci is the total number of credits in that semester.

Illustration:

CGPA after Final Semester

Semester (ith)	No. of Credits (Ci)	SGPA (Si)	Credits x SGPA (Ci X Si)
1	21	6.83	21 x 6.83 = 143.43
2	20	7.29	20 x 7.29 = 145.8
3	25	8.11	25 x 8.11 = 202.75
4	25	7.40	25 x 7.40 = 185
5	22	8.29	22 x 8.29 = 182.38
6	23	8.58	23 x 8.58 = 197.34
7	14	9.12	14 x 9.12 = 127.68
8	10	9.25	10 x 9.25 =92.50
Cumulative	**160**		**1276.88**

Thus,

$$CGPA = \frac{1276.88}{160} = 7.98$$

c. Conversion of grades into percentage:

Conversion formula for the conversion of CGPA into Percentage is:

Percentage of marks scored = CGPA Earned x 10

Illustration: CGPA Earned 8.02 x 10=80.2

d. The SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

Classification of Results

The final grade point (FGP) to be awarded to the student is based on CGPA secured by the candidate and is given as follows.

CGPA	Grade (Numerical Index) G	Letter Grade	Performance	FGP Qualitative Index
9 >= CGPA 10	10	O	Outstanding	Distinction
8 >= CGPA < 9	9	A+	Excellent	
7 >= CGPA < 8	8	A	Very Good	First Class
6 >= CGPA < 7	7	B+	Good	
5.5> = CGPA < 6	6	B	Above average	Second Class
> 5 CGPA < 5.5	5.5	C+	Average	
> 4 CGPA <5	5	C	Satisfactory	Pass
< 4 CGPA	0	F	Unsatisfactory	Unsuccessful

Overall percentage=10*CGPA

e. Provisional Grade Card: The tentative / provisional grade card will be issued by the Controller of Examinations at the end of every semester indicating the courses completed successfully. The provisional grade card provides **Semester Grade Point Average (SGPA).**

f. Final Grade Card: Upon successful completion of B Tech Degree a Final Grade card consisting of grades of all courses successfully completed by the candidate will be issued by the Controller of Examinations.

14.2 Attendance Requirement

14.2.1. All students must attend every lecture, tutorial and practical classes.

14.2.2. Student must maintain a minimum attendance of 75% in each course (Theory and Practical) and 75% attendance in aggregate of all courses in a semester, with a provision of condonation of 10% of the attendance by the Vice-Chancellor on the specific recommendation of the Director of the School.

14.2.3. In case a student is on approved leave of absence (e g:- representing the University in sports, games or athletics, placement activities, NCC, NSS activities and such others) and

/ or any other such contingencies like medical emergencies, the attendance requirement shall be minimum of 65% of the classes taught.

14.2.4. Any student with less than 75% of attendance in individual courses of respective semester including practical courses / field visits etc., shall not be permitted to appear to SEE in the respective course.

15. Re-Registration and Re-Admission

15.1 In case a candidate's class attendance in aggregate of all courses in a semester is less than 75% or as stipulated by the University, such a candidate is considered as dropped the semester and is not allowed to appear for semester end examination and he / she shall have to seek re-admission to that semester during subsequent semester / year within a stipulated period.

15.2 In such case where in a candidate drops all the courses in a semester due to personal reasons, it is considered that the candidate has dropped the semester and he / she shall seek re-admission to such dropped semester.

16. Absence during Internal Test:

In case a student has been absent from an internal test due to the illness or other contingencies he / she may give a request along with necessary supporting documents and certification from the concerned class teacher / authorized personnel to the concerned Director of the School, for conducting a separate internal test. The Director of the School may consider such request depending on the merit of the case and after consultation with course instructor and class teacher, and arrange to conduct a special internal test for such candidate(s) well in advance before the Semester End Examination of that respective semester. Under no circumstances internal tests shall be held / assignments are accepted after Semester End Examination.

17. Provision for Appeal

If a candidate is not satisfied with the evaluation of Internal Assessment components (Internal Tests and Assignments), he/she can approach the Grievance Cell with the written submission

together with all facts, the assignments, and test papers, which were evaluated. He/she can do so before the commencement of respective semester-end examination. The Grievance

Cell is empowered to revise the marks if the case is genuine and is also empowered to levy penalty as prescribed by the University on the candidate if his/her submission is found to be baseless and unduly motivated. This Cell may recommend for taking disciplinary/corrective action on an evaluator if he/she is found guilty. The decision taken by the Grievance committee is final.

17.1 Grievance Committee:

In case of students having any grievances regarding the conduct of examination, evaluation and announcement of results, such students can approach Grievance Committee for redressal of grievances.

For every program there will be one grievance committee. The composition of the grievance committee is as follows:-

- The Controller of Examinations - Ex-officio Chairman / Convener
- One Senior Faculty Member (other than those concerned with the evaluation of the course concerned) drawn from the school / department/discipline and/or from the sister schools / departments/sister disciplines – Member.
- One Senior Faculty Members / Subject Experts drawn from outside the University school / department – Member.

18. Eligibility to Appear for Semester End Examination (SEE)

Only those students who fulfil a minimum of 75% attendance in aggregate of all the courses including practical courses / field visits etc., and 70% attendance in each courses shall be eligible to appear for Semester End Examination

19. Provision for Supplementary Examination

In case a candidate fails to secure a minimum of 30% marks out of 50 (15 marks) in Semester End Examination (SEE) and a minimum of 40% marks (out of 100) together with IA and SEE to declare pass in the course, such candidate shall seek supplementary examination of only such course(s) wherein his / her performance is declared unsuccessful. The supplementary examinations are conducted after the announcement of even/Odd semester examination results. The candidate who is unsuccessful in each course(s) shall appear for supplementary examination of odd and even semester course(s) to seek for improvement of the performance.

a. A student failed in any course is eligible to take supplementary exam under following category for each course: either to improve internal marks (IA1, IA2, and Assignment/Quiz), or to improve SEE.

b. Supplementary exam is permitted only during summer vacation (between even and odd semester break)

c. Eligibility to register for supplementary exam is that the student should have maintained pre-requisite attendance of ≥ 75% in respective semester.

d. No separate additional classes would be conducted for the students availing this facility.

e. Every student should pay the supplementary exam fee for each course as prescribed by the university.

20. Provision to Carry Forward the Failed Subjects / Courses:

Students who have failed in courses totaling 16 credits or fewer across both odd and even semesters combined will be allowed to proceed to the next semester of the following year(s) of their academic program.

For vertical progression, students must clear all the courses of first year to be eligible to take admission to third year and they have to clear all the courses till second year to be eligible to take admission to fourth year. For lateral entry students, students must clear all the courses of second year to be eligible to take admission to fourth year.

Case 1: A student who has failed in a maximum of 16 credits in 1st and 2nd semester together shall move to the 3rd semester of the succeeding year.

Case 2: A student who has failed in a maximum of 16 credits from semester 1 to 4 together shall move to the 5th semester of the succeeding year only if he/she successfully completes all the courses of first and second semester.

Case 3: A students who has failed in a maximum of 16 credits from semester 3 to 6 together shall move to the 7th semester of the succeeding year only if he/she successfully completes all the courses of third and Fourth semester.

21. Re-evaluation of Answer Scripts and Announcement of Re-evaluation Results

After declaration of the results of programs within next 10 days, if any candidate wishes to apply for Photocopy/Revaluation (only theory courses), s/he shall apply to the Controller of Examinations, by paying the prescribed fees notified by the University from time to time. The photocopies of the said answer books shall be made available within next TEN working days after the last date prescribed for receipt of the application at the Office of the Controller of Examinations. Photocopies will not be issued for practical/drawing/audit courses.

22. Results of Re-Evaluation will be announced within TWENTY working days (except for third evaluation).

23. Regarding any specific case of ambiguity and unsolved problem, the decision of the Vice-Chancellor shall be final.

24. All assessments must be done by the respective Schools as per the guidelines issued by the Controller of Examinations. However, the responsibility of announcing final examination results and issuing official transcripts to the students lies with the office of the Controller of Examinations.

25. For lateral entry students, the minimum credits to be earned for the award of the degree would be the credits earned in 3 years from 2nd year to 4th year.

ABOUT THE SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

The Department has a rich blend of experienced and committed faculty who are well qualified in various aspects of Computer Science and Engineering apart from the numerous state-of-the-art digital classrooms and laboratories having modern computing equipment. The Department offers B.Tech in Computer Science and Engineering and postgraduate programs offered in the departmentare: M.Tech in Computer Science and Engineering (Full time and Part time).

In addition, the department has a unique academic collaboration with the University of Alabama in Huntsville to jointly offer an MS program in Computer Science. In addition, the department has a research center in which students can conduct cutting edge research leading to a Ph.D degree.

Curricula of both undergraduate and postgraduate programs have been designed through a collaboration of academic and industry experts in order to bridge the growing gap between industry and academia. This makes the program highly practical-oriented, and thus industry-resilient. The B.Tech program aims to create quality human resources to play leading roles in the contemporary, competitive industrial and corporate world.

The master's degree focus on quality research and design in the core and application areas of computing to foster a sustainable world and to enhance the global quality of life by adopting enhanced design techniques and applications. This thought is reflected in the various courses offered in the masters' programs.

Vision

Department of Computer Science and Engineering aspires to create a pool of high-calibre technologists and researchers in the field of computer science and engineering who have potential to contribute for development of the nation and society with their expertise, skills, innovative problem-solving abilities and strong ethical values.

Mission

MD1: To create center of excellence where new ideas flourish and from which emerge tomorrow's researchers, scholars, leaders, and innovators.

MD2: Provide quality education in both theoretical and applied foundations of computer science and engineering, related inter-disciplinary areas and train students to effectively apply the knowledge to solve real-world problems.

MD3: Amplify student's potential for life-long high-quality careers and make them competitive in ever-changing and challenging global work environment.

MD4: Forge research and academic collaboration with industries and top global universities in order to provide students with greater opportunities.

MD5: Support the society by encouraging and participating in technology transfer.

Quality Policy

The School of Computer Science and Engineering is committed to excellence through following policies.

1. Impart quality education by providing state of art curriculum, experimental learning, and state-of- the-art labs.
2. Enhance skill set of faculty members through faculty development programs and interactionwith academia and industries.
3. Inculcate competency in software/hardware design and programming through co-curricular activities like Hackathon, Project exihibition, Internship and Entrepreneurship Programmes.
4. Provide soft skill and skill development training for personality development and better placement.
5. Promote innovation and research culture among students and support faculty members for better research and development activity

21	Srujana Reddy Yerua(R21EH055) A Section AI&DS	III Year Student
22	Ronak (R21EH153) C Section AI&DS	III Year Student

B.TECH in ARTIFICIAL INTELLIGENCE AND DATA SCIENCE PROGRAM

Program Overview

The Department of Computer Science and Engineering at REVA UNIVERSITY proudly launches an innovative B. Tech in Artificial Intelligence and Data Science programme during the academic year 2021–2022. Due to the exploration of the Internet, AI and DS deliver data-driven solutions using computational principles, methods, and systems for extracting information from data and moder n computational systems that reveal capabilities of insight, way of thinking, learning, and action that are typical of human intelligence. This course to create motivated, innovative, creative thinking graduates to fill ICT positions across sectors who can conceptualize, design, analyze, and develop ICT applications to meet the modern-day requirements.

B. Tech in Artificial Intelligence & Data Science Programme prepares students with the skills to perform intelligent data analysis which is a key component in numerous real-world applications. During the past ten years, data science has emerged as one of the most high-growth, dynamic, and lucrative careers in technology. This course aims at providing not only the core technologies such as artificial intelligence, data mining, and data modeling but also gives intensive inputs in areas of machine learning and big data analytics. By this course, the students will gain cross-disciplinary skills across fields such as statistics, computer science, machine learning, and logic, data scientists and may have career opportunities in healthcare, business, e-commerce, social networking companies, climatology, biotechnology, genetics, and other important areas. The major focus of this programme is to equip students with statistical, mathematical reasoning, machine learning, knowledge discovery, and visualization skills.

The course covers the core principles of Data Science, the process of ethically acquiring, engineering, analyzing, visualizing, and ultimately monetizing data. It equips it with the basic tools and techniques of data handling, exploratory data analysis, data visualization, and data-based inference.

The course also introduces the fundamentals of Artificial Intelligence – state-space representation, search strategies, and learning algorithms. The course motivates students to work closely with data and make data-driven decisions in the field of study. The course also touches upon the ethical issues in Data Science and Artificial Intelligence and motivates students to explore the cutting-edge applications related to Big Data, Neural Networks, and Deep Learning. Applications like Alexa/Siri, Erica/Sophia, Tesla's driverless car, Chatbot, etc are becoming common. AI-powered Robots can take over human life-endangering jobs like firefighting, deep-sea oil drilling, mining works with high mortality rates, etc. The B. Tech., in Artificial Intelligence and Data Science curriculum developed by the faculty at the Department of Computer Science and Engineering, is outcome-based and it comprises required theoretical concepts and practical skills in the domain. By undergoing this programme, students

develop critical, innovative, creative thinking and problem-solving abilities for a smooth transition from academic to the real-life work environment. In addition, students are trained in interdisciplinary topics and attitudinal skills to enhance their scope. The above-mentioned features of the programme, advanced teaching, and learning resources, and the experience of the faculty members with their strong connections with the ICT sector make this programme unique.

Program Educational Objectives (PEOs)

The program helps to develop critical, analytical, innovative, creative and problem-solving abilities amongst its graduates. The programme makes the graduates employable as Software Engineers across sectors. With further education and earning of higher-level degrees help the graduates to pursue a career in academics or scientific organisations as researchers.

The Program Educational Objectives (PEOs):

After few years of graduation, the graduates of B. Tech in Artificial Intelligence& Data Science will be able to:

- **PEO-1:** Have a successful professional career in industry, government, academia and defence as an innovative engineer in a team.
- **PEO-2:** Develop a code and solutions to industry and societal needs in a rapid changing technological environment and communicate with clients as an entrepreneur.
- **PEO-3:** Pursue higher studies and continue to learn by participating in conferences, seminars, etc.

Program Outcomes (POs)

On successful completion of the program, the graduates of B. Tech in Artificial Intelligence & Data Science program will be able to:

- **PO-1: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals for the solution of complex problems in Artificial Intelligence & Data Science.

- **PO-2: Problem analysis:** Identity, formulate, research literature, and analyze engineering problems to arrive at substantiated conclusions using first principles of mathematics, natural, and engineering sciences.

- **PO-3: Design/development of solution:** Design solutions for complex engineering problems and design system components, processes to meet the specifications with consideration for public health and safety, and the cultural, societal, and environmental considerations.

- **PO-4: Conduct Investigation of complex problem:** Use research-based knowledge including design of experiments, analysis, and interpretation of data, and synthesis of the information to provide valid conclusions.

- **PO-5: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

- **PO-6: Engineering and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

- **PO-7: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

- **PO-8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilit$_{3}$i$_{0}$es and norms of the engineering practice

- **PO-9: Indiviual and Team work:** Function effectively as an individual, and as a member or leader in teams, and multidisciplinary settings.

- **PO-10: Communication:** Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective report documentation. Make effective presentations, and give and receive clear instructions.

- **PO-11: Project management and finance:** Demonstrate knowledge and understanding of engineering and management principles and apply these to one's work, as a member and leader in a team. Manage projects in multidisciplinary environments.

- **PO-12: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs)

On successful completion of the program, the graduates of B. Tech in Artificial Intelligence & Data Science program will be able to:

- **PSO-1:** Understand, analyze and develop essential proficiency in the areas related to data science and artificial intelligence in terms of underlying statistical and computational principles and apply the knowledge to solve practical problems.

- **PSO-2:** Ability to implement Artificial Intelligence and data science techniques such as search algorithms, neural networks, machine learning, and data analytics for solving a problem and designing novel algorithms for successful career and entrepreneurship.

- **PSO-3:** Use modern tools and techniques in the area of Artificial Intelligence& Data Science.

REVA UNIVERSITY
Bengaluru, India

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

HANDBOOK

B. Tech in Artificial Intelligence and Data Science

2024-28

Rukmini Knowledge Park,
Kattigenahalli, Yelahanka, Bangalore - 560 064
Phone No: +91-080-66226622, Fax: 080-28478539

Rukmini Educational
Charitable Trust

www.reva.edu
.in

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

SCHEME OF INSTRUCTIONS (2024 – 2028 BATCH)

B. Tech in Artificial Intelligence and Data Science

School of Computer Science and Engineering

I Semester (Chemistry Cycle)												
Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS0103	Multivariable Calculus and Linear Algebra	FC	3	0	0	3	3	50	50	100	BSC
2	B24AS0105	Chemical Technology for computing	FC	2	1	0	3	4	50	50	100	BSC
3	B24EE0101	Basics of Electrical and Electronics Engineering	HC	3	0	0	3	3	50	50	100	ESC
4	B24CS0104	Fundamentals of Data Science	HC	2	0	0	2	2	50	50	100	ESC
5	B24CI0109	Fundamentals of C programming	HC	1	1	0	2	3	50	50	100	ESC
6	B24ED0102	Fundamentals and Applications of Civil Engineering	HC	1	1	0	2	3	50	50	100	ESC
7	B24CS0108	Fundamentals of Data Science lab	HC	0	0	1	1	2	25	25	50	ESC
8	B24EE0102	Basics of Electrical and Electronics lab	HC	0	0	1	1	2	25	25	50	ESC
9	B24ME0101	Computer Aided Engineering Drawing	HC	1	0	1	2	3	50	50	100	ESC
10	B24CI0110	Fundamentals of C programming lab	HC	0	0	1	1	2	25	25	50	ESC
11	B24EN0102	Finance and Management	FC	1	0	0	1	1	25	25	50	HSMC
12	B24CS0109	Foundation for Learning	MC	0	0	0	0	1	25	25	50	
TOTAL				15	2	4	21	28	475	475	950	
TOTAL SEMESTER CREDITS				21								
TOTAL CUMULATIVE CREDITS				21								
TOTAL CONTACT HOURS				28								
TOTAL MARKS				950								

II Semester (Physics Cycle)												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS0203	Probability and Statistics	FC	3	0	0	3	3	50	50	100	BSC
2	B24AS0106	Physics for Computer Science	FC	3	0	0	3	3	50	50	100	BSC
3	B24ME0105	Fundamentals of Mechanical Engineering	HC	3	0	0	3	3	50	50	100	ESC
4	B24CI0201	Advanced C programming and Applications	HC	1	1	0	2	3	50	50	100	ESC
5	B24EN0101	Internet of Things	HC	1	0	1	2	3	50	50	100	ESC
6	B24AS0208	Physics for Computer Science Lab	FC	0	0	1	1	2	25	25	50	ESC
7	B24CI0202	Advanced C programming and Applications Lab	HC	0	0	1	1	2	25	25	50	ESC
8	B24ME0102	Innovation & Entrepreneurship	FC	1	0	1	2	3	50	50	100	HSMC
9	B24AH0103	Communicative English	FC	0	0	1	1	2	25	25	50	HSMC
10	B24CIET01	Engineering Exploration	ETC	1	0	0	1	1	25	25	50	HSMC
11	B24CSET02	Emerging Technologies on Full Stack Web Development	ETC	1	0	0	1	1	25	25	50	ETC
TOTAL				14	1	5	20	26	425	425	850	
TOTAL SEMESTER CREDITS				20								
TOTAL CUMULATIVE CREDITS				41								
TOTAL CONTACT HOURS				26								
TOTAL MARKS				850								

III Semester												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS	Discrete Mathematics and Graph Theory	FC	3	0	0	3	3	50	50	100	BSC
2	B24EF0301	Data Structures	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0301	Linux Operating System	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0302	Object Oriented Programming with C++	HC	3	0	0	3	3	50	50	100	PCC
5	B24EHS31X	Professional –Elective-1	SC	3	0	0	3	3	50	50	100	PEC
6	B24EF0304	Data Structures lab using C	HC	0	0	1	1	2	25	25	50	PCC
7	B24EH0303	Linux Operating Systems lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0304	Object Oriented Programming with C++ Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0305	Course-based Project (SDP) C++/Java/Prompt Engineering	SDC	0	0	1	1	2	25	25	50	Proj
10	B24CI0309	Introduction to Design Thinking	HC	1	0	1	2	3	50	50	100	ESC
11	B24EN0308	Technical Documentation	FC	1	0	0	1	1	25	25	50	HSMC
12	B24MEM301	Indian Constitution and Cyber Law	MC	1	0	0	0	1				HSMC
13	B24EF0308	AEC-1 (Ability Enhancement Course) (Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				18	0	6	23	30	450	450	900	
TOTAL SEMESTER CREDITS				23								
TOTAL CUMULATIVE CREDITS				64								
TOTAL CONTACT HOURS				30								
TOTAL MARKS				900								

IV Semester												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0401	Optimization Techniques	FC	3	0	0	3	3	50	50	100	BSC
2	B24EF0401	Design and Analysis of Algorithms	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0402	Artificial Intelligence and Machine Learning	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0403	Data Engineering	HC	3	0	0	3	3	50	50	100	PCC
5	B24EH0404	Distributed Systems	HC	3	0	0	3	3	50	50	100	PCC
6	B24EHS41X	Professional Elective-2	SC	3	0	0	3	3	50	50	100	PEC
7	B24EF0405	Design and Analysis of Algorithms Lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0405	Artificial Intelligence and Machine Learning Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0406	Data Engineering Lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EH0407	Course-based Project (SDP) AI	SDC	0	0	1	1	2	25	25	50	Proj
11	B24CS0401	Professional Ethics	FC	1	0	0	1	1	25	25	50	HSMC
12	B24EE0310	Universal Human Values	HC	1	0	0	1	1	25	25	50	HSMC
13	B24AS0403	Environmental Science	MC	1	0	0	0	1				HSMC
14	B24EF0409	AEC-2 (Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				21	0	5	25	30	475	475	950	
TOTAL SEMESTER CREDITS				25								
TOTAL CUMULATIVE CREDITS				89								
TOTAL CONTACT HOURS				30								
TOTAL MARKS				950								

V Semester												
Sl. No	Course Code	Title of the Course	HC/FC /SC/OE/MC	Credit Pattern				Contact Hours /Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0501	Feature Engineering with Python	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0502	Image processing	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0503	Big data Analytics	HC	3	0	0	3	3	50	50	100	PCC
4	B24EHS51X	Professional Elective-3	SC	3	0	0	3	3	50	50	100	PEC
5	B24EHS52X	Professional Elective-4	SC	3	0	0	3	3	50	50	100	PEC
6	B24EFO5XX	Open Elective-1	OE	3	0	0	3	3	50	50	100	OE
7	B24EH0504	Feature Engineering with Python Lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0505	Image processing lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0506	Big data Analytics lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EF0507	AEC-3(Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
11	B24ED0501	Indian Knowledge System	MC	1	0	0	0	1	25	25	50	HSMC
TOTAL				19	0	4	22	27	425	425	850	
TOTAL SEMESTER CREDITS				22								
TOTAL CUMULATIVE CREDITS				111								
TOTAL CONTACT HOURS				27								
TOTAL MARKS				850								

VI Semester

Sl. No	Course Code	Title of the Course	HC/ FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0601	Speech and Language Processing with AI	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0602	Generative Artificial Intelligence	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0603	Cloud Computing Tools	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0604	Human-Computer Interaction	HC	3	0	0	3	3	50	50	100	PCC
5	B24EHS61X	Professional Elective-5	SC	3	0	0	3	3	50	50	100	PEC
6	B24EFO61X	Open Elective-2	OE	3	0	0	3	3	50	50	100	OE
7	B24EH0605	Generative Artificial Intelligence lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0606	Cloud Computing Tools Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0607	Speech and Language Processing Lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EH0608	Mini Project	HC	0	0	2	2	4	50	50	100	Proj
11	B24EF0609	AEC-4(Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				18	0	6	24	30	450	450	900	
TOTAL SEMESTER CREDITS				24								
TOTAL CUMULATIVE CREDITS				135								
TOTAL CONTACT HOURS				30								
TOTAL MARKS				900								

VII Semester												
Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0701	Multi Large Language Models	HC	3	0	0	3	3	50	50	100	PCC
2	B24EHS71X	Professional Elective-6	SC	3	0	0	3	3	50	50	100	PEC
3	B24EH0702	Capstone Project Phase-1	HC	0	0	2	2	4	50	50	100	Proj
4	B24EH0703	Internship / Global Certification	HC	0	0	2	2	4	50	50	100	Intern
TOTAL				6	0	4	10	14	200	200	400	
TOTAL SEMESTER CREDITS				10								
TOTAL CUMULATIVE CREDITS				145								
TOTAL CONTACT HOURS				14								
TOTAL MARKS				400								

VIII Semester												
Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0801	AI in Cybersecurity	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0802	Capstone project phase-II	Proj	0	0	10	10	20	50	50	100	Proj
3	B24EH0803	Internship /Global Certification	HC	0	0	2	2	4	50	50	100	Intern
TOTAL				3	0	12	15	27	150	150	300	
TOTAL SEMESTER CREDITS				15								
TOTAL CUMULATIVE CREDITS				160								
TOTAL CONTACT HOURS				27								
TOTAL MARKS				300								

Sem	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**Total**
Credit	21	20	23	25	22	24	10	15	160

List of Professional Electives

Semester (Electives) / Choices of courses	Course code	T1 Foundational & Programming	Course code	T2 Computing	Course code	T3 Network and Cyber Security
III (PE1)	B24EHS311	Java Programming	B24EHS312	Formal Language and Automation	B24EHS213	Computer Networks
IV(PE2)	B24EHS411	Agile Software Development and DevOps	B24EHS412	Game Theory	B24EHS413	Cryptography and Network Security
V(PE3)	B24EHS511	Reinforcement Learning	B24EHS512	Cognitive Computing	B24EHS513	Block Chain Technology and Applications
V(PE4)	B24EHS521	Genomics	B24EHS522	High Performance and Quantum Computing	B24EHS523	Robotic Process Automation
VI(PE5)	B24EHS611	Machine Learning Operations (MLOps)	B24EHS612	Signals and Systems	B24EHS613	Cyber-Physical System
VII(PE6)	B24EHS711	Augmented Reality and Virtual Reality	B24EHS712	Industrial Computing	B24EHS713	Autonomous Drones

List of Open Electives

SEM	Course Code	Course Title & Code	Course Code	Course Title & Code
V (OE1)	B24EFO511	Data Science with Python	B24EFO512	Database management system
VI (OE2)	B24EFO611	Augmented Reality and Virtual Reality	B24EFO612	Data Communications and Systems

Nomenclature: **L:** Lecture, **T:** Tutorial, **P:** Practical/Practice/Hands-on, **HC**: Hard Core, **SC**: Soft Core, **FC**: Foundation Core, **OE**: Open Elective, **SDC:** Skill Development Course, **CIE**: Continuous Internal Evaluation, **SEE**: Semester End Examination, **BSC**: Basic Science Course, **HSMC**: Humanities, Social science and Management Course, **ESC**: Engineering Science Course, **PCC**: Program Core Course, **PEC**: Professional Elective Course, **MC**: Mandatory Course, **PROJ**: Project work/Internship

***Note*:** SDC-1 will be hands-on based skill enhancement course that create expertise in the domain of respective engineering

AICTE Compliance: Category wise credit distribution for HSMC, BSC, ESC, PCC, PE, PEC, OE

BATCH 2024-28		AICTE	
Course Type	**Credit**	**Course Type**	**Credit**
BSC	18	BSC	25
ESC	25	ESC	24
HSMC	7	HSMC	12
PCC	60	PCC	48
PEC	18	PEC	18
OE	6	OE	18
Project, Internship	20	Project, Internship	15
ETC & AEC	5	**TOTAL**	160
FC	1		
TOTAL	160		

BATCH 2024-28									
Course Type	**Credit**								
Semester	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**TOTAL**
BSC	6	6	3	3					18
ESC	14	9	2						25
HSMC	1	3	1	2					7
PCC			12	15	12	15	3	3	60
PEC			3	3	6	3	3		18
OE					3	3			6
Project, Internship			1	1		2	4	12	20
ETC & AEC	0	1	1	1	1	1			5
FC		1							1
TOTAL									160

Common Courses for AY 2024-28

S. no	Semester	COURSE CODE	COURSE NAME	HC/FC/SC/OE/MC	L	T	P	TOTAL CREDITS	CONTACT HOURS	AICTE course category
1	I	B24EN0102	Finance and management	FC	1	0	0	1	1	HSMC
2	II	B24ME0102	Innovation and Entrepreneurship	FC	1	0	1	2	3	HSMC
3		B24AH0103	Communicative English	FC	0	0	1	1	2	HSMC
4	III	B24CI0309	Introduction to Design Thinking	HC	1	0	1	2	3	ESC
5		B24EN0308	Technical Documentation	FC	1	0	0	1	1	HSMC
6		B24MEM301	Indian Constitution and Cyber law	MC	1	0	0	0	1	HSMC
7	IV	B24CS0401	Professional Ethics	FC	1	0	0	1	1	HSMC
8		B24EE0310	Universal Human Values	HC	1	0	0	1	1	HSMC
9		B24AS0403	Environmental Science	MC	1	0	0	0	1	HSMC
10	V	B24ED0501	Indian Knowledge Systems	MC	1	0	0	0	1	HSMC

I Semester (Chemistry Cycle)

Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS0103	Multivariable Calculus and Linear Algebra	FC	3	0	0	3	3	50	50	100	BSC
2	B24AS0105	Chemical Technology for computing	FC	2	1	0	3	4	50	50	100	BSC
3	B24EE0101	Basics of Electrical and Electronics Engineering	HC	3	0	0	3	3	50	50	100	ESC
4	B24CS0104	Fundamentals of Data Science	HC	2	0	0	2	2	50	50	100	ESC
5	B24CI0109	Fundamentals of C programming	HC	1	1	0	2	3	50	50	100	ESC
6	B24ED0102	Fundamentals and Applications of Civil Engineering	HC	1	1	0	2	3	50	50	100	ESC
7	B24CS0108	Fundamentals of Data Science lab	HC	0	0	1	1	2	25	25	50	ESC
8	B24EE0102	Basics of Electrical and Electronics lab	HC	0	0	1	1	2	25	25	50	ESC
9	B24ME0101	Computer Aided Engineering Drawing	HC	1	0	1	2	3	50	50	100	ESC
10	B24CI0110	Fundamentals of C programming lab	HC	0	0	1	1	2	25	25	50	ESC
11	B24EN0102	Finance and Management	FC	1	0	0	1	1	25	25	50	HSMC
12	B24CS0109	Foundation for Learning	MC	0	0	0	0	1	25	25	50	
TOTAL				15	2	4	21	29	475	475	950	
TOTAL SEMESTER CREDITS				21								
TOTAL CUMULATIVE CREDITS				21								
TOTAL CONTACT HOURS				28								
TOTAL MARKS				950								

Course Title	Multivariable Calculus and Linear Algebra				Course Type		FC	
Course	B24AS0103	Credits	3		Class		I Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	0	0	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

The course Differential Calculus deals with the basic aspects differential calculus. The students of Computer Science are equally benefited with this course as stepping stone to the broad areas of calculus. This course familiarize students with important concepts coming under differential calculus and to develop strong foundations on these concepts. In Computer Science, Calculus is used for machine learning, data mining, scientific computing, image processing, and creating the graphics and 3D visuals for simulations. Calculus is also used in a wide array software program that require it. Linear algebra provides concepts that are crucial to many areas of computer science, including graphics, image processing, cryptography, machine learning, optimization, graph algorithms, information retrieval and web search.

COURSE OBJECTIVE (S):

The objective of this course is to:

1. Explain the scalar and vector point functions and their operations.
2. Illustrate how to find angle between polar curves with a suitable example.
3. Demonstrate the use of radius of curvature of the curves can be best suited for machine learning techniques with big data analytics.
4. Describe the concepts of Linear algebra and calculus theory.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO-1	Apply knowledge of divergence, curl, and scalar potential in the vector field	1,2,4	1
CO-2	Apply Taylor's and Maclaurin's series for finding series expansions of the functions and approximating values	1,2,4	1
CO-3	Evaluate the limit using L' Hospital rule after identifying an indeterminate form.	1,2,4	1
CO-4	Solve the given system of equations by Gauss elimination and the Gauss Jordon method for the consistent equations.	1,2,4	1
CO-5	Determine the Eigen values, the corresponding Eigen vectors and diagonalize the given square matrix.	1,2,4	1
CO-6	Apply Rayleigh's Power method to find the largest Eigen value and the corresponding Eigen vector.	1,2,4	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO-1			√			
CO-2			√			
CO-3					√	

CO-4			√			
CO-5					√	
CO-6			√			

COURSE ARTICULATION MATRIX

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
CO-1	3	3		1									1		
CO-2	3	3		2									1		
CO-3	3	2		1									1		
CO-4	3	3		1									1		
CO-5	3	3		1									1		
CO-6	3	2		1									1		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Vector Calculus: Velocity, Acceleration, Tangent and normal vectors, Gradient, Divergence, Curl, Solenoidal and Irrotational vectors, Scalar potential, Vector identities (Basic identities)**.** *Case study on perception learning in neural networks.* ******** *Applications: vector theory for data transmission, social network analysis*.

Differential Calculus: nth derivatives of standard function (without proof, *simple problems), Leibnitz theorem (without proof)-simple problems, Taylors series and McLaurin's series expansion for a function of one variable (only problems), Indeterminate forms 'solve using L- Hospital's rule. *Case study on image segmentation using edge detection.* ******** *Applications: creating graphs or visuals, simulations, coding in applications, creating statistic solvers.*

LinearAlgebra-1: Echelon form, Normal form of a matrix, Rank of Matrix, Gauss-Jordon method to find the inverse of a matrix, Gauss elimination, and Gauss-Jordon method to solve the system of equations. Linear Algebra for statistics. *Case study on regression analysis.* ******** *Application: Image processing, computer graphics, encryption, and decryption of the codes.*

Linear Algebra-2: Linear transformation, Eigen values, and Eigen Vectors up to 3*3 matrices, Diagonalization for 2*2 matrices, Rayleigh power method to determine largest Eigen value and the corresponding Eigen vector, Complex matrices.

******** *Application: (i) Additional information providing to students only for knowledge.*

(ii) Major part of assignments questions chosen from applications.

(iii) Students' presentations/seminars topics chosen from applications.

TEXT BOOKS:

1. Theodore Shifrin, "Multi-Variable Calculus and Linear Algebra with Applications", Wiley, 1st edition, Volume 2, 2018.

2. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 43nd edition, 2015.

3. Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley Publications, 9th edition,2013.

4. Ron Larson, "Multivariable Calculus, Cengage Learning", 10th Edition, 2013.

REFERENCE BOOKS:

1. B.V. Ramana, "Higher Engineering Mathematics", Tata McGraw Hill Publications, 19th Reprint edition, 2013.

2. R.K.Jain and S.R.K.Iyengar, "Advanced Engineering Mathematics", Narosa Publishing House, 4th edition, 2016.

3. Stanley I. Grossman, "Multivariable Calculus, Linear Algebra, and Differential Equations", 2nd edition, Academic Press 1986.

JOURNALS/MAGAZINE:

https://www.sciencedirect.com/journal/linear-algebra-and-its-applications

SWAYAM/NPTEL/MOOCs:

1. https://nptel.ac.in/courses/111/107/111107108/
2. https://nptel.ac.in/noc/courses/noc20/SEM1/noc20-ma07/
3. https://nptel.ac.in/courses/111/106/111106051/
4. https://nptel.ac.in/courses/111/104/111104092/
5. https://nptel.ac.in/courses/111/104/111104085/

SELF-LEARNING EXERCISES:

Vector spaces, Curvilinear co-ordinates: Cylindrical and spherical co-ordinates

Course title	Chemical Technology for Computing			Course type		FC	
Course code	B24AS0105	Credit	3	Class		I semester	
	LTP	Credit	Contact Hours	Workload	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	2	2	2				
	Tutorial	1	2	2	Theory	Practical/tutorial	CIE	SEE
	Practical	-		-	28	28	50	50
	Total	3	4	4				

COURSE OVERVIEW:

Engineering chemistry integrates highly pertinent subjects that are suitable for engineering students enlightening them about the significance of several facets of fundamental science in engineering. Engineering chemistry includes the study of instrumental technology analysis methods, technologies for clean and green energy storage and conversion devices, and novel materials for electronic devices. Additionally, the course emphasizes the study of the chemical properties and uses of functional materials in several engineering fields. This field of science is highly interdisciplinary and provides an opportunity for students to enhance their understanding of engineering principles concerning sustainable energy, conversion, and storage devices, semiconducting materials, and polymeric materials. These areas have become especially interesting for engineering research. The topic encompasses a wide range of engineering materials, their characteristics, and their use in the engineering domain.

COURSE OBJECTIVES:

The objectives of the course are to:

1. Understand the electronic transitions in materials, instrumental methods of analysis in technology using various instruments.
2. Illustrate renewable energy storage and conversion device technologies for batteries, super capacitors of operating mechanisms and applications.
3. Acquire a comprehensive understanding of advanced materials used in electronic devices, with a specific emphasis on the significance of semiconducting materials and the principles of metal finishing in industrial procedures.
4. Identify efficient polymeric materials, liquid crystals in display technologies and electronic applications.

COURSE OUTCOMES (COs):

After successful completion of this course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Interpret the instrumental methods of analysis in technology, electronic transitions in materials and the detection of samples using various sensors.	1,2,3,5	1
CO2	Demonstrate the principles and uses of clean-green energy storage and conversion device technologies.	1,2,3,5,7	1
CO3	Identify the significance of modern materials in electronic devices and the theory of electroplating in industrial techniques.	1,2,3,5,7	1
CO4	Analyze the importance of polymeric materials and Nano materials for the various applications.	1,2,3,6	1
CO5	Illustrate the fundamental principles of electrode systems, batteries, and super capacitors.	1,2,3,5,7	1
CO6	Explain the practical uses of advanced materials in engineering and technology.	1,2,3,7	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3				√		
CO4				√		
CO5			√			
CO6				√		

COURSE ARTICULATION MATRIX:

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1				1						1		
CO2	2	1	3		1		2						1		
CO3	3	2	1		1		1						1		
CO4	3	2	1			1							1		
CO5	3	2	1		1		2						1		
CO6	3	2	1				1						1		

Note:1-Low; 2-Medium; 3-High

COURSE CONTENTS:

Sensors: Introduction, types, principles, and applications of thermal, optoelectronic, piezoelectric, electrochemical, bio, and gas sensors.

Principles, instrumentation, and applications of UV-visible spectrophotometry, flame photometry, potentiometry, and conductometry.

Hands-on-practice: Quantification of Bio molecules/ sodium/ potassium and iron.

Clean Energy Storage and Conversion Device Technologies

Energy storage devices: Types of electrodes, Battery components and characteristics: Primary battery (dry cell, Li-MnO_2), secondary battery (lead-acid, Li-ion).

Hands-on-practice: Assembly and Fabrication of Daniel cell. Detection of acidity, alkalinity and pH of a soil by automated electrode.

Case study: Choice of battery materials for e-vehicles

Supercapacitors: The electrochemical double Layer-Helmholtz model, charge accumulation mechanisms in EDLC, pseudo and hybrid capacitors with applications.

Photovoltaic cell: Principle, purification of Materials, working, advantages and applications.

Hands-on-practice: Fabrication of dye sensitized solar cell

Electro less plating: Principle of electro less plating, manufacture of Printed circuit board using electroless plating of copper and applications.

Hands-on-practice: Electro plating and electro less plating of copper.

Case study: Materials for printed circuit boards.

Polymeric Materials: Introduction, biodegradable and biocompatible polymeric materials. Conducting polymers-electron transport mechanism, and applications of poly acetylene and poly aniline.

Hands-on-practice: Thin film fabrication of a conducting polymer for device applications by a) electro polymerization b) chemical oxidative method.

Liquid crystals: Classification, and applications in electronic display systems.

Nanomaterials: Introduction, classification, and quantum confinement. Size-dependent properties.

Hands-on-practice: Preparation of semiconducting nanomaterials.

Smart materials for signal conversion: Solvetochromism, Mechanochromism, Piezoelectric materials, shape memory alloys, electro active materials, magnetorheological materials, photochromic materials and magnetostrictive materials.

Hands-on-practice: Study of optical band gap of quantum dot semiconducting nanoparticles by UV-Visible spectrophotometer.

Case study*:* Recent advances in nanotechnology for flexible electronic devices.

Fast Learners: 1. Average molecular weight determination by viscosity method.

2. Industrial Wastewater treatment by Photocatalysis.

3. Preparation of Polystyrene and Phenol - Formaldehyde or Urea - Formaldehyde Resin.

4. Membrane preparation for water purification.

Self-Learning Component: Chemical Structure drawing (2D &3D), energy minimization by using ISIS draw, Avogadro software, Auto Dock software, and Various types of plots using Origin software.

TEXTBOOKS:

1. R.V. Gadag and Nithyanandashetty N, "Engineering Chemistry" Ik International Publishing house, 3rd edition, 2014.
2. B. Siva Shankar, "Engineering Chemistry", Tata Mc Graw Hill Publishing Limited, 3rd Edition, 2015.
3. S. S. Dara, Mukkanti, "Text of Engineering Chemistry", S. Chand & Co, New Delhi, 12th Edition, 2006.
4. B.S. Jai Prakash, R.Venugopal, Sivakumaraiah & Pushpa Iyengar., "Chemistry for Engineering Students", Subhash Publications, Bangalore.
5. R. P. Mani, K. N. Mishra, "Chemistry of Engineering Materials", Cengage Learning, 3rd Edition, 2015.
6. P. C. Jain, Monica Jain, "Engineering Chemistry", Dhanpat Rai Publishing Company, 17th Edition, 2015.
7. D. Singh, B. Deshwal, S. . Vats, "Comprehensive Engineering Chemistry", Dreamtech Press, 2020.

REFERENCE BOOKS:

1. V.R. Gowariker, N.V. Viswanathan & J. Sreedhar., "Polymer Science", Wiley-Eastern Ltd. Callister W.D., "Materials Science and Engineering", John Wiley & Sons, 10th edition, 2013.
2. B. M Chandrashekar and B.C. Basavaraju, ''Engineering Chemistry''Basuchandras Publishing 3rd edition 2016.
3. S. Agarwal ''Engineering Chemistry'', Cambridge University Press, 2nd edition, 2019.
4. O.V. Roussak and H. D. Gesser , Applied Chemistry: A textbook for Engineers and Technologists, Springer New York, NewYork, 2nd edition, 2013.
5. V.M. Balsaraf, ''Applied Chemistry-II'', Dreamtech Press, 1st edition, 2020.
6. B. Bhardwaj ''Applied Chemistry'', S.K. Kataria & Sons, 2nd edition, 2024.

Additional Links:

1. **https://www.youtube.com/watch?v=JlZhHhpVRrI**
2. **https://www.youtube.com/watch?v=xAS4NS9RuI4**
3. **https://www.youtube.com/watch?v=KvdlWobaBkM**
4. **https://www.youtube.com/watch?v=HTFjHINBJts**
5. **https://www.youtube.com/watch?v=iFVVv23ZzGE**

Course Title	BASICS OF ELECTRICAL & ELECTRONICS ENGINEERING				Course Type		HC	
Course Code	B24EE0101	Credit	03		Class		I semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:

This course introduces the basic concepts of electrical as well as electronics engineering. This course provides the knowledge of electrical power generation through hydro, thermal , wind and solar energy sources. Provides the basic knowledge of smart grid and electric vehicle systems. It also helps the students to understand applications of semiconductor devices such as Diodes and BJT along with their characteristics and applications. This course provides better understanding of digital electronics.

COURSE OBJECTIVES:

The objectives of this course are to:

1. Explain and to make the students familiar about the basics of Electrical Circuits.
2. Illustrate the basics of magnetic circuits, DC machines, Transformers and domestic protection system
3. Illustrate the characteristics of semiconductor devices and their applications
4. To familiarize the students about digital circuits

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO-1	Explain and apply Ohm's Law, Kirchhoff's Current Law (KCL), and Kirchhoff's Voltage Law (KVL) in DC circuit analysis.	1,2	1
CO-2	Analyze RL, RC, and RLC circuits in series and parallel configurations and interpret phasor diagrams for single phase and 3-phase ac systems.	1,2	1
CO-3	Understand the working of Hydro electric power plant, thermal power plant , wind and solar power plants and grid integration of solar power plants.	1,2,7	1,3
CO-4	Familiarize with components of smart grid and electric vehicle.	1,2,7	1,3
CO-5	Describe the operation and characteristics of semiconductor diodes, Zener diodes, bipolar junction transistors (BJTs) and their applications.	1,2	1
CO-6	Apply Principles and theorems of Digital Electronics to simplify the logic expressions, converting between number systems and designing logic circuits using basic gates.	1,2	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply (L3)**	**Analyze (L4)**	**Evaluate (L5)**	**Create (L6)**
CO1			✓			
CO2				✓		
CO3		✓				
CO4		✓				
CO5		✓				
CO6			✓			

COURSE ARTICULATION MATRIX:

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3											1		
CO2	3	3											1		
CO3	3	2					2						1		1
CO4	2	2					2						1		1
CO5	3	3											1		
CO6	3	2											1		

Note:1-Low,2-Medium,3-High

COURSE CONTENTS

Introduction to Electrical engineering: Basics of DC Circuits: Ohms law, Kirchhoff's Current Law, Kirchhoff's Voltage law, Numerical examples as applicable. Basics of AC Circuits: Sinusoidal voltage and currents, Magnitude and phase, polar and rectangular representation, RL, RC and RLC series and parallel circuits, power factor, phasor diagrams. 3-phase circuits- Star and Delta connections.

Electrical Power generation: An Overview of generation, transmission and distribution system. Hydro Electric Power plant- Layout, working, factors for selection of site, classification, merits and de merits. Thermal Power Plant- Layout, working, merits and demerits, Wind power plant- General layout, working, factors for site selection, merits and demerits, Solar Power Plant- General arrangement and operation, Basic concepts of on-grid PV systems, merits and de merits.

Smart Grid : Introduction, components of smart grid, comparison between conventional and smart grid.

Electrical Vehicle: Schematic of Electric Vehicle, Classification.

Self learning component: Hybrid Electric Vehicle.

Semiconductor Diodes and Transistors: Semiconductor Diodes: P-N junction diode, V-I Characteristics, Half-wave rectifier, Full-wave rectifier, Bridge rectifier, Capacitor filter circuit, Zener diode voltage regulators, Clipping and clamping circuit, Numerical examples as applicable. Transistors: Bipolar junction Transistors BJT configuration: BJT Operation, Common Base, Common Emitter and Common Collector, Characteristics, Numerical examples as applicable.

Digital Electronics and Number System: Introduction, Switching and Logic Levels. Number Systems and its conversions: Decimal Number System, Binary Number System, Hexadecimal Number System, Binary addition, Binary subtraction. Boolean Algebra Theorems, DeMorgan's theorem. Digital Circuits: Logic gates, Algebraic Simplification, Realization of all logic and Boolean expressions. Realization of basic gates using universal gates.

Self Learning Component: Hybrid Electric Vehicle.

TEXT BOOKS:

1. Nagrath I.J. and D. P. Kothari, "Basic Electrical and Electronics Engineering", Second Edition Tata McGraw Hill, 2020.
2. Hayt and Kimberly, "Engineering Circuit Analysis", 8th Edition, Tata McGraw Hill, 2013.
3. Electrical power generation Manish Goel, Gaurav Guptha Laxmi publications Pvt Limited, First edtition,2011
3. Kulshreshtha D.C., "Basic Electrical Engineering", Second Edition, Tata McGraw Hill, 2019.
4. David A. Bell, "Electronic Devices and Circuits", Oxford University Press, 5th Edition, 2008.
5. Electric and Hybrid Electric Vehicles -Published by Pearson, 1st edition, (June 7, 2022) © 2023.
6. "Smart Grid: Technology and Applications" by Janaka Ekanayake, Nick Jenkins, Kithsiri Liyanage, Jianzhong Wu, Akihiko Yokoyama
(Wiley, 1st Edition, 2012
.

REFERENCE BOOKS:

1. . Hughes, “Electrical Technology”, International Students 9th Edition, Pearson, 2005.

2. Robert L. Boylestad and Louis Nashelsky, “Introduction to Electricity, Electronics and Electromagnetics” Prentice Hall, 5th edition.2001

3. “Electrical power generation transmission and distribution” edited by Leonard Grigby, 3rd Edition,2012

4. Textbook of Digital Electronics 0th Edition, Kindle Edition by S.S. Bhatti Rahul Malhotra, 2011

5. "Smart Grid Security: Innovative Solutions for a Modernized Grid" by Florian Skopik, Erich Wenger (Syngress, 1st Edition, 2015)

Additional material:

SWAYAM/NPTEL/MOOCS

1. https://nptel.ac.in/courses/108108076
2. https://nptel.ac.in/courses/108101091
3. https://www.udemy.com/course/basic-electrical-engineering-part-1

Course Title	Fundamentals of Data Science				Course Type		HC	
Course Code	B24CS0104	Credits	2		Class		I Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	2	2	2				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**2**	**2**	**2**	**28**	**-**	**50**	**50**

COURSE OVERVIEW:

Data Science is an interdisciplinary, problem-solving oriented subject that is used to apply scientific techniques to practical problems. The course orients on preparation of datasets and programming of data analysis tasks. This course covers the topics: Set Theory, Probability theory, Tools for data science, ML algorithms and demonstration of experiments either by using MS-Excel/Python/R.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the fundamental concepts of Excel.
2. Illustrate the use of basic concepts of Data Science in the real-world applications.
3. Demonstrate the use of SQL commands in real world applications.
4. Discuss the functional components of Data Science for real world applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Make use of the basic concepts of Data Science in developing the real-world applications.	1 to 4, 12	1,2,3
CO2	Apply the SQL commands in developing the real-world applications.	1 to 5,12	1,2,3
CO3	Build the data analytics solutions for real world problems, perform analysis, interpretation and reporting of data.	1 to 5	1, 2, 3
CO4	Demonstrate visualization of Data using python libraries	1 to 5, 8 to 10	1,2, 3
CO5	Find modeling Error in Linear Regression.	1 to 5	1, 2, 3
CO6	Use statistical principles to solve mean and standard deviations for given data.	1 to 4, 12	1,2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**Apply (L3)**	**Analyze (L4)**	**Evaluate (L5)**	**Create (L6)**
CO1			√			
CO2			√			
CO3			√	√		
CO4			√	√	√	√
CO5		√				
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	2								2	3	1	1
CO2	2	3	2	1	2	2						2	3	2	2
CO3	2	3	3	2	2								3	3	3
CO4	3	3	3	2	2			2	2	2			3	3	3
CO5	2	3	2	2	2								3	3	3
CO6	3	3	2	2								2	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Microsoft Excel: History and importance of Microsoft Excel, Creating Excel tables, understand how to Add, Subtract, Multiply, Divide in Excel. Excel Data Validation, Sorting, Filtering, Grouping, Ungrouping and Subtotal. Introduction to formulas and functions in Excel. Logical functions (operators) and conditions. Visualizing data using charts in Excel.

Import XML Data into Excel, How to Import CSV Data (Text) into Excel, How to Import MS Access Data into Excel, Working with Multiple Worksheets.

Introduction to Data Science: What is Data Science? Applications of Data Science, Data science life cycle, Tools for data science, definition of AI, types of machine learning (ML), list of ML algorithms for classification, clustering, and feature selection.

Probability theory, bayes theorem, bayes probability; Cartesian plane, equations of lines, graphs; exponents.

Introduction to SQL: SQL Commands experimental demonstrations-DDL, DML, DCL, TCL, DQL. Import SQL Database Data into Excel.

Data Relationship Methods: Introduction to Correlation, Description of linear regression and Logistic Regression, Introducing the Gaussian, Introduction to Standardization, Standard Normal Probability Distribution in Excel, Calculating Probabilities from Z-scores, Central Limit Theorem, Algebra with Gaussians, Markowitz Portfolio Optimization, Standardizing x and y Coordinates for Linear Regression, Standardization Simplifies Linear Regression, Modeling Error in Linear Regression, Information Gain from Linear Regression.

Introduction to Statistics and Python: Statistical Analysis: Descriptive statistics- Mean, Standard Deviation for Continuous Data, Frequency, Percentage for Categorical Data.

Python basics, Strings, Lists, Tuples, Sets, Dictionaries. Introduction to python libraries - Numpy, Matplotlib, Pandas, Scikit-Learn, Implementation of ML.

Self Learning Component:

1. Relational database management system.

2. Advanced MS-Excel and JMP(SAS) tool

TEXT BOOKS:

1. B.S. Grewal, “Higher Engineering Mathematics”,43rdEdition, Khanna Publishers, 2015.
2. Ramakrishnan and Gehrke, “Database Management systems”, 3rdEdition, McGraw Hill Publications, 2003.
3. “Mastering Data Analysis in Excel” - https://www.coursera.org/learn/analytics-excel.
4. Kenneth N. Berk, Carey, “Data Analysis with Microsoft Excel”, S. Chand & Company,2004.
5. Joel Grus,”Data science from scratch - First principles with Python” , OâReily, 2015.

REFERENCE BOOKS:

1. B.V. Ramana, "Higher Engineering Mathematics", 19th edition, Tata McGraw Hill Publications, 2013.
2. ErwinKreyszig, "Advanced Engineering Mathematics", 9th edition, Wiley Publications, 2013.
3. Seymour Lipschutz, John J. Schiller, "Schaum's Outline of Introduction to Probability and Statistics", McGraw Hill Professional, 1998.

JOURNALS/MAGAZINES:

1. https://www.journals.elsevier.com/computational-statistics-and-data-analysis
2. https://www.springer.com/journal/41060International Journal on Data Science and Analytics
3. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=8254253IEEE Magazine on Big data and Analytics

SWAYAMNPTEL/MOOCs

1. Excel Skills for Business: Essentials, Macquarie University (https://www.coursera.org/learn/excel- essentials)
2. SQL for Data Science, University of California, Davis(https://www.coursera.org/learn/sql-for-data-science)
3. Data Science Math Skills, Duke Universityhttps://www.edx.org/course/subject/data-science
4. https://onlinecourses.nptel.ac.in/noc19_cs60/preview

Course Title	Fundamentals of C programming				Course Type		HC	
Course Code	B24CI0109	Credits	2		Class		I Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	otal Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	1	2	2				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**2**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:

Algorithms and flowcharts are the fundamental tools for problem solving which can be used by the computers. The computer programs can be developed using algorithms and flowcharts to provide solutions to problems. C Language is a general-purpose, structured and procedure oriented programming language. It is one of the most popular computer languages today because of its structure and higher-level abstraction C. This course introduces algorithms, flowcharts and various C Programming language constructs for the development of real world applications.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Understand the basic structure and history of the C programming language, and set up an environment to execute simple programs.
2. Write programs in modules using functions and recursive calls to achieve modularity
3. Understand and utilize concepts of arrays and pointers to write efficient programs.
4. Memory management: Effectively use the memory management system to achieve minimal compilation and execution time.
5. Use algorithms and flowcharts as fundamental tools for problem-solving in C programming.

COURSE OUTCOMES(CO'S)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand the foundational concepts of the C language, including its history and basic building blocks..	1-3	1
CO2	Utilize conditional and unconditional control flow statements to write efficient programs.	1-5	1
CO3	Understand strings and library functions on strings.	1-5	1
CO4	Apply functions and recursive function calls to achieve modularity.	1-3,5	2,3
CO5	Manipulate pointers to write efficient programs.	1-5	2,3
CO6	Explore command line functionalities to control programs from command prompt.	1-5	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**Apply (L3)**	**Analyze (L4)**	**Evaluate (L5)**	**Create (L6)**
CO1				✓		
CO2			✓			
CO3			✓			
CO4						✓
CO5		✓	✓			
CO6						✓

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1	3										3		
CO2	1	3	2	2	2								3		
CO3	2	2	2		1									3	3
CO4	3	3	3	1	1									3	3
CO5	3	3	3	2	2										
CO6	3	3	3	2	2				3				3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

1. Introduction to C Programming: Overview of C language, Structure of a C program, Compilation and execution, Basic syntax and semantics, Variables and data types, Constants and literals and Input and output operations

2. Building Blocks of C: Arithmetic operators, Relational operators, Logical operators, Bitwise operators, Assignment operators, Conditional operator, Precedence and associativity of operators and Type conversion

3. Control Statements: Decision making with if and if-else, Nested if-else statements, switch case statements, looping constructs: for, while, and do-while loops, Nested loops, break and continue statements and goto statement and labels **4.Functions:** Defining and calling functions, Function prototypes, Scope and lifetime of variables, passing arguments to functions, Return values and Recursive functions

5. Arrays & Strings One-dimensional arrays, Multidimensional arrays, Array initialization and manipulation, Strings and string functions and Array of strings, matrix operations using 2-dimensional arrays.

6. Pointers: Understanding pointers, Pointer arithmetic, Pointers and arrays, Pointers and strings, Pointers to functions, Pointers and dynamic memory allocation, Introduction to structures and unions.

Self-Learning Component:

1. Evaluations of programming languages
2. Different programming languages and their applications in real world
3. Know more about most popular programming languages in the world by using : https://www.tiobe.com/tiobe-index/

TEXT BOOKS:

1. Herbert Schildt, “C: The Complete Reference”, 4th edition, TATA McGraw Hill, 2000.
2. B.W. Kernighan & D.M. Ritchie, "C Programming Language", 2nd Edition, PRENTICE HALL SOFTWARE SERIES, 2005.
3. B.S. Anami, S.A. Angadi and S. S. Manvi, “Computer Concepts and C Programming: A Holistic Approach”, second edition, PHI,2008.
4. Yashavant Kanetkar, “Let us C” , BPB , 19th edition, 2022

REFERENCE BOOKS:

1. Balaguruswamy,” Programming in ANSI C”, 4th edition, TATA MCGRAW Hill, 2008.
2. Donald Hearn, Pauline Baker,“ Computer Graphics C Version”, second edition, Pearson Education, 2004.
3. **Jon Bentley, “**Programming Pearls”, 2nd edition, PEARSON,2002.

JOURNALS/MAGAZINES:

1. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6294 (IEEE Journal/Magazine on IT Professional)
2. https://ieeexplore.ieee.org/document/1267572 (IEEE Computing in Science and Engineering)

SWAYAMNPTEL/MOOCs

1. https://online courses.nptel.ac.in/noc20_cs06/preview (Problem Solving through Programming in C)

2. https://www.edx.org/course/c-programming-getting-started (C Programming Getting started)
3. https://www.coursera.org/specializations/c-programming (Introduction to C programming)

<table>
<tr><td>Course Title</td><td colspan="4">Fundamentals and Applications of Civil Engineering</td><td colspan="2">Course Type</td><td colspan="2">HC</td></tr>
<tr><td>Course Code</td><td>B24ED0102</td><td>Credits</td><td colspan="2">2</td><td colspan="2">Class</td><td colspan="2">I Semester</td></tr>
<tr><td rowspan="5">Course Structure</td><td>LTP</td><td>Credits</td><td>Contact hours</td><td>Work load</td><td colspan="2" rowspan="2">Total Number of Classes per semester</td><td colspan="2" rowspan="2">Assessment in Weightage</td></tr>
<tr><td>Lecture</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>Tutorial</td><td>1</td><td>2</td><td>2</td><td rowspan="2">Theory</td><td rowspan="2">Practical</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td></tr>
<tr><td>Practical</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Total</td><td>2</td><td>3</td><td>3</td><td>42</td><td>-</td><td>50</td><td>50</td></tr>
</table>

COURSE OVERVIEW

This course, serves as an introductory exploration into the diverse realms of civil engineering, emphasizing the integration of software tools and applications. It covers fundamental fields of civil engineering, including infrastructure development, construction materials, traffic management, Geo Informatics and smart city solutions. The course aims to bridge traditional civil engineering practices with modern techniques, providing students with the knowledge and skills necessary to address contemporary engineering challenges.

COURSE OBJECTIVES

The objective of this course are to

1. Know the various fields and scope of Civil Engineering and the socio-economic impact of different infrastructure types.
2. To develop the ability to analyze forces, support reactions, and loads in engineering mechanics, focusing on both concurrent and non-concurrent force systems.
3. To analyze and visualize the load distribution, shear force, bending moments and deflections of statically determinate beams using tools.
4. To understand the principles of transportation engineering, classification of roads, pavements, application of Intelligent Transportation Systems (ITS) and smart cities.
5. Use of geo-informatics principles, including remote sensing, GIS, and image processing, and apply these concepts in practical scenarios through case studies.
6. Create cross-sectional views of different road types and understand their practical applications.

COURSE OUTCOMES

After completion of the course students will be able to:

CO1: Understand the scope and fields of Civil Engineering, types of infrastructure, and their socio-economic impact.

CO2: Analyze forces, support reactions, and loads in engineering mechanics.

CO3: Analyse and visualize the load distribution, shear force, bending moments and deflections of statically determinate beams using tools.

CO4: Understand transportation engineering principles, including road classification, ITS and smart cities.

CO5: Utilize geo-informatics principles for remote sensing, GIS, and image Processing.

CO6: Apply software tools and create road cross-sections and understand their practical applications.

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	✓	✓				
CO2		✓	✓	✓		
CO3		✓	✓			
CO4		✓	✓			
CO5		✓	✓	✓		
CO6		✓	✓	✓		

COURSE ARTICULATION MATRIX

CO#	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	3				3	3					3	3			
CO2	3	3		2					3	3		3	3	3	1	
CO3	3	3		3			2		3	3		3	3	3	1	
CO4	3	3				3	3			3		3	3	3	1	
CO5	3	3		3		3						3	3	3		1
CO6	3	3					3					3	3	3		1

COURSE CONTENT

Introduction to Civil Engineering - Scope of different fields of Civil Engineering. Types of infrastructure, Importance of Infrastructural Development, Effect of the infrastructural facilities and social-economic development of a country. Construction materials.

Engineering Mechanics: Forces and Classification, analysis of concurrent force systems and non-concurrent force systems-resultants. Types of loads, supports and beams. Numerical problems on beams: Calculation of support reactions with point load , uniformly distributed load and moments.

- Load applications and analysis: To visualize the load distribution, shear force, bending moments and deflections of statically determinate beams using Staad Pro/MS Excel.

Transportation Engineering: Classification of Roads and their functions, Flexible and Rigid Pavements. Road safety and management, Causes - Collection of data for traffic studies, Traffic management techniques, ITS and its application. **Smart cities:** Smart energy resources, Smart disposal of solid waste, smart road construction.

- Create cross-sectional views of different road types using AutoCAD software

Geo Informatics: Principles of remote sensing - Satellites and Sensors, Introduction to GIS principles, Global positioning system, Introduction to image processing and its application. Case studies.

TEXT BOOKS

1. Elements of Civil Engineering Jagadeesh, T. R., & Jayaram, M. A. Sapna Book House. ISBN: 978-8120341675
2. Engineering Mechanics. Khurmi, R. S. S. Chand & Company Ltd. ISBN: 978-8121901014
3. Environmental Engineering. Rao, P. V. (2002)Prentice-Hall of India Pvt. Ltd. ISBN: 978-8123919352
4. Remote Sensing and GIS Bhatta, B. Oxford University Press. ISBN: 978-0198072393
5. Traffic Engineering and Transport Planning. Kadiyali, L. R. Khanna Publishers. ISBN: 978-8185596780
6. Building Construction. Punmia, B. C., Jain, A. K., & Jain, A. K. Laxmi Publications. ISBN: 978-8131804289

REFERENCE BOOKS

1. Structural Health Monitoring. Balageas, D., Fritzen, C. P., & Güemes, A. John Wiley & Sons. ISBN: 978-0470394381
2. Principles of Geotechnical Engineering. Das, B. M. Cengage Learning. ISBN: 978-0495668107
3. Geographic Information Systems and Science. Longley, P. A., Goodchild, M. F., Maguire, D. J., & Rhind, D. W. John Wiley & Sons. ISBN: 978-0470721446
4. Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia. Townsend, A. M. W.W. Norton & Company. ISBN: 978-0393349786
5. Construction Project Management: A Practical Guide to Field Construction Management (. Sears, S. K., Sears, G. A., & Clough, R. H. John Wiley & Sons. ISBN: 978-1118745052

Course Title	Fundamentals of Data Science Lab				Course Type		HC	
Course Code	B24CS0108	Credits	1		Class		I Semester	
Course Structure	LTP	Credits	Contact Hours	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	**1**	**2**	**2**	**0**	**28**	**50**	**50**

COURSE OVERVIEW:

Data Science is an interdisciplinary, problem-solving oriented subject that is used to apply scientific techniques to practical problems. The course orients on preparation of datasets and programming of data analysis tasks. This course covers the topics: ML algorithms, SQL and demonstration of experiments by using MS-Excel and MySQL and Python.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the fundamental concepts of Excel.
2. Explain the algorithms of Machine learning.
3. Demonstrate the use of SQL commands in real world applications.
4. Discuss the functional components of Data Science for real world applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Make use of the concepts of Microsoft Excel in developing the real-world applications.	1 to 5, 8 to 10	1,2,3
CO2	Apply the SQL Queries in developing the real-world applications.	1 to 5, 8 to 10	2, 3
CO3	Build the solutions for real world problems, perform analysis, interpretation and reporting of data using regression algorithms.	1 to 5, 8 to 10	1, 2, 3
C04	Design ER diagrams for database.	1 to 5, 8 to 10	1, 2, 3
CO5	Use Excel to solve Multiple Linear Regression.	1 to 5, 8 to 10	1, 2, 3
CO6	Demonstrate visualization of Data using python libraries	1 to 5, 8 to 10	1,2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1			✓			
CO2			✓			
CO3			✓	✓		
C04			✓	✓	✓	✓
CO5		✓				
CO6			✓			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2			1	3	3			3	3	3
CO2	2	2	2	2	2			1	3	3			3	3	3
CO3	3	3	2	2	2			1	3	3			3	3	3
CO4	3	3	3	2	2			1	3	3			3	3	3
CO5	3	3	3	2	2			1	3	3			3	3	3
CO6	3	3	3	2	2			1	3	3			3	3	3

Practice:

No	Title of the Experiment	Tools and Technics	Expected Skill/Ability
1	The height (in cm) of a group of fathers and sons are given below, Findthe lines of regression and estimate the height of son when the height of father is 164 cm. **Hgt of Father**: 158, 166, 163, 165, 167, 170, 167, 172, 177, 181 **Hgt of Son**: 163, 158, 167, 170, 160, 180, 170, 175, 172, 175	MS Excel And JMP Tool	Create and perform operations on Excel data set by applying Linear regression
2	Using the data file DISPOSABLE INCOME AND VEHICLE SALES,perform the following: **i)** Plot a scatter diagram. **ii)** Determine the regression equation. **iii)** Plot the regression line (hint: use MS Excel's Add Trend line feature. **iv)** Compute the predicted vehicle sales for disposable income of $16,500 and of $17,900. **v)** Compute the coefficient of determination and the coefficient of correlation	MS Excel And JMP Tool	Perform prediction and visualiza - tion of data
3	Managers model costs in order to make predictions. The cost data in the data file INDIRECT COSTS AND MACHINE HOURS show the indirect manufacturing costs of an ice- skate manufacturer. Indirect manufacturing costs include maintenance costs and setup costs. Indirect manufacturing costs depend on the number of hours the machines are used, called machine hours. Based on the data for January to December, perform the following operations. **i)** Plot a scatter diagram. **ii)** Determine the regression equation. **iii)** Plot the regression line (hint: use MS Excel's Add Trend line feature).	MS Excel And JMP Tool	Perform prediction and visualization of data

<table>
<tr><td></td><td>iv) Compute the predicted indirect manufacturing costs for 300 machine hours and for 430 machine hours.

v) Compute the coefficient of determination and the coefficient of correlation.</td><td></td><td></td></tr>
<tr><td>4</td><td>Apply multiple linear regression to predict the stock index price which is a dependent variable of a fictitious economy based on two independent / input variables interest rate and unemployment rate.
<table><tr><th>Year</th><th>Month</th><th>Interest rate</th><th>Un employment rate</th><th>Stock index price</th></tr><tr><td>2022</td><td>10</td><td>2.75</td><td>5.3</td><td>1464</td></tr></table></td><td>MS Excel And JMP Tool</td><td>Perform prediction and visualization of data</td></tr>
<tr><td>5.</td><td>Calculate the total interest paid on a car loan which has been availed from HDFC bank. For example, Rs.10,00,000 has been borrowed from a bank with annual interest rate of 5.2% and the customer needs to pay every month as shown in table below. Calculate the total interest rate paid for availed of Rs.10, 00,000during 3 years.
<table><tr><th>Sl.no</th><th>A</th><th>B</th></tr><tr><td>1</td><td>Principal</td><td>Rs.10,00,000</td></tr><tr><td>2</td><td>Annual interest rate</td><td>5.2%</td></tr><tr><td>3</td><td>Year of the loan</td><td>3</td></tr><tr><td>4</td><td>Starting payment number</td><td>1</td></tr><tr><td>5</td><td>Ending payment number</td><td>36</td></tr><tr><td>6</td><td>Total interest paid during period</td><td>?</td></tr></table></td><td>MS Excel And JMP Tool</td><td>Create Excel data and perform EMI estimator</td></tr>
<tr><td>6</td><td>Create a supplier database of 10 records with SUPPLIER_ID as primary key, SUPPLIER_NAME, PRODUCTS, QUANTITY, ADDRESS, CITY, PHONE_NO and PINCODE, Where SUPPLIER_NAME, PRODUCTS, QUANTITY and PHONE_NO should not be NULL.</td><td>SQL</td><td>Creating Tables</td></tr>
<tr><td>7</td><td>Create the customer database of a big Market with CUSTOMER_ ID as primary key, CUSTOMER_NAME, PHONE_NO, EMAIL_ID, ADDRESS, CITY and PIN_ CODE. Store at least twenty customer’s details where CUSTOMER_NAME and PHONE_NO are mandatory and display the customer data in alphabetical order.</td><td>SQL</td><td>Creating and retrieving Tables</td></tr>
<tr><td>8</td><td>Apply the linear regression, compare the average salaries of batsman depending on the run rate scored/ recorded in the matches. Assume your own database.</td><td>MS Excel And JMP Tool</td><td>Apply Linear regression</td></tr>
<tr><td>9</td><td>Apply Multiple linear regression to predict the factory products which is A, B and C are independent variables and cost dependent variable.</td><td>MS Excel And JMP Tool</td><td>Apply Linear regression</td></tr>
<tr><td>10</td><td>Logistic Regression-case study using Kaggle Database</td><td>MS Excel And JMP Tool</td><td>Apply Logistic regression</td></tr>
<tr><td>11</td><td>Design the ER diagram and create schema of the REVA library Management system.</td><td>Entity Relationship</td><td>Entity Relationship</td></tr>
</table>

12	Perform Exploratory Data Analysis to predict customer churn in telecommunications company using Python libraries such as Pandas, and Matplotlib to aid in this process. (Use datasets from Kaggle/NCBI.)	Jupyter/Colab - Python	Apply Exploratory Data Analysis
13	"LIST1" is a list that contains "N" different SRN of students read using a user defined function with the help of input() function. It is required to add SRN of "M" more students that are to be appended or inserted into "LIST1" at the appropriate place. The program must return the index of the SRN entered by user.	Jupyter/Colab - Python	Create and perform operations on list.
14	"TUPLE1" and "TUPLE2" are two tuples that contain "N" values of different data types read using the user defined function "READ" with the help of input() function. Elements of "TUPLE1" and "TUPLE2" are to be read one at a time and the "larger" value among them should be placed into "TUPLE3". Display all tuples.	Jupyter/Colab - Python	Create and perform operations on Tuples.

PART_B:Projects

No	Title of the Experiment	Tools and Technics	Expected Skill/Ability
1	Big Mart sales forecasting	MS Excel	Apply Linearregression
2	Bangalore crime analysis	MS Excel	Apply Linearregression

TEXT BOOKS:

1. B.S. Grewal, "Higher Engineering Mathematics", 43rd edition, Khanna Publishers, 2015.
2. Ramakrishnan and Gehrke, "Database Management systems", 3rd Edition, McGraw Hill Publications, 2003.
3. "Mastering Data Analysis in Excel" - https://www.coursera.org/learn/analytics-excel.
4. Kenneth N. Berk, Carey, "Data Analysis with Microsoft Excel", S. Chand & Company,2004.

REFERENCE BOOKS:

1. B.V. Ramana, "Higher Engineering Mathematics", 19th edition, Tata McGraw Hill Publications, 2013.
2. ErwinKreyszig, "Advanced Engineering Mathematics", 9th edition, Wiley Publications, 2013.
3. Seymour Lipschutz, John J. Schiller, "Schaum's Outline of Introduction to Probability and Statistics", McGraw Hill Professional, 1998.

JOURNALS/MAGAZINES:

1. https://www.journals.elsevier.com/computational-statistics-and-data-analysis
2. https://www.springer.com/journal/41060 International Journal on Data Science and Analytics
3. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=8254253 IEEE Magazine on Big data and Analytics

Course Title	BASICS OF ELECTRICAL & ELECTRONICS ENGINEERING LAB				Course Type		HC	
Course Code	B24EE0102	Credit	1		Class		I Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2	-	28	50	50

COURSE OVERVIEW:

Basic Electrical & Electronics Engineering lab covers the concept of various types of electrical apparatus, tools, and conduction of experiments to Analyse, Design of KCL & KVL, two-way switch or staircase wiring, Determination of VI characteristics Zener Diode, Silicon Diode, Half Wave rectifier using Diode, study& analyses of Lead & lag component, verification of logic gates.

COURSE OBJECTIVES

The objectives of this course are to:

1. Establish a broad concept of various types of electrical apparatus, tools and instrumentation.
2. provide hands on experience with electrical apparatus and electrical safety norms.
3. Train students to read and understand schematics so as to make electrical connection for different appliances.
4. Train students in collecting and interpreting experimental data.
5. Enhance written skills of students.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Use appropriate electrical tools for electrical connections and repair of equipment's	1,2,4,5	1
CO2	Recognize various symbols in a schematic and make connection as per circuit diagram	1,2,9,10	1
CO3	List out various safety procedures	4,5,9,10	1
CO4	Make use of various measuring instruments to collect experimental data	2,4,9,10	1
CO5	Analyse the results obtained from experiments.	2,3,9,10	1
CO6	Demonstrate the ability to critically evaluate the performance of electrical appliances.	1,2,9,10	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			✓			
CO2		✓				
CO3	✓					
CO4			✓			
CO5				✓		
CO6					✓	

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3		3	3								1		
CO2	3	3											1		
CO3				3	3								1		
CO4		3		3									1		
CO5		3	2												
CO6	3	2													

Note:1-Low,2-Medium,3-High

COURSE CONTENT

List of Experiments:

S. No.	Name of the Practice Session
1	To verify KCL and KVL for the given circuit both practically and theoretically.
2	Simulation of RL and RC circuit using PSIM and power factor Calculation.
3	Staircase wiring circuit layout for multi storage building
4	To study and verify series and parallel circuits.
5	Study and analysis of Voltage regulator using Zener Diode
6	Demonstration of Clamper circuit with reference voltage
7	Design half wave, Full wave-centre tap and Bridge rectifier with and without capacitive filter and measure efficiency and ripple factor.
8	Demonstration of Clippers with reference voltages
9	Study and analysis of input output Characteristics of CE configuration of BJT.
10	Realization of basic gates using universal gates

Demo Experiments:

1. To Study the importance of Earthing during accidental shorting of line wire and the body of equipment.
2. To study the Importance and mechanism of MCB.

Textbooks

1. Nagrath I.J. and D. P. Kothari, “Basic Electrical Engineering”, Third Edition Tata McGraw Hill, 2009.
2. Hayt and Kimberly, “Engineering Circuit Analysis”, 8th Edition, Tata McGraw Hill, 2013.
3. Kulshreshtha D.C., “Basic Electrical Engineering”, Tata McGraw Hill, 2009.
4. Rajendra Prasad, “Fundamentals of Electrical Engineering”, Prentice Hall, India, 2009.

Reference Books

1. Theodore Wildi, “Electrical Machines, Drives, and Power, 5thSystems”, Pearson Edition, 2007.

2. Hughes, “Electrical Technology”, International Students 9th Edition, Pearson, 2005

SWAYAM/NPTEL/MOOCs

1. https://nptel.ac.in/courses/108108076
2. https://nptel.ac.in/courses/108101091
3. https://www.udemy.com/course/basic-electrical-engineering-part-1

Course Title	Computer Aided Engineering Drawing				Course Type		HC	
Course Code	B24ME0101	Credit	2		Class		I Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	-	0	0	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	**2**	**3**	**3**	**14**	**28**	**50**	**50**

COURSE OVERVIEW:

Engineering Graphics or Drawing is known as language of engineers. All phases of engineering process require the conversion of new ideas and design concepts into the basic line language of graphics. There are many areas such as civil, mechanical, electrical, architectural, computer, electronics and industrial applications where knowledge and skills of the drawing play major roles in the design and development of new products or construction. This course emphasizes on orthographic projection of point, line, plane surfaces and solids. It also provides knowledge about representing the object in terms of 3D view and also development of the objects.

COURSE OBJECTIVES:

The objectives of this course are to:

1. Introduce the concepts like dimensioning, conventions and standards of engineering drawings in order to become professionally efficient.
2. Enable students to learn about the software tool for preparing engineering drawings
3. Teach the concepts and principles of orthographic projections, development of lateral surfaces and isometric projection of simple solids.
4. Communicate the concepts/ideas through the language of technical drawing and sketching.

COURSE OUTCOMES (COs)

On successful completion of this course, the student shall be able to:

CO	Course Outcomes	POs	PSOs
CO1	Apply orthographic projection principles to represent points accurately in multiple views with or without aid of CAD software, ensuring consistency and clarity in engineering drawings.	1,2,5,10	1
CO2	Apply orthographic projection principles to draw the projection of line accurately with or without aid of CAD software, ensuring consistency and clarity in engineering drawings.		
CO3	Develop a thorough understanding of the principles of projection, including how plane surfaces are projected onto projection planes in orthographic projection with or without aid of CAD software	1,2,5, 10	1
CO4	Learn how to visualize the shape, size, and features of solids in three dimensions and accurately represent them in orthographic projection drawings with or without aid of CAD software	1,2,5, 10	1
CO5	Develop a solid conceptual understanding of the development of lateral surfaces of solids such as prisms, pyramids, cylinders and cones with aid of computer, including the principles of unfolding or unwrapping the surfaces.	1,2,5, 10	1
CO6	Create isometric drawing of solids and combined solids and learn how to visualize and represent the individual components of the combined solid and assemble them into a coherent isometric projection using CAD software.	1,2, 3,5,10	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX:

CO / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1			2					3			3		
CO2	3	2			2					3			3		
CO3	3	2			2					3			3		
CO4	3	2			2					3			3		
CO5	3	2	2		2					3			3		
CO6	3	1	2		2					3			3		

Note:1-Low,2-Medium,3-High

COURSE CONTENTS

Introduction – Geometrical constructions, engineering drawing standards, Introduction to CAD Software, Orthographic projection of points in first and third Quadrant only.
Orthographic projection of straight lines inclined to both horizontal and vertical planes.
Orthographic projection of regular plane surfaces -surface is inclined to both HP and VP.
Projections of Solids: Orthographic projection of regular solids like prisms, pyramids, cone and cylinder when the axis is inclined to both HP and VP.
Development of Lateral Surfaces of Solids: Parallel line method for prisms and cylinders, Radial line method for pyramids and cones
Isometric Drawing: Concept of isometric drawing and isometric projection, Isometric drawing of simple solid and combined solids (only 2 Solids).

PRACTICE:

Sl. No	Practice	Tools and Techniques	Expected Skill /Ability
1.	Use of solid edge software and familiarization of tools	Solid Edge Software	Use of commands to draw the drawings
2.	Draw the projection of point locating in first and third quadrant	Solid Edge Software	Analysing and software skill
3.	Draw the projection of straight lines locating in first quadrant	Solid Edge Software	Draw the views of the line and software skill
4.	Draw the projection of triangular and square lamina inclined to both HP and VP	Solid Edge Software	Analysing and software skill
5.	Draw the projection of rectangular and pentagonal lamina inclined to both HP and VP	Solid Edge Software	Analysing and software skill
6.	Draw the projection of hexagonal and circular lamina inclined to both HP and VP	Solid Edge Software	Analysing and software skill

Sl. No	Practice	Tools and Techniques	Expected Skill /Ability
7.	Draw the projection of prisms inclined to both HP and VP	Solid Edge Software	Interpretation and software skill
8.	Draw the projection of pyramids inclined to both HP and VP	Solid Edge Software	Analysing and software skill
9	Drawing development diagram of lateral surface of prisms.	Solid Edge Software	Analysing and Software Skill
10	Drawing development diagram of lateral surface of pyramids	Solid Edge Software	Analysing and Software Skill
11	Draw the isometric drawing of two co-axial solids	Solid Edge Software	Creative and Software Skill
12	Draw the isometric drawing of two co-axial solids	Solid Edge Software	Creative and Software Skill
13	Draw the isometric drawing of two co-axial solids	Solid Edge Software	Creative and Software Skill

TEXT BOOKS:

1. K S Narayanswamy and Mahesh L, "Engineering Drawing", WILEY Publishers, 1st Edition, 2017.

2. K. R. Gopalakrishna and Dr. M S Reddy, "Engineering Graphics-1", Subhas Publications, 2015.

3. Bhatt N.D., Panchal V.M and Ingle P.R, "Engineering Drawing", Charotar Publishing House Pvt. Ltd, 53rd Edition, 2019.

REFERENCE BOOKS:

1. Luzadder and Duff, "Fundamental of Engineering Drawing", Printice Hall of India Pvt. Ltd. 11th Edition, 2001.

2. Shah, M.B. and Rana B.C., "Engineering Drawing and Computer Graphics", Pearson Education,
2008. SWAYAM/NPTEL/MOOCS

Additional material:

1. https://nptel.ac.in/courses/112/103/112103019/
2. https://www.udemy.com/course/ed/

Course Title	Fundamentals of C programming lab				Course Type		HC	
Course Code	B24CI0110	Credits	1		Class		I Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	IA	SEE
	Practice	1	2	2				
	Total	1	2	2	-	28	**50**	**50**

COURSE OVERVIEW:

Pseudocode, Algorithms and flowcharts are the fundamental tools for problem solving which can be used by the computers. The computer programs can be developed using algorithms and flowcharts to provide solutions to problems. C Language is a general-purpose, structured and procedure-oriented programming language. It is one of the most popular computer languages today because of its structure and higher-level abstraction C. This course introduces algorithms, flowcharts and various C Programming language constructs for the development of real-world applications.

COURSE OBJECTIVE (S):

1. Explain algorithms, flowcharts and different programming constructs of C to be used for Development of applications.
2. Illustrate the use of iterative statements and conditional Statements for solving the real-world problems.
3. Demonstrate the use of functions with parameter passing mechanisms for solving the real-world problems.
4. Discuss the use of structures, unions, pointers and file operations for solving the real-world Problems.
5. Learn new algorithms and technologies in C Programming and apply for suitable application development.
6. Develop solutions by using C Programming to the complex problems, either individually or as a part of team and report the results.

COURSE OUTCOMES (COs)

On successful completion of this course; the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify the programming constructs of C language to solve a given problem.	1-3	1
CO2	Apply the concepts of matrices to develop data processing and analysis solutions in various application domains.	1-5	1
CO3	Develop text processing-based applications using string operations.	1-3,5	2,3
CO4	Create solutions for real world problems using Pointers, Union, Structures and file operations.	1-5	2,3
CO5	Use algorithms and technologies in C Programming for suitable application development	1-5	2,3
CO6	Develop solutions by using C Programming to the complex problems, either individually or as a part of the team and report the results	1-5,9	2,3

BLOOM'S LEVELOF THECOURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1				✓		
CO2			✓			
CO3			✓			
CO4						✓
CO5		✓	✓			
CO6						✓

COURSE ARTICULATIONMATRIX

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1	3						3	3			3		
CO2	1	3	2	2	2				3	3			3		
CO3	2	2	2		1				3	3				3	3
CO4	3	3	3	1	1				3	3				3	3
CO5	3	3	3	2	2				3	3					
CO6	3	3	3	2	2				3	3			3	3	2

Note:1-Low,2-Medium,3-High

No	Title of the Experiment
1	Write a C program using switch statement to design a basic calculator that performs the basic operations such as addition, subtraction, multiplication, and division.
2	A quadratic equation as an equation of degree 2, meaning that the highest exponent of this function is 2. Write a C program to find the coefficients of a quadratic equation and compute its roots.
3	Develop a C Program to find the largest among two and three different numbers entered by the user.
4	A leap year is a calendar year that contains an additional day compared to a common year. Design and develop C program to check whether the given year is leap year (or) not.
5	A palindrome is a word, sentence, verse, or even number that reads the same backward or forward. Write a C program to check if a given number is a Palindrome.
6	Design and Develop a C Program to print the following pattern using Looping concepts. A B C D E F A B C D E A B C D A B C A B A *
7	Consider student's marks in Computer Test. Write a C Program to display the grade obtained by a student in Computer Test based on range of marks.

8	If a four-digit number is input through the keyboard, write a program to obtain the sum of the first and last digit of this number.
9	In computer-based applications, matrices play a vital role in the projection of three-dimensional image into a two-dimensional screen, creating the realistic seeming motions. Write a C program using 2-dimensional array to check for compatibility of two matrices and perform matrix Multiplication.
10	Write a C program to implement bubble sort with appropriate input and output.
11	Write a C program to implement binary search with appropriate input and output.
12	Develop a program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of N real numbers.

TEXT BOOKS:

1. B.W. Kernighan & D.M. Ritchie, "C Programming Language", 2nd Edition, PRENTICE HALL
2. SOFTWARE SERIES, 2005.
3. Herbert Schildt, “C: The Complete Reference”, 4th edition, TATA McGraw Hill, 2000.
4. B.S. Anami, S.A. Angadi and S. S. Manvi, “Computer Concepts and C Programming: A
5. Holistic Approach”, second edition, PHI,2008.

REFERENCE BOOKS:

1. Balaguruswamy,” Programming in ANSI C”, 4th edition, TATA MCGRAW Hill, 2008.

Donald Hearn, Pauline Baker,“ Computer Graphics C Version”, second edition, Pearson Education, 2004.

Course title	Finance and Management				Course type		FC	
Course code	B24EN0102	Credits	1		Class		I Semester	
	LTP	Credits	Contact hours	Work load	Total Number of classes per semester		Assessment in weightage	
	Lecture	1	1	1				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**1**	**1**	**1**	**14**	**-**	**50**	**50**

COURSE OVERVIEW

This course provides an introduction to the fundamentals of finance and management principles essential for understanding business operations and decision-making processes. It covers key concepts in finance such as the time value of money, financial markets, and institutions, along with principles of management including planning, organizing, leading, and controlling.

COURSE OBJECTIVES

1. Understand the concept of finance and its various applications in both personal and business contexts.
2. Evaluate various financial instruments, markets, and investment opportunities to make informed investment decisions.
3. Develop the ability to analyze financial situations using time value of money principles.
4. Apply management theories and techniques to effectively lead teams, manage conflicts, and enhance organizational performance.
5. Examine Henri Fayol's fourteen principles of management and their relevance to contemporary organizational practices.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course outcomes	POs	PSo
C01	Understand how financial principles influence everyday choices, such as savings, investment, and expenditure.	9,10,11,12	1
C02	Understand the role of financial intermediaries such as banks, investment banks, and stock exchanges.	9,10,11,12	1
CO3	Understand Frederick Taylor's scientific management theory and its emphasis on efficiency and productivity.	9,10,11,12	1
CO4	Identify the functions of management: planning, organizing, leading, and controlling.	9,10,11,12	1
C05	Understand theories of motivation and their application to employee behavior.	9,10,11,12	1
CO6	Develop effective communication skills and decision-making abilities essential for managerial roles, including interpersonal communication, conflict resolution, and problem-solving.	9,10,11,12	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
C01	✓	✓	✓			
C02	✓	✓	✓			
CO3	✓	✓	✓			
CO4	✓	✓	✓			
CO5	✓	✓	✓			
CO6	✓	✓	✓			

COURSE ARTICULATION MATRIX

CO#	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	P012	PS01	PSO2	PS03
CO1									1	1	1	1	1		
C02									1	1	1	1	1		
CO3									1	1	1	1	1		
CO4									1	1	1	1	1		
C05									1	1	1	1	1		
C06									1	1	1	1	1		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENTS:

Introduction to Finance

Overview of Finance Definition and Scope of Finance Importance of Finance in Business and Personal Decision Making Goals of Financial Management Financial Markets and Institutions: Personal Financial Planning. Reading Understanding financial statements. Time Value of Money Concepts: Future Value, Present Value, Discounting Calculating Future Value and Present Value of Cash Flows Applications of Time Value of Money in Financial Decision Making Financial Markets and Institutions Types of Financial Markets: Money Market, Capital Market, Primary Market, Secondary Market Financial Instruments: Stocks, Bonds, Derivatives Role of Central Banks and Regulatory Bodies in Financial markets Principles of management.

Principles of Management

Management: Introduction- Meaning, nature and characteristics of management Definition of management Evolution of management thought Scope and Functions of management (Planning, Organizing, Leading, Controlling). Management as a science art or profession. Management and administration. Levels of management (Top, Middle, Front-line). Principles of management- Fayols principles of management. Taylor's Scientific Management Theory- Taylor's Principles of Management, Maslow's theory of Hierarchy of Human Needs, Douglas McGregor's Theory X and Theory Y- Hertzberg Two Factor Theory of Motivation - Leadership Styles, Social responsibilities of Management..Porter's Five Force Model, BCG Matrix and PESTLE analysis.

Self Learning Component:

1. Study Future Value, Present Value, and Discounting concepts through online tutorials or videos.
2. Investigate examples of companies demonstrating social responsibility and their impact on stakeholders.
3. Explore how each financial instrument is traded and valued in the market.
4. Reflect on personal leadership qualities and areas for development.

TEXT BOOKS:

1. "TAXMANN's Financial Management" by Ravi M. Kishore(2020) - Publisher: Taxmann Publications Pvt. Ltd. ISBN: 978-9389702385
2. Management Appannaiah, H. R. & Reddy, P. Second Revised Edition Himalaya Publishing House.

REFERENCE BOOKS

1. "Financial Management" Prasanna Chandra –Publisher: McGraw-Hill Education.(2020)ISBN-13: 978-9389143895 Management Richard L. Daft Publisher: Cengage Learning.

<table>
<tr><td>Course Title</td><td colspan="3">Foundation for Learning</td><td colspan="2">Course Type</td><td colspan="2">MC</td></tr>
<tr><td rowspan="6">Course Code</td><td>B24CS0109</td><td>Credits</td><td colspan="2">0</td><td colspan="2">Class</td><td colspan="2">I Semester</td></tr>
<tr><td>LTP</td><td>Credits</td><td>Contact Hours</td><td>Work Load</td><td colspan="2" rowspan="2">Total Number of Classes Per Semester</td><td colspan="2" rowspan="2">Assessment in Weightage</td></tr>
<tr><td>Lecture</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>Tutorial</td><td>-</td><td>-</td><td>-</td><td rowspan="2">Theory</td><td rowspan="2">Practical</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td></tr>
<tr><td>Practice</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Total</td><td>0</td><td>1</td><td>1</td><td>14</td><td>-</td><td>25</td><td>25</td></tr>
</table>

COURSE OVERVIEW

This course was designed to provide students with hands-on experience and an in-dept understanding of fundamental key concepts considered crucial for personal and academic success. Students will engage in activities to explore, measure and learn about various evidence-based strategies or techniques that enhance their skills. Students will investigate and assess self-motivation, learn and use self-regulation skills, and develop resilience. They will also learn about self-efficacy, its assessment, and how to increase it. The practical's involve comprehending learning concepts, evaluating learning techniques, and creating successful learning strategies. Students will also be enabled to take charge of their personal and academic life via goal setting, planning, and decision-making activities. These goals seek to offer students a thorough grasp and practical application of key psychological principles, hence promoting personal growth and academic achievement.

Pre-requisites

1. Willingness to examine personal strengths, weakness and behaviors.

Pedagogy

This course will be run at two levels of differentiated learning. Basic course will involve a series of sessions in auditorium where students will complete self-assessment for a given skill, attend a talk and then write a quiz. More engaged students will complete a series of home assignments that help them improve their skills with support from Class Mentors. Students will work in teams supporting each other to complete the activities and compete with other teams on basic student engagement related parameters.

The pedagogy includes both Teacher-Centered and Learner Centered Approaches. Such as:

1. ICT Based learning
2. Direct Instruction learning
3. Flipped Classroom
4. Inquiry based learning
5. Kinesthetics Learning
6. Expeditionary Learning

COURSE OBJECTIVES

1. To explore **goal-setting, planning, decision making** and empower students to take control of their personal and academic functioning.
2. To understand the **principles of learning,** evaluate the effectiveness of different learning approaches and develop strategies for enhancing learning skills.
3. To understand, measure and explore techniques to enhance **self-motivation**.
4. To define **self-regulation**, assess and identify and practice self-regulation techniquesto improve focus and productivity.
5. To explore the concept of **resilience**, evaluate and learn methods for cultivatingresilience.
6. To comprehend the concept of **self-efficacy**, measure and learn to build and self-efficacy.

COURSE OUTCOMES

1. Students will explore goal setting and decision-making, be able to measure these skills, and adopt a proactive approach to managing their future, leading to improved overall functioning and success.
2. Students will grasp the principles of effective learning, be able to evaluate their learning approaches, and identify techniques to strengthen their learning skills.

3. Students will gain a thorough understanding of self-motivation, be able to measure their levels of self-motivation, and identify various techniques to enhance it.
4. Students will be able to define and assess their self-regulation skills and identify techniques to improve them.
5. Students will explore the concept of resilience, be able to measure their resilience, and identify techniques to enhance it.
6. Students will understand the concept of self-efficacy, be able to measure their self- efficacy, and identify techniques to enhance it.

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1		√	√			
CO2		√	√			
CO3		√	√			
CO4		√	√			
CO5		√	√			
CO6		√	√			

COURSE ARTICULATION MATRIX

Note:1-Low,2-Medium,3-High

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	2					1	3	3	2	2	0	0	0
CO2	3	3	3	2				2	3	3	2	3	0	0	0
CO3	2	2	2					1	2	2	2	2	0	0	0
CO4	3	3	2					2	3	3	2	3	0	0	0
CO5	2	2	2					1	2	2	2	2	0	0	0
CO6	3	3	3	2				2	3	3	2	3	0	0	0

COURSE CONTENT:

1. Taking charge of one's destiny: Introduction to Decision Making and Goal Setting, Assessment, Enhancement Techniques.
2. Learning to learn: Definition, Learning Types, Theories, Assessment, Enhancement Techniques.
3. Motivation: Definition, Types, Theories, Assessment and Interpretation, Consequences, and Techniques for Improvement.
4. Self-regulation: Definition, Assessment and Interpretation, Consequences, Enhancement Techniques.

5. Resilience: Definition, Assessment and Interpretation, Consequences, Techniques for Improvement.

6. Self-efficacy: Definition, Assessment and Interpretation, Consequences, and Techniques to Increase Self-efficacy.

TEXTBOOKS:

Worksheets containing informative material tailored to this course will be provided to the students in the classroom.

REFERENCE:

1. Snyder, C.R, Lopez, Shane. J & Pedrotti, Jennifer. Teramoto, *Positive Psychology*, Sage Publications India Pvt Ltd.,(2nd edition), 2011.

2. Branscombe, N R and Baron, R A. *Social Psychology*, Pearson India Education Services Pvt. Ltd.,14thedition, 2018.

3. UNICEF Azerbaijan, *Basic Life Skills Course Facilitator's Manual*, United Nations Children's Fund (UNICEF), Baku, Azerbaijan, 2011. [Online]. Available: https://www.unicef.org/azerbaijan/reports/basic-life-skills-course-facilitators-manual. [Accessed: 26-Jul-2024].

4. AIF, *Handbook on Activities for Life Skills*, American India Foundation (AIF), 2012. [Online]. Available:https://aif.org/wp-content/uploads/2017/04/LAMP-Life-Skills-Activity-Handbook.pdf

5. M0063_Life_Skills_Cover.indd (peacecorps.gov)

   SP_FacilitatorsManual_Life-skills_Aug2014.pdf (edu.na)

II Semester												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS0203	Probability and Statistics	FC	3	0	0	3	3	50	50	100	BSC
2	B24AS0202	Physics for Computer Science	FC	3	0	0	3	3	50	50	100	BSC
3	B24ME0105	Fundamentals of Mechanical Engineering	HC	3	0	0	3	3	50	50	100	ESC
4	B24CI0201	Advanced C programming and Applications	HC	2	0	0	2	3	50	50	100	ESC
5	B24EN0101	Internet of Things	HC	1	0	1	2	3	50	50	100	ESC
6	B24AS0208	Physics for Computer Science Lab	FC	0	0	1	1	2	25	25	50	ESC
7	B24CI0202	Advanced C programming and Applications Lab	HC	0	0	1	1	2	25	25	50	ESC
8	B24ME0102	Innovation & Entrepreneurship	FC	1	0	1	2	3	50	50	100	HSMC
9	B24AH0103	Communicative English	FC	0	0	1	1	2	25	25	50	HSCMC
10	B24CSET01	Engineering Exploration	FC	1	0	0	1	1	25	25	50	FC
11	B24CSET02	Emerging Technologies on Full Stack Web Development	ETC	1	0	0	1	1	25	25	50	ETC
TOTAL				15	0	5	20	26	425	425	850	
TOTAL SEMESTER CREDITS				20								
TOTAL CUMULATIVE CREDITS				41								
TOTAL CONTACT HOURS				26								
TOTAL MARKS				850								

Title	Probability and Statistics				Course Type		FC	
Course Code	B24AS0203	Credits	3		Class		II semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

The course Probability and Statistics for Computer Science treats the most common discrete and continuous distributions, showing how they find use in decision and estimation problems, and constructs computer algorithms for generating observations from the various distributions. Probability in the design and analysis of randomized algorithms. Common randomized algorithms are things like Quicksort and Quickselect. Probabilistic method can also useful to prove various important results. Probabilistic methods used to prove some partition theorems that were then used to create efficient data structure.

COURSE OBJECTIVE (S):

1. Describe Curve fitting and regression in various problems in Computer Science and engineering fields.
2. Illustrate the applications of Probability and statistics in various computer science and engineering Fields like data mining, classification problems etc.,
3. Discuss Sampling theory concepts to solve various engineering problems like structured and unstructured data models
4. Demonstrate Stochastic problem as Markov model as a problem solving methods for systematic model buildings.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO-1	Solve the problems of Curve fitting and regression in various problems in Computer Science and Engineering fields.	1,2,4	1
CO-2	Apply the concepts of Probability and statistics in various computer science engineering fields like data mining, classification problems etc.	1,2,4	1
CO-3	Apply the concept stochastic processes and Markov processes in Operating	1,2,4	1
CO-4	Make use of sampling theory concepts to solve various engineering problems like structured and unstructured data models.	2,3,4	1
CO-5	Apply the concepts of test of hypothesis for means and proportions and confidence limits for means.	1,2,4	1
CO-6	Apply the concepts of student's t-distribution and Chi-square distribution as a test of goodness of fit.	1,2,4	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6				√		

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3		1									3		
CO2	3	3		1									3		
CO3	3	3		1									2		
CO4		3		1									1		
CO5	2			1									1		
CO6		3		1									2		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Curve Fitting: Curve fitting by the method of least squares and fitting of the curves of the form,y = ax + b, y = ax2 + bx + c, y = aebx (*y = axb and y=abx)
Statistical Methods: Correlation-Karl Pearson's coefficient of correlation- problems. Regression analysis- lines of regression, problems. Rank correlation. Case study on statistical methods for data analysis.
** Application: Curve fitting and statistics for data science.

Probability distributions: Random variables, Discrete and continuous probability distributions. Binomial, Poisson, normal distributions (only problems) and *exponential (definition with one /two examples).
** Application: Probability distribution in machine learning, Computer vision: object recognition and image segmentation Computer graphics: behaviour of light and other physical phenomena in computer graphics.

Joint Probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance, and correlation coefficient. Stochastic processes- Stochastic processes, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability-simple problems.
** Application: Stochastic processes and Markov processes in Operating System.

Sampling theory: Sampling, Sampling distributions, standard error, test of hypothesis for means and proportions, confidence limits for means, student's t-distribution. Chi-square distribution as a test of goodness of fit. Case study on sampling for machine learning.
** Application: Sampling process in computer graphics, sampling theory in machine learning

** Application: (i) Additional information providing to students only for knowledge.

(ii) Major part of assignments questions chosen from applications.

(iii) Students' presentations/seminars topics chosen from applications.

TEXT BOOKS:

1. B.S. Grewal, "Higher Engineering Mathematics", 43nd edition Khanna Publishers, 2015.
2. ErwinKreyszig, "Advanced Engineering Mathematics", 9th edition Wiley Publications, 2013.
3. Seymour Lipschutz, John J. Schiller., "Schaum's Outline of Introduction to Probability and Statistics", McGraw Hill Professional, 1998.

REFERENCE BOOKS:

1. B.V. Ramana, "Higher Engineering Mathematics", 19th Reprint edition, Tata McGraw Hill Publications, 2013.
2. R.K.Jain and S.R.K.Iyengar, "Advanced Engineering Mathematics", 4th edition, Narosa Publishing House, 2016.
3. Sundarapandian, "Probability, Statistics and Queueing theory", PHI Learning, 2009.
4. Dr. B. Krishna gandhi, Dr. T.K.V. Iyengar, Dr. M.V.S.S.N. Prasad&S. Ranganatham.,"Probability and Statistics"S. Chand Publishing, 2015.
5. J. K. Sharma "Operations Research theory and applications", 5th edition, Macmillan publishers, ,2013.

JOURNALS/MAGAZINES

1. https://www.hindawi.com/journals/jps/
2. https://www.journals.elsevier.com/statistics-and-probability-letters
3. http://www.isoss.net/japs/
4. SWAYAM/NPTEL/MOOCs:
5. https://www.coursera.org/browse/data-science/probability-and-statistics
6. https://nptel.ac.in/courses/111/105/111105041/
7. https://onlinecourses.swayam2.ac.in/cec20_ma01/preview

SELF-LEARNING EXERCISES:

1. Curve fitting for application problems, Regression analysis for a bivariate data.
2. Probability distribution- Geometric, gamma- distributions, Joint probability distribution of continuous random variables.
3. Sampling analysis of real time problems. Applications to computer science: Data mining, classification problems etc.

TEXT BOOKS:

1.B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 43nd edition, 2015.
2.Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley Publications, 10th edition, 2015.

REFERENCE BOOKS:

1.B.V. Ramana, "Higher Engineering Mathematics", Tata McGraw Hill Publications, 19th edition, 2013.
2.R. K. Jain and S. R. K. Iyengar, "Advanced Engineering Mathematics", Narosa Publishing House, 5th edition, 2014.

Course Title	Physics for computer science				Course Type		FC	
Course Code	B24AS0106	Credit	3		Class		II Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practical				Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course introduces the basic concepts of Physics and its applications in Computer Science Engineering courses by emphasizing concepts like Construction and working of Laser sand optical Fibers in point-to-point communication. Quantum Mechanics and its application in Quantum Computation and Haptic Technology. The course has basic laws, expressions, and theories which help to increase the scientific knowledge to analyse upcoming technologies. The course also consists of real-time applications and numerical examples, which makes the course interesting and attractive.

COURSE OBJECTIVES:

1. The objectives of this course are to:
2. Interpret the knowledge of lasers, and optical fibers and their applications in computer science.
3. Exemplifying the fundamentals of quantum mechanics and its applications in quantum computing.
4. Study physics related to computation aspects.
5. Interpret the properties of dielectric materials for sensor applications
6. Understand the fundamentals and importance of haptic technology.

COURSE OUTCOMES(COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Describe the basic principles of lasers and relevant applications in computer science.	1,2,3	2
CO2	Apply the knowledge of dielectrics for different applications	1,2,3	2
CO3	Classify optical fibers with relevant applications	1,2,3	2
CO4	Discuss the fundamentals of quantum mechanics with applications	1,2,3	2
CO5	Illustrate the importance of quantum computation as an emerging technology.	1,2,3	2
CO6	Summarize the basics of haptic technology	1,2,3	2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	✓	✓	✓			
CO2	✓	✓	✓			
CO3	✓	✓	✓			
CO4	✓	✓				
CO5	✓	✓				
CO6	✓	✓				

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3													1
CO2	3	2													1
CO3	3	2													1
CO4	3	2													1
CO5	3	2													1
CO6	3	2													1

Note: 1-Low,2-Medium,3-High

COURSE CONTENTS:

Lasers: Laser and the characteristics, Interaction of radiation with matter (induced absorption, spontaneous and stimulated emission); Expression for energy density of radiation at thermal equilibrium in terms of Einstein's coefficients, Conditions for laser operation (population inversion and Meta stable state); Requisites of laser system, semiconductor laser and its application in laser printing. Numerical.

Dielectric materials: Introduction, Polarization, susceptibility, polarizability, polarization mechanisms, Piezoelectric, pyroelectric and ferroelectric dielectrics, properties and applications

Optical Fibers: Construction and light propagation mechanism in optical fibres (total internal reflection and its importance), Acceptance angle, Numerical Aperture (NA), Expression for numerical aperture in terms of core and cladding refractive indices, Condition for wave propagation in optical fibre, V-number and Modes of propagation, Types of optical fibres, Attenuation and reasons for attenuation, Applications: Explanation of optical fibre communication using block diagram, Optical source (LED) and detector (Photodiode) and their applications, Advantages and limitations of optical communications. Numerical.

Concepts leading to quantum mechanics: De-Broglie hypothesis, Expression for de-Broglie wavelength of an electron in terms of accelerating potential. Phase velocity and group velocity, Relation between phase velocity and group velocity in terms of C, relation between particle and group velocity. Numericals

Quantum Physics: Need for quantum mechanics, Wave function, properties of wave function and physical significance.

Probability density and Normalization of wave function, Schrodinger independent wave equation, Eigen values and Eigen functions. Applications of Schrödinger wave equation – energy; Eigenvalues of a free particle, Particle in one dimensional infinite potential well with numerical examples. Quantum structures, quantum wire, quantum well and quantum dot structures.

Quantum Computation: Quantum bits and representation of qubits, distinction between classical bit and quantum bit,quantum logic gates; Qubit as a two-level system. Introduction to quantum Information technology and quantum cryptography. Distinction between classical computer and quantum computer; Open challenges in quantum computation.

Introduction to Haptics: Types and importance of Haptics

Self-Learning Component:

Holograms using LASERs; Ultrasonics; and Statistical Physics for Computing

TEXTBOOKS:

1. William T. Silfvast, "Laser Fundamentals", Cambridge University press, New York, 2nd Ed, 2004
2. Halliday, R. Resnick, and J. Walker, "Fundamentals of Physics",John Wiley and Sons, New York, 10th edition, 2013.
3. R. K. Gaur and S.L. Gupta, "Engineering Physics", Dhanpat Rai Publications (P) Ltd, New Delhi. 10th Ed, 2014.
4. M.N. Avadhanulu and P.G. Kshirsagar, "A textbook of Engineering Physics", S. Chand and Company, New Delhi, 17th Ed, 2014.
5. Lorrain and O. Corson, "EM Waves and Fields",3rd Ed, CBS Publishers, 2003.
6. Dr. Nidhi, Ismail Keshta, Dr. Abhishek Agarwal, Dr. Vandana Rathore, "Quantum computing for beginners", Xoffencer International Publisher, 1st Ed, 2023.
7. Martin Grunwald, "Human Haptic Perception Basics and Applications", Springer Basel AG, 1st Ed, 2008.
8. R.A., Serway, John W. Jewett, Physics for Scientists and Engineers, Technology, Lachina publishing Services, 9th Ed, 2015.

REFERENCEBOOKS:

1. Arthur Beiser, "Concepts of modern Physics", Tata McGraw Hill publications, NewDelhi, 8th Ed, 2011.
2. S. O. Pillai, "Solid State Physics", New Age International publishers, New Delhi, 9th Ed, 2010.
3. Michael A. Nielsen, Isaac L. Chuang," Quantum Computation and Quantum Information",Cambridge University Press, 1st Ed, 2010.

Additional material:

1. LASER: https://www.youtube.com/watch?v=WgzynezPiyc
2. Optical Fiber : https://www.youtube.com/watch?v=N_kA8EpCUQo
3. Polarization: https://www.youtube.com/watch?v=KBJl1qiYOgo
4. Quantum Mechanics : https://www.youtube.com/watch?v=p7bzE1E5PMY&t=136s
5. Quantum Computing : https://www.youtube.com/watch?v=jHoEjvuPoB8
6. Quantum Computing :https://www.youtube.com/watch?v=ZuvCUU2jD30
7. Statistical Physics Simulation : https://phet.colorado.edu/sims/html/plinko-probability/latest/plinkoprobability_en.html
8. NPTEL Quantum Computing : https://archive.nptel.ac.in/courses/115/101/1151010

Course Title	Fundamentals of Mechanical Engineering				Course Type		HC	
Course Code	B24ME0105	Credit	3		Class		II Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	0	0	0	Theory	Practical	CIE	SEE
	Practical	0	0	0				
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

This course is designed to understand the concepts of green energy, hybrid vehicles, IC engines, refrigeration and air condition system, mechatronics, control systems, advanced manufacturing and automation. Further, this course includes applications of python programming for mechanical engineering. The students will be imparted to evaluate the performance of IC engines, use of machine elements like, belt drives, chain drives and gear drives. Students are introduced to the modern manufacturing technologies like CNC, 3D printing and applications of IIoT, Machine learning and robotics.

COURSE OBJECTIVES:

The objectives of this course are:

1. To develop the basic knowledge on energy resources, green energy concept, IC engines, turbines and refrigeration and air conditioning.
2. To incorporate the concept of different types of machine elements such as, gear drives, belt drives & chain drives.
3. To incorporate the concepts of mechatronics and robotics systems, modern manufacturing processes like CNC, additive manufacturing and digital manufacturing technology and its applications.
4. To provide a basic understanding role of computer science in mechanical engineering applications.
5. To provide the basic understanding of coding for mechanical engineering applications.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO	Course Outcomes	POs	PSOs
CO1	Apply the knowledge of energy resources to analyze real-world energy challenges and propose potential solutions and understanding of the concept of sustainability in the context of energy resources.	1, 2	1,2
CO2	Develop a comprehensive understanding of the working principles and application underlying prime movers and evaluate the performance of IC engines using python programming.	1, 2, 5	1,2
CO3	Gain a solid understanding of the fundamental principles refrigeration and air conditioning systems and develop a comprehensive understanding of working mechanical drives such as gears, belts, chains.	1	1,2
CO4	Identify the key elements of mechatronics system and interface sensors / transducer output with microprocessor for design or controlling of the system.	1, 5	1
CO5	Gain a comprehensive understanding of the principles and concepts underlying Industry 4.0, including, the Internet of Things (IoT), Machine Learning (AI), 3D printing and digital technologies.	1, 5	1,2,3
CO6	Understand the applications of robotics in various fields, such as manufacturing, agriculture, logistics, aerospace, etc and understand the concept of measurement system and control system used in robotics.	1	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3		√				
CO4		√	√			
CO5		√				
CO6		√				

COURSE ARTICULATION MATRIX:

CO / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1											2	1	
CO2	2	2			1								2	1	
CO3	2												2	1	
CO4	2				1								2		
CO5	2				1								2	1	1
CO6	2												2	1	1
Average	2	1.5			1								2	1	1

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Energy Resources: Types of energy resources: Green energy concept – Hydrogen energy; production using electrolysis process, advantages and disadvantages. Solar P-V system and wind energy.

IC Engines: Fuels and properties, classification of IC engines, parts and terminology, working of 4-stroke petrol engine, numerical on IP, BP, FP and Mechanical Efficiency, python programming for evaluation of engine performance.

Concept of electric and hybrid vehicles.

Gas Turbines: Open and closed cycle gas turbine, hydraulic turbines, Classification and working of Pelton wheel.

Refrigeration and Air Conditioning: Principle of Refrigeration system, working of domestic refrigerator (VCR) and window air conditioner.

Machine Elements: Belt drives, open and crossed belt drive, gear drives and chain drives, numerical on velocity ratio, python programming for computation of speed ratio.

Mechatronics Systems: Definition of mechatronics, components of mechatronics, basic Terminologies, Open loop and Closed loop control systems, microprocessor based control systems, static characteristics of sensor, working of capacitance sensor, eddy current sensor, hall effect sensor-Light sensors, optical encoders.

Introduction to Advanced Manufacturing and Industry 4.0 -Computer numerical control (CNC) machines, laser engraving, 3D printing, classification, applications, working of FDM, Introduction to Digital manufacturing- machine learning (ML) applications, Industrial Internet of Things (IIoT), Human Machine Interface.

Robotics and Automation: Need of Automated Guided Vehicle (Retrieval/storage system) in industries, applications of Robots, measurement system, open and closed loop control system. Adoption of logic gates in automation.

Self-Learning Component:

Hybrid vehicles, Engineering Materials, Need of Automated Guided Vehicle. Basics of numerical python and MAT Plot.

TEXT BOOKS:

1. K R Gopala Krishna, Sudheer Gopala Krishna and S C Sharma, “Elements of Mechanical Engineering”, Subhash Publishers, 13th Edition, 2015.
2. Roy & Choudhury, "Elements of Mechanical Engineering", Media Promoters & Publishers Pvt. Ltd, 2000.
3. Manaranjan Pradhan and Dinesh Kumar U, “Machine Learning using Python”, Wiley Publishers, 1st Edition, 2019

REFERENCE BOOKS:

1. SKH Chowdhary, AKH Chowdhary and Nirjhar Roy, “The Elements of Workshop Technology - Vol I & II”, Media Promotors and publisher, 11th Edition, 2001.
2. William Bolton, “Mechatronics Electronics Control Systems in Mechanical and Electrical Engineering”, Pearson, 2015.
3. K. K. Appukuttan, “Introduction to Mechatronics”, Oxford University Press, 2007.
4. Madhumathy P, M Vinoth Kumar, R. Umamaheswari “Machine Learning and IoT for Intelligent Systems and Smart Applications” CRC Press publisher, 2021.
5. John V. Guttag, “Introduction to Computation and Programming using Python”, MIT Press, 2016.

JOURNALS/MAGAZINES

1. International Journal of Refrigeration.
2. https://www.journals.elsevier.com/international-journal-of-hydrogen-energy
3.https://www.researchgate.net/publication/369942210_Fuel_Cell_Electric_Vehicle_Modeling_Using_HybridBatteryFuel_Cell_Vehicle_Modeling_Tool_HBFCMT.

SWAYAM/NPTEL/MOOCs:

1. https://www.coursera.org/browse/physical-science-and-engineering/mechanical-engineering
2. https://www.my-mooc.com/en/categorie/mechanical-engineering

Course Title	Advanced C programming and Applications				Course Type		HC	
Course Code	B24CI0201	Credits	2		Class		II Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	2	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	2	3	3	42	-	50	50

COURSE OVERVIEW:

Algorithms and flowcharts are the fundamental tools for problem solving which can be used by the computers. The computer programs can be developed using algorithms and flowcharts to provide solutions to problems. C Language is a general-purpose, structured and procedure oriented programming language. It is one of the most popular computer languages today because of its structure and higher-level abstraction C. This course introduces algorithms, flowcharts and various C Programming language constructs for the development of real world applications.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Review and build upon the fundamentals of C programming.
2. Define and use structures and unions to manage related data efficiently..
3. Handle file input and output operations using standard library functions.
4. Understand and implement various data structures for efficient programming.
5. Understand the working of the Unix platform for shell scripting.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Review and apply fundamental C concepts to learn advanced techniques.	1-3	1
CO2	Structure and manage code efficiently using structures, unions, and enumerations	1-5	1
CO3	Perform file operations such as opening, reading, writing, and closing files, and manage data stored in files.	1-3,5	2,3
CO4	Develop modular and robust C programs using standard library functions..	1-5	2,3
CO5	Write shell scripts on the Unix platform.	1-5	2,3
CO6	Understand the memory management system and to use relevant functions for effectively memory utilization	1-5,9	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1				✓		
CO2			✓			
CO3			✓			
CO4						✓
CO5		✓	✓			
CO6						✓

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1	3										3		
CO2	1	3	2	2	2								3		
CO3	2	2	2		1									3	3
CO4	3	3	3	1	1									3	3
CO5	3	3	3	2	2										
CO6	3	3	3	2	2				3				3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

1. Review of C Fundamentals: Quick review of basic C programming concepts. Control and looping statements, arrays, strings and pointers.
2. Structures, Unions and Enumerations: Introduction to Structures, Defining and Using Structures, Arrays of structures, Pointers to Structures, Self-Referential Structures, Unions, Enumerations, Typedef.
3. File Input and Output: File Operations, reading from and Writing to Files, Binary File I/O, File Handling Functions.
4. Data Structures: Linked lists: singly, doubly, and circular linked lists, Stacks and queues: implementation and applications.
5. C Standard Library: Standard I/O Functions, Mathematical Functions, Utility Functions, Memory Management Functions.
6. Shell Scripting using C: Overview of UNIX operating system, UNIX architecture, Features of UNIX, Introduction to UNIX file system, Basic UNIX commands and utilities, UNIX shells: Bourne Shell, C Shell, Korn Shell, and Bash, Shell scripting basics, Variables, operators, and control structures in shell scripts, writing and executing shell scripts.
7: Dynamic Memory Allocation: Understanding Dynamic Memory Management, using malloc, calloc, realloc, and free and Memory allocation errors & Command-Line Arguments

Self Learning Component:

Evaluations of programming languages

Different programming languages and their applications in real world
Know more about most popular programming languages in the world by using : https://www.tiobe.com/tiobe-index/

TEXT BOOKS:

1. B.W. Kernighan & D.M. Ritchie, "C Programming Language", 2nd Edition, PRENTICE HALL SOFTWARE SERIES, 2005.
2. Herbert Schildt, “C: The Complete Reference”, 4th edition, TATA McGraw Hill, 2000.
3. B.S. Anami, S.A. Angadi and S. S. Manvi, “Computer Concepts and C Programming: A Holistic Approach”, second edition, PHI,2008.
4. Yashavant Kanetkar, “Let us C” , BPB , 19th edition, 2022
5. Jon Bentley, “Programming Pearls”, 2nd edition, PEARSON,2002.

REFERENCE BOOKS:

1. Balaguruswamy,” Programming in ANSI C”, 4th edition, TATA MCGRAW Hill, 2008.
2. Donald Hearn, Pauline Baker,“ Computer Graphics C Version”, second edition, Pearson Education, 2004.
3. Jon Bentley, “Programming Pearls”, 2nd edition, PEARSON,2002.

JOURNALS/MAGAZINES:

1. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6294 (IEEE Journal/Magazine on IT Professional)
2. https://ieeexplore.ieee.org/document/1267572 (IEEE Computing in Science and Engineering)

SWAYAMNPTEL/MOOCs

1. https://online courses.nptel.ac.in/noc20_cs06/preview (Problem Solving through Programming in C)
2. https://www.edx.org/course/c-programming-getting-started (C Programming Getting started)
3. https://www.coursera.org/specializations/c-programming (Introduction to C programming)

Course Title	Internet of Things				Course Type		HC(Integrated)	
Course Code	B24EN0101	Credit	2		Class		II Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	2	3	3	14	28	50	50

COURSE OVERVIEW:

The Internet of Things (IoT) expands access to the world-wide web from computers, smartphones, and other typical devices to create a vast network of appliances, toys, apparel, and other goods that are capable of connecting to the Internet. This introductory course focuses on IoT architecture, Boards, sensors, actuators and communication protocols. The course is supported with hands on sessions that incorporates different types of sensors interfaced with IoT board to build IoT projects to solve real time problems. The case study of deployment of IoT in various applications are provided.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To understand the fundamentals of Internet of Things.
2. Inculcate knowledge of IoT devices, Sensors and Communication Protocols in various application domains.
3. Gain expertise in interfacing of various sensors to IoT Boards, Arduino and Node MCU.
4. Discuss various applications of IoT.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	Pos	PSOs
CO1	Describe IoT architecture and Models of IOT	1,2,3,4,5	1,2
CO2	Identify and work on different IoT development boards	1,2,3,4,5	1,2
CO3	Identify communication technologies, protocols, and cloud services	1,2,3,4,5	1,2
CO4	Identify sensors & actuators and demonstrate the interfacing of sensors & actuators to IoT boards	1,2,3,4,5,9,10, 12	1,2
CO5	Interpret various Applications of IoT	1,2,3,4,5,6,9,10,12	1,2,3
CO6	Develop simple IoT projects and modules	1,2,3,4,5,9,10,11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√	√				
CO2	√	√				
CO3	√	√	√			
CO4	√	√	√	√		
CO5	√	√	√	√		
CO6	√	√	√	√		

COURSE ARTICULATION MATRIX

CO#/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1								3	3	
CO2	2	3	1	1	1								3	3	
CO3	3	2	1	1	2								2	2	
CO4	3	3	3	2	2				2	1		2	2	2	
CO5	3	2	2	1	2	2			2	1		2	2	2	1
CO6	3	2	2	1	2				2	2	2	2	2	2	2

Note:1-Low,2-Medium,3-High

COURSE CONTENT

IoT Basics and Arduino UNO Board

Introduction to IoT and its working, IoT Architecture, Models of IOT, Characteristics of IoT, Merits, demerits, futures scope and challenges of IoT, IOT Echo system, Arduino UNO board, memories, microcontroller, Basic programming of Arduino UNO board: Numeric data types, text data types, Array data types, Digital input output functions, Communication Technologies: Bluetooth, ZigBee, Wi-Fi, Cellular

IoT Enabling Technologies and Applications

IoT Development Boards: Node MCU, Raspberry Pi; Sensors and Actuators: Temperature Sensor, PIR Sensor, Ultrasonic senor, LDR, Applications of IoT, Protocols: HTTP, MQTT, CoAP; IoT Cloud Platforms: Arduino Cloud, Thing Speak, Blink Cloud

Experiments:

Sl. No.	Name of the Practice Session	Tools and Techniques	Expected Skill /Ability
	Arduino UNO		
1	Experiment-01 Introduction to Arduino Board & getting started with Arduino IDE software. Write a program to blink an LED: a) Infinite number of times with ON & OFF duration of 1 sec b) Only 3 times with ON and OFF duration 2 sec	Hardware & software for Arduino UNO Arduino UNO Board Arduino IDE LED	Identifications of various parts of Arduino UNO and Coding
2	Experiment-02 a) Write a program to increase and decrease the brightness of LED. b) Write a program to control the brightness of LED using Potentiometer.	Arduino UNO Arduino IDE LED Potentiometer	Interface of Potentiometer
3	Experiment-03 a) Write a program to interface LDR to Arduino board and display the voltage across LDR on serial monitor. b) Write a program to control the brightness of LED based on the intensity of light on LDR.	Arduino UNO Arduino IDE LED LDR	Interface LDR sensor
4	Experiment-04 a) Write a program to interface temperature sensor and display the values on the serial monitor. b) Write a program to display the range of temperature on LCD.	Arduino UNO Arduino IDE LCD Temperature sensor	Interface Temperature sensor
5	Experiment-05 Write a program to interface ultrasonic sensor and display the distance from an object.	Arduino UNO Arduino IDE Ultrasonic sensor	Interface Ultrasonic sensor
	NODU MCU		
8	Experiment-06 Introduction to NODE MCU and Write a Program for 2-BIT COUNTER using Node MCU.	Hardware & software for NODE MCU NODE MCU Board Two LED's	Identifications of various parts of NODE MCU and Coding

9	Experiment-07 Write a program to interface motion sensor and display its status using LED. If motion is detected, turn on LED otherwise keeps the LED off.	NODE MCU LED PIR sensor	Interface PIR sensor
10	Experiment-08 Write a program to interface Single LED blinking using Node MCU and Bluetooth module.	Node MCU Bluetooth module LED	Connection of Bluetooth to NODE MCU
11	Experiment-09 Write a program to control the status of LED using relay.	Node MCU Bluetooth module LED	Control of LED through Relay
12	Experiment-10 Write a program to interface temperature sensor and display the values on Web Page/Cloud.	Node MCU Temperature Sensor LED Cloud Service	Connect to cloud and display the values

TEXTBOOKS:

1. Vijay Madisetti, Arshdeep Bahga , "Internet of Things: A Hands-On- Approach " Second edition 2014, ISBN: 978 0996025515. REFERENCE BOOKS:
2. Raj Kamal," Internet of Things: Architecture & design Principle", McGraw Hill Education 2017.

Additional Material:

1. https://www.coursera.org/learn/iot
2. https://www.coursera.org/learn/interface-with-arduino
3. https://www.youtube.com/watch?v=APH6Nrar27w&list=PLYwpaL_SFmcB8fDd64B8SkJiPpEIzpCzC
4. https://www.youtube.com/watch?v=NSUj_NMV5t0&list=PLuAADu3OvBt4SUxlYPu_xJogSmVfSLZF0

Course Title	Physics for Computer science Lab				Course Type		FC	
Course Code	B24AS0208	Credits	1		Class		II Semester	
	LTP Lecture	Credits -	Contact Hours -	Workload -	Total Number of classes per Semester		Assessment in Weightage	
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2	0	28	25	25

COURSE OVERVIEW:

Engineering Physics is very important and necessary basic subject for all branches of engineering students. It provides the fundamental knowledge of basic principles of Physics which is required for basic foundation in engineering education irrespective of branch. This course introduces the experimental concepts of Physics and its applications to Computer Science Engineering courses by emphasizing the following concepts: electrical properties, dielectrics, semiconductors, LASERs, and optical properties. This course provides a basic understanding of the working of different electronic components. The course also emphasizes simulating the working of some electronic components.

COURSE OBJECTIVE(S):

The objectives of this course are to:

1. Demonstrate the principles covered in your study material in physics.
2. Provide familiarity with apparatus and enable them to handle the instruments and apparatus with purpose.
3. Identify the process to study conditions for a given experiment.
4. Develop an attitude of perfection in practical tasks.
5. Simulate the working of different electronic components using mobile or a computer.

COURSE OUTCOMES (COs)

After the completion of the course the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Constructing simple circuits and performing experiments to study voltage-current response.	1, 2,3 4	1,2,3
CO2	Determination of unknown physical parameters related to LASERs	1,2, 3,5	2, 3
CO3	Determination of unknown physical parameters related to optical fibers.	2,3, 4, 8,	1, 2, 3
C04	Determine the dielectric constant of the material.	2,3, 4,8,	1, 2, 3
CO5	Simulating the experimental data of physical experiments	1 to 5	1, 2, 3
CO6	Simulate the working of electronic circuits	1 to 5, 8 to 10	1,2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply(L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1						
CO2						
CO3						
C04						
CO5						
CO6						

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3							3	3					1
CO2	3	3							3	3					2
CO3	3	2							3	3					1
CO4	3	2							3	3					1
CO5	3	2							3	3					2
CO6	3	2							3	3					1

List of Experiments

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Light-dependent characteristics of a photosensor.	Photodiode, Ammeters, and voltmeters.	Circuit construction, Perform, and plotting of data using Matlab/origin
2	Determination of wavelength of IR LEDs and energy gap in applications to remote controllers.	LEDs, AC source, and voltmeters.	Circuit construction, Perform, and plotting of data using Mat lab/origin
3	Simulation of experimental data using Origin software	Origin software	Simulated graphs
4	Determination of numerical aperture and acceptance angle to launch the optical power into the optical fiber for digital communication.	LASER. Optical fiber light source, DC supply	Experimental setup, plotting of data using Matlab/origin
5.	Determination of the attenuation coefficient of the given optical fiber for long-distance communication.	LASER. Optical fibers light source, DC supply	Experimental setup, plotting of data using Matlab/origin
6	Determination of wavelength of LASER for LASER printing and digital communications.	LASER. Optical fiber light source, DC supply	Experimental setup, plotting of data using Matlab/origin

7	Verification of splicing loss experimental for optical fiber.	Diode laser, digital dc micrometer two OFC (1.5m &2.5m), optical sensor	Circuit construction, Perform, and plotting of data
8	Determination of dielectric constant using charging and discharging of capacitor	Capacitor, resistor, voltmeters	Circuit construction, Perform, and plotting of data
9	Study of motion using spreadsheets	Simulation software	Circuit construction, Perform, and plotting of data
10	PhET interactive simulations (Projectile motion, wave-particle dualism).	Simulation software	Circuit construction, Perform, and plotting of data

Part B: Demo and Simulation

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Generation of ultrasonic waves by piezoelectric transducers and its velocity determination in liquid for the Haptics applications.	piezoelectric transducers	To understand the concept of Haptics
2	Demonstration of superposition of waves of monochromatic light.	Oscillators and CRO	Interference of waves to understand the quantum superposition
Experiments for the mathematical model			
3	Bending of a single cantilever, its mathematical model. Time-period calculation of simple pendulum, its mathematical model.	Every circuit (android app) Tina (Online simulator)	Visualize, simulate, and analyze

TEXTBOOKS:

1. M.N. Avadhanulu and P.G. Kshirsagar, “A Textbook of Engineering Physics”, S. Chand & Company Ltd, New Delhi, 10thEd, 2008.
2. Gaur and Gupta, “Engineering Physics”,Dhanpat Rai Publications, 6thEd, 2017.
3. S.P. Singh, “Advanced Practical Physics Vol.- I”, Pragati Prakashan, 15th Ed, 2017.
4. G.L.Souires, “Practical Physics”, Cambridge University, UK, 4th Ed,2001.

Reference Books:

1. Arthur Beiser, “Concepts of Modern Physics”, Tata McGraw Hill Edu Pvt. Ltd- New Delhi, 6thEd, 2006.
2. S O Pillai, “Solid State Physics”, New Age International Publishers, 8thEd, 2018.
3. S M Sze, Physics of Semiconductor Devices, Wiley, 3rd Ed, 2004.
4. Walter Fox Smith, “Experimental Physics”-Principles and Practice for the Laboratory, CRC Press, Taylor & Francis group, 1stEd, 2020.

Course Title	Advanced C programming and Applications Lab				Course Type		HC	
Course Code	B24CI0202	Credits	1		Class		II Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	IA	SEE
	Practice	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

Pseudocode, Algorithms and flowcharts are the fundamental tools for problem solving which can be used by the computers. The computer programs can be developed using algorithms and flowcharts to provide solutions to problems. C Language is a general-purpose, structured and procedure-oriented programming language. It is one of the most popular computer languages today because of its structure and higher-level abstraction C. This course introduces algorithms, flowcharts and various C Programming language constructs for the development of real world applications.

COURSE OBJECTIVE (S):

1. Define and use structures & Unions to group and manage related data efficiently.
2. Perform file input and output operations using standard library functions.
3. Understand structuring of data using data structures for efficient programming
4. To understand working of Unix platform for shell scripting.

COURSE OUTCOMES (COs)

On successful completion of this course; the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To review C fundamentals to learn advanced concepts	1-3	1
CO2	To make the code structured & easier to read and maintain, using concept of structures, unions & integral constants	1-5	1
CO3	Perform file operations such as opening, reading, writing, and closing files. Utilize file handling	1-3,5	2,3
CO4	Develop the ability to write efficient, modular, and robust C programs using standard C library functions.	1-5	2,3
CO5	Learn to write shell scripts on Unix platform.	1-5	2,3
CO6	Interface C code from other programing languages.	1-5,9	2,3

BLOOM'S LEVELOF THECOURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1				✓		
CO2			✓			
CO3			✓			
CO4						✓
CO5		✓	✓			

CO6						✓

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	**2**	**1**	**3**										**3**		
CO2	**1**	**3**	**2**	**2**	**2**								**3**		
CO3	**2**	**2**	**2**		**1**									**3**	**3**
CO4	**3**	**3**	**3**	**1**	**1**									**3**	**3**
CO5	**3**	**3**	**3**	**2**	**2**										
CO6	**3**	**3**	**3**	**2**	**2**				**3**				**3**	**3**	**2**

Note:1-Low,2-Medium,3-High

Sl. No	**Program**
1	Design and Develop C program to perform following operation using string. Concatenation of two strings Length of a string
2	Program to count the total number of vowels and consonants in a string.
3	Write a C program to define a structure named Em with name and DOB, where, DOB in turn is a structure with day, month and year. Using the concept of nested structures display your name and date of birth. Defines a structure to hold employee information (name, ID, salary). b. Allows the user to input details for multiple employees. c. Displays the entered employee details.
4	Write a C program to define a structure named Student with name and DOB, where, DOB in turn is a structure with day, month and year. Using the concept of nested structures display your name and date of birth.
5	Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX). Push an Element on to Stack Pop an Element from Stack Demonstrate Overflow and Underflow situations on Stack Display the status of Stack Exit

	Support the program with appropriate functions for each of the above operations
6	Design, Develop and Implement a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX) Insert an Element on to Circular QUEUE Delete an Element from Circular QUEUE Demonstrate Overflow and Underflow situations on Circular QUEUE Display the status of Circular QUEUE Exit Support the program with appropriate functions for each of the above operations
7	Write a C program for the following file operations: Program to create and write data to a text file. Program to read data from a text file and display it on the console.
8	Design and demonstrate the use of file handling functions like fopen(), fclose(), fgets(), fputs(), fread(), and fwrite().
9	Create, delete, and manage files and directories using touch, rm, mkdir, rmdir, and mv.
10	Write a script to check if a number is positive, negative, or zero.

TEXT BOOKS:

1. B.W. Kernighan & D.M. Ritchie, "C Programming Language", 2nd Edition, PRENTICE HALL
2. SOFTWARE SERIES, 2005.
3. Herbert Schildt, “C: The Complete Reference”, 4th edition, TATA McGraw Hill, 2000.
4. B.S. Anami, S.A. Angadi and S. S. Manvi, “Computer Concepts and C Programming: A
5. Holistic Approach”, second edition, PHI,2008.

REFERENCE BOOKS:

1. Balaguruswamy,” Programming in ANSI C”, 4th edition, TATA MCGRAW Hill, 2008.
2. Donald Hearn, Pauline Baker,“ Computer Graphics C Version”, second edition, Pearson Education, 2004.
3. NPTEL Courses- www.swayam.go

Course Title	Innovation & Entrepreneurship				Course Type		FC Integrated	
Course Code	B24ME0102	Credits	2		Class		II Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	2	3	3	14	28	50	50

COURSE OVERVIEW

NEN Ignite is an entrepreneurship program based on experiential learning that aims to support startups' founders through a structured pathway from Idea Discovery to Pitch Deck.

COURSE OBJECTIVES

The objectives of this course are to:

1. Discover an entrepreneurial opportunity
2. Articulate a compelling value proposition
3. Build a sustainable business model and business plan
4. Create and validate an MVP with potential customers
5. Select an appropriate Go-to-Market Strategy
6. Pitch the business idea to different stakeholders

COURSE OUTCOMES (CO'S)

On successful completion of this course; the student shall be able to:

CO	Course Outcomes	POs	PSOs
CO1	Identify the different aspects that can impact their business	3,9,10,11,12	1
CO2	Acquire in-depth knowledge about tools to build any business idea	3,9,10,11,12	1
CO3	Acquire in-depth knowledge about the different growth tools to grow their business.	3,9,10,11,12	1
CO4	Create a financial plan for their business	3,9,10,11,12	1
CO5	Create a pitch deck for their business and present it to different stakeholders	3,9,10,11,12	1

COURSE ARTICULATION MATRIX

CO/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			2						2	2	3	2	2		
CO2			2						2	2	3	2	2		
CO3			2						2	2	3	2	2		
CO4			2						2	2	3	2	2		
CO5			2						2	2	3	2	2		

COURSE CONTENT

Introduction to Entrepreneurship: Entrepreneurs; entrepreneurial personality and intentions - characteristics, traits and behavioural; entrepreneurial challenges. Taking product or service ideas to creating value: Why should one choose to become an entrepreneur, Entrepreneurial mind-set, Intrapreneurship.

Orientation for WE Ignite program, Ice Breaking session, self-work Instructions and timelines Platform Demo Introduction to Ignite program flow and milestones , Introduction to Entrepreneurship and Human centred Approach to Design Thinking , Are you enterprising?. New generations of entrepreneurship viz. social entrepreneurship, Edupreneurship, Health entrepreneurship, Tourism entrepreneurship, Women entrepreneurship etc., Barriers to entrepreneurship, Creativity and entrepreneurship, Innovation and inventions, Skills of an entrepreneur, Decision making and Problem Solving

100 Rupee Venture; Debrief of Group Activity- Presentation and Sharing Learning Experience

Entrepreneurial Opportunities: Opportunities. Discovery/ creation, Pattern identification and recognition for venture creation: prototype and exemplar model, reverse engineering. Problem Identification and Opportunity Discovery. Entrepreneurial Process and Decision Making: Entrepreneurial ecosystem, Ideation, development and exploitation of opportunities; Negotiation, decision making process and approaches, Effectuation and Causation

Customer and Markets : Customer Discovery: Exploring Customer Personas & Market Estimation for your Ideas, Create a compelling value proposition & Competitive Advantage

Build your MVP : Building a MVP that customers Love

Crafting business models and Lean Start-ups: Introduction to business models; Creating value propositions-conventional industry logic, value innovation logic; customer focused innovation; building and analysing business models; Business model canvas, Introduction to lean start-ups, Business Pitching

Business Model: Developing strong business models Create and present your Lean Canvas Financial Feasibility: Introduction to Business plan and its components; Basics of Finance. Institutional Support for Entrepreneurship:

Organization Assistance to an entrepreneur, New Ventures, Industrial Park (Meaning, features, & examples), Special Economic Zone (Meaning, features & examples), Financial assistance by different agencies, MSME Act Small Scale Industries, Carry on Business (COB) license, Environmental Clearance, National Small Industries Corporation (NSIC), e-tender process, Excise exemptions and concession, Exemption from income tax, The Small Industries Development Bank of India(SIDBI), Incentives for entrepreneurs

Go To market Strategy: Getting products to market: Channels & Strategies; Managing growth and Targeting Scale: Understand the Unit economics for your venture; Funding Strategy: Securing funding for your Startup and Preparing for pitch.

TEXT BOOK:

1. Wadhwani Foundation Curriculum K. Ramachandran, “Entrepreneurship Development”, Tata Mc. Graw Hill, 2008
2. Sangeeta Sharma, “Entrepreneurship Development” PHI Publications, 2016

REFERENCE BOOKS:

1. Baringer and Ireland, “Entrepreneurship”, Pearson, 11th Edition, 2020.
2. Drucker F Peter: “Innovation and Entrepreneurship”, 1985.Heinemann, London.
3. Doanld F Kuratko & Richard M Hodgeth, “Entrepreneurship in the New Millennium”, India Edition - South-Western,
4. Cengage Learning Entrepreneurship –by Robert D. Hisrich (Edition-9)
5. Entrepreneurship- Theory, Process & Practice –by Kuratko & Hodgetts, Thompson South-Western Publication
6. Technology Entrepreneurship Taking Innovation to the Marketplace – by Thomas N. Duening, Robert D. Hisrich and Michael A. Lechter, Elsevier

JOURNALS/MAGAZINES

1. International Small Business Journal: https://journals.sagepub.com/home/isb
2. Journal of Development Entrepreneurship: https://www.worldscientific.com/worldscinet/jde

SWAYAM/NPTEL/MOOCs:

1. Entrepreneurship: https://nptel.ac.in/courses/110/106/110106141/

Course Title	Communicative English				Course Type		FC	
Course Code	B24AH0103	Credit	1		Class		II Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	0	0	0				
	Tutorial	0	0	0	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2		28	25	25

COURSE OVERVIEW:

Today many companies are working on international projects where English is increasingly used by engineers across the world to communicate with all groups involved. This course is designed for entry-level Engineering and Technology curriculum enabling the students to learn, acquire, and apply for their learning and career. The course is aimed at providing effective skills for promoting communication skills through English. Hence in the present scenario Engineering English course is to be fully tailored to the specific needs of engineers. The outcome of this course contains a refined level of English proficiency by acquiring all four skills, listening, speaking, reading, and writing to prepare them for global readiness.

COURSE OBJECTIVES:

The objectives of this course are :

To develop the communicative competence of

learners To utilize language effectively in academic

contexts

To change various listening strategies to comprehend various types of audio materials like lectures, discussions, videos, etc.

To build on students' interest in English language skills by engaging them in listening, speaking, and grammar learning activities that are relevant to authentic contexts.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Analyse Listening, Comprehend and Correspond with others in various contexts	10	
CO2	Apply proper communication modules to Speak legibly and fluently under various lifetime situations	10	
CO3	Analyse the meaning and language while reading a variety of writings and technical texts	10	
CO4	Demonstrate products and processes and explain their uses and purposes clearly and accurately	10	
CO5	Make use of various communicative skills in precise and efficient ways in technological contexts	10	
CO6	Create situational conversations and technical writing styles for interpersonal and effective communication	10	

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1				√		
CO2			√			
CO3				√		
CO4		√				
CO5			√			
CO6						√

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1										2					
CO2										2					
CO3										2					
CO4										2					
CO5										2					
CO6										3					

Note:1-Low,2-Medium,3-High

COURSE CONTENTS

Fundamentals of Communication

Listening: Listening for general information-specific details- conversation: Introduction to classmates - Audio /video (formal & informal); Telephone conversation; Listening to voicemail & messages; Listening and filling a form. Listening to TV News and guest Lecturers.

Speaking: Pronunciation Common Vocabulary – Technical Vocabulary – Answering Peer Questions – Conversation with Teacher, making telephone calls Introduction; Introducing a friend

Reading: News magazines, Reading for unfamiliar words, Variety of News Items

Writing: Word formation – Auxiliary verbs – Modal Verbs – Sentence Types – Affirmative, Negative, Interrogative, Concord – Dialogue Writing, Paragraph writing Narration and Summation

Listening: Listening to Peer Conversations – Brief Speeches – Listening for Specific Information – Recap of Speeches, Listening to podcasts, anecdotes/stories / event narration; documentaries and interviews with celebrities.

Speaking: Telephonic enquiries official/formal enquiries, narrating personal experiences / events-Talking about current and temporary situations & permanent and regular situations - describing experiences and feelings- engaging in small talk- describing requirements and abilities.

Reading: Technical Essays – Identifying Sentence Types – Classifying the verb patterns

Writing: Tenses – Simple Present, Present Progressive, Present Perfect, Present Perfect Continuous – Voice – Active & Passive, email writing

Description of A Process / Product

Listening: Listen to product and process descriptions; a classroom lecture; and advertisements about products. Digital Videos for World Information

Speaking: Picture description- describing locations in workplaces- Giving instruction to use the product- explaining uses and purposes- Presenting a product- describing shapes and sizes and weights- talking about quantities(large & small)-talking about precautions.

Reading: Coherence, Development of Thoughts

Writing: Tenses – Simple Past, Past Progressive, Past Perfect, Past perfect continuous – Impersonal Passive-Narrating the past events, Permission Letter for Industrial Visit/Functions held

Classification and Recommendations

Listening: Listening to TED Talks; Listening to Dialects of English – British & American Regional Listening to achievers, eminent personalities

Speaking: Role Plays, Extempore, Responding to specific questions

Reading: Texts on self-confidence, motivation, success path, Comprehensive passages, Reading for specific points

Writing: Tenses – Simple Future, Future progressive, Future Perfect, Future Perfect continuous – Definition – Phrases of Reason – Cause & Effect, Report writing

Self Learning Component: Online Activities on LSRW Skills

Teaching Pedagogy: MALL, CALL, Online, Blended Teaching, Flipped Classroom

TEXT BOOKS:

1. Board of Editors. Using English A Course book for Undergraduate Engineers and Technologists. Orient Black Swan Limited, Hyderabad: 2015
2. Richards, C. Jack. Interchange Students' Book-2 New Delhi: CUP, 2015.

REFERENCE BOOKS:

1. Murphy, Raymond English Grammar in Use with Answers: Reference and Practice for Intermediate Students, Cambridge: CUP, 2004
2. Thomson, A.J. and Martinet, A.V. A Practical English Grammar, OUP, New Delhi: 1986
3. Anne Laws, —Writing Skills, Orient Black Swan, Hyderabad, 2011
4. Green, David. Contemporary English Grammar Structures and Composition. MacMillan, 2010.
5. Thorpe, Edgar and Showick Thorpe. Basic Vocabulary. Pearson Education India, 2012.
6. Leech, Geoffrey and Jan Svartvik. A Communicative Grammar of English. Longman, 2003.
7. Rizvi, M. Ashraf. Effective Technical Communication. Tata McGraw-Hill, 2005.
8. Riordan, Daniel. Technical Communication. New Delhi: Cengage Publications, 2011.
9. Sen et al. Communication and Language Skills. Cambridge University Press, 2015.

Course Title	Engineering Exploration				Course Type		FC	
Course Code	B24CIET01	Credits	1		Class		II semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	0	-	-				
	Practice	0	-	-	Theory	Practical	CIE	SEE
	Total	1	1	1	14		25	25

COURSE OVERVIEW:

This course provides a comprehensive introduction to the field of engineering, aimed at equipping students with the foundational knowledge and skills necessary to pursue a career in this dynamic and impactful discipline. The curriculum is designed to cover a broad spectrum of topics, from the historical and contemporary contributions of engineering to the principles and practices that define the profession today, with a particular emphasis on the Indian context.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Introduce in the field of engineering, emphasizing the role of engineers as problem solvers and innovators.
2. Familiarize with the global and Indian engineering industries, highlighting various career opportunities.
3. Develop an engineering mindset by analyzing the design and functioning of common engineering artifacts.
4. Enhance ability to work effectively in multidisciplinary teams, utilizing problem-solving skills, project management concepts, and effective communication of engineering ideas through written reports and presentations.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain what engineering is, the role of an engineer as a problem solver and appreciate the history and impact of significant engineering achievements especially in the Indian context.	1,2,12	1
CO2	Be familiar with the global and Indian engineering industry and the various career opportunities it offers.	1,2	1
CO3	Demonstrate Engineering Mindset by analyzing design and working of common engineering artefacts and Gain practical insights through case studies, product dissection, and industry interactions.	1,4	2
CO4	Apply the engineering design process and principles to identify needs, define problems, and develop solutions	2,3	1
CO5	Demonstrate the ability to work effectively in multidisciplinary teams, Use problem Solving Skills, apply project management concepts, and communicate engineering ideas through written reports and presentations.	5,9,10	3
CO6	Recognize the importance of professional ethics, sustainability, and the societal impact of engineering solutions, and incorporate these considerations into the design process.	6,7,8,12	3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√					
CO2		√				
CO3		√				
CO4			√			
CO5		√				
CO6				√		

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3										2	3		
CO2	3	3											3		
CO3	3		3											3	
CO4		3	3										3		
CO5					3				3	3					3
CO6						2	3	3				3			3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Definition of engineering, its scope, and various disciplines, Understanding the engineer's role in identifying, defining, and solving problems, Examples of engineering problem-solving in everyday life and industry.

Overview of engineering through different historical periods, Key milestones and developments in engineering practices, Case studies on significant engineering achievements and their societal impacts, Overview of historical engineering achievements in India (e.g., ancient water management systems, Indus Valley civilization), Modern engineering marvels in India (e.g., ISRO's achievements, infrastructure projects).

Present-day engineering issues (e.g., sustainability, climate change, urbanization), Role of engineers in addressing these challenges, Exploration of emerging fields (e.g., artificial intelligence, biotechnology, renewable energy), Career opportunities global and Indian engineering industry.

Definition and characteristics of an engineering mindset, Importance of critical thinking, problem-solving, and creativity in engineering, Design thinking process, Application of design principles in engineering artefacts, Common engineering product for its design principles and functionality.

Case Study Analysis: Successful Engineering Designs, Examination of case studies (e.g., bridges, consumer electronics,

medical devices), Factors contributing to the success of these designs, Product Dissection, Techniques and importance of product dissection in understanding engineering principles, Hands-on dissection of a chosen engineering product (Mobile or Two Wheller or Laptop).

Importance of Multidisciplinary Teams in Engineering Projects, Team Dynamics and Collaboration, Engineering Problem- Solving Process, Application of problem-solving frameworks (e.g., TRIZ, Six Sigma), Project Management Concepts in Engineering, Visual Communication in Engineering, diagrams, graphs, and visual aids in technical reports and presentations, Tools for creating effective visual representations of engineering concepts.

Improvements for an existing engineering artefact, considering ethical and sustainability aspects.

TEXTBOOKS:

1. "Engineering Fundamentals: An Introduction to Engineering" by Saeed Moaveni
2. "Introduction to Engineering" by Paul H. Wright
3. "The Engineering Design Process" by Atila Ertas and Jesse C. Jones
4. "Ethics in Engineering" by Mike W. Martin and Roland Schinzinger
5. Web Based Resources and E-books:
6. "Engineering Your Future: A Brief Introduction to Engineering" by William Oakes, Les Leone, and Craig Gunn

Course Title	Emerging Technologies on Full Stack Web Development				Course Type		ETC	
Course Code	B24CSET02	Credit	1		Class		II Semester	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial				Theory	Practical	CIE	SEE
	Practical							
	Total	1	1	1	14	0	25	25

COURSE OVERVIEW:

The basics of Web application tools such as HTML, and CSS are introduced. The course also provides knowledge about advanced research topics such as JavaScript, and PHP.

COURSE OBJECTIVES:

The objectives of this course are to:

1. Explain the basic concepts of HTML code.
2. Illustrate the use of Cascading Style Sheets in web pages.
3. Demonstrate the use of Java Scripts in real world applications.
4. Describe the usage of PHP in web development.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Experiment with various HTML tags to create a web page for real world applications.	1 to 5 , 9,10,12	1
CO2	Apply Cascading Style Sheetsto design a web page for real world applications.	1 to 5 , 9,10,12	1
CO3	Develop client-side environment using JavaScripts for Web based applications.	1 to 5 , 9,10,12	2,3
CO4	Build a server-side environment using PHP for Web based application	1 to 5 , 9,10,12	2,3
CO5	Develop real time web applications.	1 to 5 , 9,10,12	1,3
O6	Make use of front end technologies and server side scripting languages to develop web applications.	1 to 5 , 9,10,12	3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	3	3	2				3	3		3	3		
CO2	3	2	2	3	2				3	3		3	3		
CO3	3	2	3	3	2				3	3		3		3	3
CO4	3	2	3	3	2				3	3		2		3	3
CO5	3	3	2	2	3				3	3		3	3		3
CO6	3	2	3	3	1				3	3		3		3	3

Note:1-Low,2-Medium,3-High

COURSE CONTENTS:

Introduction to HTML: HTML Syntax, Semantic Mark-up, Structure of HTML Documents, HTML Elements, HTML Semantic Structure Elements, HTML Web Storage. HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form concepts.

Introduction to CSS: What is CSS, CSS Syntax, Location of Styles, Selectors, and The Cascade: How Styles Interact, CSS Text Styling. Types Of CSS Layout with Examples.

Introduction to JavaScript: JavaScript: Client-Side Scripting, JavaScript's History and Uses, JavaScript Control statements, Functions, JavaScript Objects, Constructors, Validation Example for HTML form.

Introduction to PHP: Installation of XAMPP/WAMP Server, Introduction to PHP, data types, conditional statements, looping statements, functions, Arrays and Superglobals, Working with Databases, MySQL, Database APIs,. Accessing MySQL in PHP.

TEXT BOOKS:

1.Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", Pearson Education India, 1st Edition, 2016. 2.Jeffrey C. Jackson, "Web Technologies--A Computer Science Perspective", Pearson Education, 1st Edition, 2006. 3.Robert. W. Sebesta, "Programming the World Wide Web", Pearson Education, 4th Edition, 2007.

REFERENCE BOOKS:

1. Robin Nixon, "Learning PHP, MySQL &JavaScript with jQuery, CSS and HTML5", O'Reilly Publications, 4th Edition, 2015.
2. Luke Welling, Laura Thomson, "PHP and MySQL Web Development", Pearson Education, 5th Edition 2016.
3. Nicholas C Zakas, "Professional JavaScript for Web Developers", Wrox/Wiley India, 3rd Edition 2012.
4. David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 1st Edition, 2014
5. Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", Murachs/Shroff Publishers & Distributors Pvt Ltd, 3rd Edition, 2016.
6. Gerardus Blokdyk, "Representational State Transfer: Practical Integration", CreateSpace Independent Publishing Platform, 1st Edition, 2018 7. Michael Fitzgerald, 'Learning Ruby", O'Reilly, 1st Edition, 2007

JOURNALS/MAGAZINES:

1. https://www.inderscience.com/jhome.php?jcode=ijwet
2. http://stmjournals.com/Journal-of-Web-Engineering-and-Technology.html
3. https://www.scimagojr.com/journalsearch.php?q=15657&tip=sid
4. https://www.smashingmagazine.com/
5. https://www.computer.org/publications/computing-edge

SWAYAM/NPTEL/MOOCs:

1. Coursera - Web Design: Wireframes to Prototypes
2. Coursera – Web Application Technologies and Django
3. https://nptel.ac.in/courses/106/105/106105084/
4. https://www.edx.org/learn/web-development .

III SEMESTER

III Semester												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24AS	Discrete Mathematics and Graph Theory	FC	3	0	0	3	3	50	50	100	BSC
2	B24EF0301	Data Structures	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0301	Linux Operating System	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0302	Object Oriented Programming with C++	HC	3	0	0	3	3	50	50	100	PCC
5	B24EHS31X	Professional –Elective-1	SC	3	0	0	3	3	50	50	100	PEC
6	B24EF0304	Data Structures lab using C	HC	0	0	1	1	2	25	25	50	PCC
7	B24EH0303	Linux Operating Systems lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0304	Object Oriented Programming with C++ Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0305	Course-based Project (SDP) C++/Java/Prompt Engineering	SDC	0	0	1	1	2	25	25	50	Proj
10	B24CI0309	Introduction to Design Thinking	HC	1	0	1	2	3	50	50	100	ESC
11	B24	Technical Documentation	FC	1	0	0	1	1	25	25	50	HSMC
12	B24	Indian Constitution and Cyber Law	MC	1	0	0	0	1				HSMC
13	B24EF0308	AEC-1 (Ability Enhancement Course) (Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				18	0	6	23	30	450	450	900	
TOTAL SEMESTER CREDITS				23								
TOTAL CUMULATIVE CREDITS				64								
TOTAL CONTACT HOURS				30								
TOTAL MARKS				900								

Course Title	Discrete Mathematics and Graph Theory				Course Type		FC	
Course Code		Credits	3		Class		III Semester	
Course Structure	TLP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Theory	3	3	3				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	-	-	-	-				
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:

Discrete Mathematics is the study of discrete objects. Discrete Mathematics is used to develop our ability to understand and create mathematical arguments and also used to provide the mathematical foundation for advanced mathematics and computer science courses.

Graphs (abstract networks) are among the simplest mathematical structures, which are used in most of the areas of Computer Science to solve the complex problems.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain mathematical arguments using logical connectives and quantifiers.
2. Illustrate the operation on discrete structures such as sets, relations and functions.
3. Describe the theory and application of graphs, fundamental theorems and their proofs.
4. Demonstrate the use of graphs to model many types of relations and processes in physical, biological. Social and information system.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Construct mathematical arguments using logical connectives	1 ,2,3	2
CO2	Student should learn about predicate logic in mathematical statements. Student should understand the concept of logical inference and be able identify valid and invalid arguments.	1 ,2,3	2
CO3	To understand and to distinguish different types of relations and functions.	1 ,2,3	2
CO4	Student should develop the ability to use logical reasoning to prove properties of relations and functions.	1 ,2,3	2
CO5	Illustrate the concept of isomorphic graphs and isomorphism invariant properties of graphs	1 ,2,3	2
CO6	Develop a model using advanced concepts of graph for real world applications	1 ,2,3	2

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4		√				
CO5		√				
CO6			√			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	1											3	
CO2	3	2	3											3	
CO3	3	1	2											3	
CO4	3	2	3											3	
CO5	3	2	2											3	
CO6	3	2	2											3	

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Mathematical Logic: Propositions, Logical Connectives and truth tables (Illustrative Examples), Logical equivalence, Laws of logic, Duality, NAND and NOR connectives (Circuits), Converse, Inverse and Contrapositive, Rules of Inference, Quantifier, Logical implication involving quantifiers.
Applications: Design of computing machines, artificial intelligence, definition of data structures for programming languages etc

Relations and Functions : Cartesian product of sets (Illustrative Examples), Matrices and Digraph of the relations, Properties of relations, Equivalence relations, Partial ordered relations, Posets, Hasse diagrams, Extremal elements in posets, Types of Functions, The pigeon hole principle, Sterling number of second kind.
Application: Data structures, Class-based object-oriented systems, Machine learning, Databases, Pattern matching, and by extension, compilers

Introduction to graph theory: Konigsberg's bridge problems, Utilities problem, Seating Problem, Graphs, Representation of Graphs. Directed graphs, Incidence, Adjacency, Degree, In degree, Out degree, Regular graphs, Complete graphs, Null Graph, Bipartite Graphs, Isomorphism, Directed Graphs, Sub graphs, Walk, Trail, Path, Circuit, Cycle, Connected and disconnected graphs.

Applications: Finding shortest routes in car navigation systems, Search engines use ranking algorithms based on graph theory, Analysis of social networks, Compilers use coloring algorithms to assign registers to variables, Path planning in robotics.

Euler and Hamiltonian graphs and Graph coloring: Definition of Euler Graphs, Hamiltonian Graphs, Standard Theorems on Euler and Hamiltonian graphs, planar graph, Dual of planar graphs, Graph coloring, Chromatic polynomial, Five Color Theorem, Matching, , Cut set, Network flow, and its applications.
Applications: Graph coloring especially used various in research areas of science such data mining, image segmentation, clustering, image capturing, networking etc.

TEXT BOOKS:

1. Ralph P Grimaldi, "Discrete and Combinatorial Mathematics", Pearson Education, 5th Edition, 2014.
2. Nasingh Deo, "Graph Theory with Applications to Engineering Computer Science", Prentice-Hall, 2014.

REFERENCE BOOKS:

1. Keneth H Rosen, "Discrete Mathematics and its applications", 5th Edition,, Tata McGraw Hill, 2014.
2. C L Liu, "Elements of Discrete Mathematics", 4th edition, Tata MacGraw Hill 2014.
3. Thomas Khoshy, "Discrete Mathematics with applications", Elsvier, 2012.
4. Ralph P Grimaldi, "Discrete and Combinatorial Mathematics", Pearson Education, Asia, 2015.
5. Frank Harary, "Graph Theory", Norosa, 2013.
6. J. A. Bondy and V. S. R. Murthy,"Graph Theory with Applications", Macmillan, London, 2013.

JOURNALS/MAGAZINES:

1. https://www.journals.elsevier.com/discrete-mathematics
2. http://www.math.iit.edu/~kaul/Journals.html
3. https://www.siam.org/publications/journals/siam-journal-on-discrete-mathematics-sidma
4. https://onlinelibrary.wiley.com/journal/10970118
5. https://iopscience.iop.org/article/10.1088/1742-6596/1175/1/012069/meta
6. https://iopscience.iop.org/article/10.1088/1742-6596/1188/1/012065/meta
7. https://www.worldscientific.com/worldscinet/jml
8. https://www.scimagojr.com/journalsearch.php?q=12000154480&tip=sid

SWAYAM/NPTEL/MOOCs:

1. https://nptel.ac.in/courses/111/107/111107058/
2. https://nptel.ac.in/courses/106/103/106103205/
3. https://onlinecourses.swayam2.ac.in/cec20_ma02/preview
4. https://onlinecourses.nptel.ac.in/noc20_ma05/preview
5. https://onlinecourses.swayam2.ac.in/cec20_ma03/preview
6. https://www.coursera.org/learn/graphs

Course Title	Data Structures				Course Type		HC	
Course Code	B24EF0301	Credits	3		Class		III SEMESTER	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42	--	50	50

COURSE OVERVIEW:

This course is designed for undergraduate students to develop a strong foundation in the principles and practices of data structures, a fundamental aspect of computer science. This course provides an in-depth understanding of various data structures and their applications, essential for effective algorithm design and software development. By the end of this course, students will have a solid understanding of various data structures, their implementation, and applications. They will be equipped with the skills to design efficient algorithms and optimize software performance through effective use of data structures. The course combines theoretical concepts with practical coding exercises to ensure comprehensive learning and application.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Master the implementation and manipulation of fundamental data structures in C.
2. Develop proficiency in using pointers and structures for dynamic memory management.
3. Design and analyze algorithms to solve computational problems efficiently.
4. Understand and implement advanced data structures like AVL trees and heaps.
5. Enhance problem-solving skills by applying appropriate data structures to real-world problems.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	Pos	PSOs
CO1	Demonstrate proficiency in using pointers and structures for dynamic memory management and manipulation.	1 to 5,9,10,12	1
CO2	Implement and analyze various linked lists, stacks, and queues for efficient data organization.	1 to 5, 9,10,12	1,2
CO3	Apply different data structures to design and optimize algorithms for solving complex problems.	1 to 5, 9,10,12	1,2
CO4	Construct and traverse binary trees, AVL trees, and other advanced tree structures to manage hierarchical data.	1 to 5, 9,10,12	1,2
CO5	Utilize graphs and heaps in solving real-world problems involving network and priority management.	1 to 5, 9,10,12	1,2,3
CO6	Compare and contrast different data structures and algorithms to choose the most appropriate one for specific applications.	1 to 5, 9,10,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	emember(L1)	derstand(L2)	Apply(L3)	Analyze(L4)	valuate(L5)	Create(L6)
CO1			√			
CO2				√		
CO3			√			
CO4			√			
CO5			√			
CO6				√		

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	3	2				3	3		3	3	3	3
CO2	3	2	3	3	2				3	3		3	3	3	3
CO3	3	1	2	3	1				3	3		3	3	3	3
CO4	3	1	3	3	2				3	3		3	3	3	3
CO5	3	3	3	3	1				3	3		3	3	3	3
CO6	3	3	3	3	2				3	3		3	3	3	3

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Pointers in C: Declaring Pointer Variables, Pointer Expressions and Pointer Arithmetic, Null Pointers, Generic Pointers, Pointer to Pointers, Drawback of Pointers, Pointers and Arrays, Arrays of Pointers, Pointers and Two-dimensional Arrays, Pointers and Three-dimensional Arrays, Pointers and Strings.
Structures in C: Introduction, Structure Declaration, Typedef Declarations, Initialization of Structures, Accessing the Members of a Structure, Copying and Comparing Structures, Nested Structures, Arrays of Structures, Structures and Functions, Passing Individual Members, Passing the Entire Structure, Passing Structures through Pointers, Self-referential Structures.
Introduction to Data Structures: Basic Terminology, Elementary Data Structure Organization, Classification of Data

Structures, Operations on Data Structures, Abstract Data Type, Algorithms, Different Approaches to Designing an Algorithm.
Linked Lists: Introduction, Singly Linked Lists, Circular Linked Lists, Doubly Linked Lists, Circular Doubly Linked Lists, Header Linked Lists, Multi-linked Lists, Applications of Linked Lists.
Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on a Stack, Linked Representation of Stacks, Operations on a Linked Stack, Multiple Stacks, Applications of Stacks.
Queues: Introduction to Queues, Array Representation of Queues, Linked Representation of Queues, Types of Queues, Applications of Queues.
Heaps: Binary Heaps, Binomial Heaps, Fibonacci Heaps, Binomial, and Fibonacci Heaps, Applications of Heaps.
Graphs: Introduction, Graph Terminology, Directed Graphs, Bi-connected Components, Representation of Graphs.
Trees: Introduction, Types of Trees, Creating a Binary Tree from a General Tree, Traversing a Binary Tree, Huffman's Tree, Applications of Trees.
Efficient Binary Trees: Binary Search Trees, Operations on Binary Search Trees, Threaded Binary Trees, AVL Trees, Red-Black Trees, Splay Trees.

SELF-LEARNING EXERCISES:

1 Storing game entities in a array

2 Pseudo-random number generators

3 Reversing an array using a stack

4 Matching parentheses and HTML tags

5 Double ended queues

6 Application of tree traversals

TEXTBOOKS:

1. Thareja, Reema. "Data structures using C." (2014).
2. Richard Reese, Understanding and Using C Pointers, First Edition, O'Reilly, 2013.
3. Langsam, Yedidyah, Moshe J. Augenstein, and Aaron M. Tenenbaum. Data Structures using C and C++. Prentice Hall Press, 2000.
4. Seymour Lipschutz, Data Structures With C, Schaum's Outlines, McGraw Hill Education, 2017.
5. Data Structures using C, E Balagurusamy ; McGraw-Hill Education (India), 2013

REFERENCE BOOKS:

1. Richard Gilberg, Behrouz Forouzan, "DataStructures: A Pseudocode Approach with C", Cengage Learning,2004.
2. DebasisSamanta, "Classic DataStructures", second edition, PHI Learning Private Limited,2011.

JOURNALS/MAGAZINES:

1. https://www.imedpub.com/scholarly/data-structure-journals-articles-ppts-list.php
2. https://www.mdpi.com/journal/algorithms/special_issues/Efficient_Data_Structures
3. https://ieeexplore.ieee.org/document/4055607
4. https://ieeexplore.ieee.org/abstract/document/6312216
5. https://www.sciencedirect.com/science/article/pii/0022000083900065
6. https://www.sciencedirect.com/journal/journal-of-algorithms

SWAYAM/NPTEL/MOOCs:

1. Coursera – Data Structures and Algorithms Specialization
2. Coursera – Data Structures, University of California San Diego
3. Data Structures and Algorithms, National Research University Higher School of Economics
4. https://nptel.ac.in/courses/106/102/106102064/
5. https://nptel.ac.in/courses/106/106/106106127/
6. https://nptel.ac.in/courses/106/103/106103069/

Course Title	Linux Operating System				Course Type		HC	
Course Code	B24EH0301	Credits	3		Class		III SEMESTER	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:

The course "Linux Operating System" provides an in-depth exploration of Linux operating systems, tailored for students aiming to master Linux fundamentals, system administration, and advanced system programming. The curriculum covers a broad spectrum of topics, from basic commands and shell scripting to complex system programming and network management, ensuring a holistic understanding of Linux systems. This course requires a foundational knowledge of C programming. This curriculum also promises a robust and thorough education in Linux systems, equipping participants with the skills needed to excel in various technical and development roles.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Understand the foundational concepts and history of the Linux operating system.
2. Master essential Linux commands for file system navigation, file operations, and process management.
3. Develop proficiency in writing and executing shell scripts for task automation.
4. Gain the ability to manage users, groups, and software packages on a Linux system.
5. Learn to monitor system performance and analyze system logs using various tools.
6. Acquire skills in advanced Linux system programming, including system calls, process management, IPC, and network programming.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain the history and fundamentals of Linux, including different distributions and their use cases.	1,2,3,6,7,10,12	1,2,3
CO2	Demonstrate proficiency in using basic Linux commands for navigation, file operations, permissions management, system information retrieval, process management, and networking.	1,2,4,5,9,10,12	1,2,3
CO3	Construct shell scripts incorporating variables, loops, conditionals, and functions to automate routine tasks and enhance system functionality.	1,2,3,4,5,7,9,10,11,12	1,2,3
CO4	Utilize package management tools to install, update, and remove software efficiently, and manage user accounts and groups.	1,2,4,5,7,9,10,11,12	1,2,3
CO5	Make use of monitoring tools and logs to analyze system performance and identify potential issues affecting system stability and efficiency.	1,2,3,4,5,7,9,10,11,12	1,2,3
CO6	Construct networked applications using the sockets API, and employ system calls and threading techniques to handle inter-process communication and synchronization.	1,2,3,4,5,7,9,10,11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2		√				
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2	1	1	-	-	1	1	-	-	1	-	2	2	1	1
CO2	2	2	-	1	3	-	-	-	1	1	-	2	3	3	3
CO3	2	2	2	1	2	-	1	-	2	1	1	2	3	3	3
CO4	2	2	-	1	3	-	1	-	2	1	1	2	3	3	3
CO5	2	2	2	2	3	-	2	-	2	1	1	2	3	3	3
CO6	2	2	3	2	3	-	2	-	2	1	1	2	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Linux
Linux Basics - What is Linux? History of Linux; Linux distributions (Ubuntu, Fedora, CentOS, etc.)
Basic Commands - Navigation (ls, cd, pwd, mkdir, rmdir); File Operations (cp, mv, rm, touch, cat, nano/vi); Permissions (chmod, chown, chgrp); System Info (uname, top, df, du), Process Management (ps, kill, nice, cron, at); Networking (ifconfig, ping, netstat, ssh)
Shell Scripting - Writing simple scripts, Variables, loops, and conditionals, Functions and scripts interaction **System Administration**
User and Group Management - Adding and deleting users; User permissions and groups
Package Management - Using apt, yum, dnf, snap
System Monitoring and Logging - System logs (/var/log), Monitoring tools (top, htop, iotop, vmstat, dstat) **Linux System Programming - 1**
Files and Directories - System calls (open, read, write, close, lseek); Directory operations (opendir, readdir, closedir)
Process Management - Forking processes (fork, exec, wait);
Signals - Signal handling (signal, sigaction); Sending and receiving signals (kill, raise)
Threads - Creating and managing threads (pthread_create, pthread_join); Synchronization (mutexes, semaphores)
Linux System Programming - 2
IPC - pipe, message queues, shared memory, Message passing
Network Programming - Sockets API (socket, bind, listen, accept, connect); Simple client-server model
Performance Tuning and Debugging - Profiling tools (gprof, valgrind); Debugging with gdb
Self-Learning Component: Advanced Commands and Utilities - File Compression (tar, gzip, bzip2); **Networking Configuration** - Configuring network interfaces, DNS configuration, **Disk Management** - Partitioning (fdisk, gdisk), Mounting and unmounting filesystems (mount, umount), Filesystem management (mkfs, fsck)

TEXT BOOKS:

1. “The Linux Programming Interface: A Linux and UNIX System Programming Handbook”, Michael Kerrisk, 1st Edition, No Starch Press, 2010.
2. "Linux Command Line and Shell Scripting Bible", Richard Blum and Christine Bresnahan, 3rd Edition, Wiley, 2015.
3. "Linux System Administration", Tom Adelstein and Bill Lubanovic, 1st Edition, O'Reilly Media, 2007.

REFERENCE BOOKS:

1. "UNIX and Linux System Administration Handbook", Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Dan Mackin, 5th Edition, Pearson, 2017

2. "Linux System Programming: Talking Directly to the Kernel and C Library", Robert Love, 2nd Edition, O'Reilly Media, 2013

3. "Linux Bible", Christopher Negus, 9th Edition, Wiley, 2015

4. "Advanced Programming in the UNIX Environment", W. Richard Stevens, Stephen A. Rago, 3rd Edition, Addison-Wesley Professional, 2013

5. "Pro Bash Programming: Scripting the GNU/Linux Shell", Chris F.A. Johnson, Jayant Varma, 2nd Edition, Apress, 2015.

6. "Beginning Linux Programming", Neil Matthew, Richard Stones, 4th Edition, Wrox, 2007

JOURNALS/MAGAZINES:

Magazines

1. Linux Journal
2. Linux Magazine (Linux Pro Magazine)
3. Linux Format

Journals

1. The Linux Gazette
2. IEEE Transactions on Software Engineering
3. ACM Transactions on Computer Systems (TOCS)
4. Journal of Systems and Software

SWAYAM/NPTEL/MOOCs

1. Introduction to Linux – Platform NPTEL
2. Operating Systems – Platform NPTEL
3. Linux Programming & Scripting – Platform NPTEL
4. Linux Basics and Shell Programming – Platform SWAYAM
5. Linux Foundation's Introduction to Linux – Platform edX
6. Linux Server Management and Security – Platform Coursera
7. Open Source Software Development, Linux, and Git – Platform Coursera
8. Linux Command Line Basics – Platform Udacity

9. Linux for Developers – Platform Coursera

Course Title	Object-Oriented Programming with C++				Course Type		HC	
Course Code	B24EH0302	Credits	3		Class		III Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	0	0	0				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**0**	**50**	**50**

COURSE OVERVIEW:

This course introduces the basic concepts of object oriented programming. Familiarizes with object creation, and accessing members of object. OOP concepts such as data abstraction, information hiding (Encapsulation), overloading operations, inheritance mechanisms, virtual functions and friend functions are discussed. File I/O operations, Templates and exception handling mechanisms are also introduced.

COURSE OBJECTIVE (S):

The Objectives of this course are to **:**

1. Explainthebasicconceptsofobjectsandclassesforarealworldapplication.
2. Illustratetheuseofoperatoroverloading/functionoverloadingforgiven applications.
3. Demonstrate the use of inheritance in real world applications.
4. Describe I/O streams and Templates and Exception Handling.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Apply the basic concepts of objects and classes for a real world application.	1 to 5, 9 , 10	1
CO2	Utilize operator overloading / function overloading in given real world applications.	1 to 5,9,10	2
CO3	Develop programs using Inheritance feature of object oriented programming.	1 to 5, 9, 10,11	1,2
CO4	Solve programs using I/O streams, Templates and Exception Handling.	1 to 5,9,10,11	1,3
CO5	Classify inheritance based on early and late binding, exception handling and generic programming in real world application.	1 to 5, 9, 10	2,3
CO6	Explain the concept of data security using access specifier in building real world application.	1 to 5, 9 , 10	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**pply (L3)**	**nalyze (L4)**	**aluate (L5)**	**reate (L6)**
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5		√				
CO6		√				

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	3	2				2	2			3		
CO2	2	2	2	2	1				2	2				3	
CO3	3	2	3	3	1				2	2	2		3	3	
CO4	3	2	3	3	1				2	2	2		3		3
CO5	3	2	3	3	1				2	2				3	3
CO6	2	2	3	2	1				2	2				3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction: Introduction to object oriented concepts: Encapsulation, Abstraction, Inheritance, Polymorphism, Overview of OOP, Introduction to variables in C++, I/O operators, Function overloading, Inline function, Recursive function.

Classes and Objects: Introduction to Classes, Member Functions and Member data, Constructors and Destructors, Static Class members.

Operator overloading: Introduction to Objects, Array of Objects, Dynamic Objects, Pointers to objects, Friend Function, Access specifiers. The scope resolution operator, Operator overloading using Friend function: Unary operator '+' & binary operator.

Inheritance: Base Class, Inheritance and Protected members, inheriting multiple base classes, Constructors, Destructors and Inheritance, Virtual base classes. Virtual functions and Polymorphism: Virtual functions, pure virtual functions, Early and late binding.

Templates and Exception handling: Introduction to Generic Programming**,** Template Functions, Template Classes, Fundamentals of Exception handling, and handling derived class exceptions.

TEXT BOOKS:

1. HerbertSchildt,"The Complete Reference C++",4thEdition,TataMcGrawHill,2003.
2. SanleyB.Lippmann,JoseeLajore,"C++Primer",4th Edition, Pearson Education,2005.

REFERENCE BOOKS:

1. Paul J Deitel, Harvey M Deitel, "C++ for Programmers", Pearson Education, 2009.
2. K R Venugopal, Rajkumar Buyya,T Ravi Shankar,"Mastering C++",TataMcGrawHill, 1999.
3. ACM, ACM Transactions on Programming Languages and Systems (TOPLAS).
4. ACM Journal on Object-Oriented Programming

JOURNALS/MAGAZINES:

1. https://www.geeksforgeeks.org/introduction-of-object-oriented-programming/
2. https://www.geeksforgeeks.org/best-practices-of-object-oriented-programming-oop/
3. https://www.britannica.com/technology/object-oriented-programming
4. https://www.codeproject.com/Articles/22769/Introduction-to-Object-Oriented-Programming-Concept

SWAYAMNPTEL/MOOCs

1. https://www.coursera.org/specializations/object-oriented-programming
2. https://www.coursera.org/lecture/writing-running-fixing-code/introduction-to-the-programming-environment-sXizU
3. https://www.coursera.org/learn/object-oriented-design

Professional Elective 1

Course Title	Java Programming				Course Type		SC	
Course Code	B24EHS311	Credits	3		Class		III SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**0**	**50**	**50**

COURSE OVERVIEW:

Java is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is intended to let application developers write once, and run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java was first released in 1995 and is widely used for developing applications for desktop, web, and mobile devices. Java is known for its simplicity, robustness, and security features, making it a popular choice for enterprise-level applications.

COURSE OBJECTIVE (S):

The Objectives of this course are to

1. Explain the basic data types and control structures of the Java language.
2. Illustrate the creation of classes and objects in Java.
3. Demonstrate the extending a class (inheritance) and use proper program anomaly handling structures.
4. Discuss the use of Java generics and collections.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify object oriented programming features, Basic concepts required for the creating java programs, understanding the differences between JDK, JRE, and JVM.	1 to 5,9,10,12	1,2,3
CO2	Make use of concepts of inheritance and polymorphism to improve the code reusability.	1 to 5, 9,10,12	1,2,3
CO3	Develop application using packages, abstract classes, and interfaces in solving complex problems.	1 to 5, 9,10,12	1,2,3
CO4	Understanding usage of String, String constant pool and string methods ,String Builder and String Buffer concepts	1 to 5, 9 to 12	1,2,3
CO5	Understanding Exception Handling and implementing customized exception classes.	1 to 5, 9 to 12	1,2,3
CO6	Develop applications using generic methods & classes and developing applications using Collection Framework and wrapper classes.	1 to 5, 9 to 12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	3	2	3	1	3				3	1		3	3	3	3
CO2	3	2	3	1	3				3	1		3	3	3	3
CO3	3	2	3	2	3				3	2		3	3	3	3
CO4	3	2	3	2	3				3	2	2	3	3	3	3
CO5	3	2	3	2	3				3	2	2	3	3	3	3
CO6	3	2	3	2	3				3	3	3	3	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

OOP CONCEPTS& JAVA FUNDAMENTALS: Introduction to Software and types of software's, Procedural and object oriented programming paradigm, Differences between C++ and java, Classes and objects, data abstraction, encapsulation, inheritance, benefits of inheritance, polymorphism. Java programming: History of java, JDK, JRE, JVM Architecture, Java Programming environment, comments, data types, variables, constants, scope and life time of variables, operators, operator hierarchy, expressions, control flow statements, jump statements arrays, console input and output, Introducing classes, Methods and Constructors, types of constructors.

INHERITANCE, POLYMORPHISM, PACKAGES AND INTERFACES:

Inheritance: Basic concepts, Types of inheritance, Member access rules, Usage of this and Super key word, Method Overloading, Method overriding, Abstract classes, Dynamic method dispatch, Usage of final keyword. **Packages and interfaces:** Defining package, Types of packages, Access protection, importing packages, Defining and Implementing interfaces, and Extending interfaces, marker interface.

STRINGS, EXCEPTION HANDLING, INNER CLASSES:

Strings: Introduction to strings, String constant pool, strings methods, String Buffer class, String Builder class, Creating Immutable class, to String(), StringTokenizer class, **Exception Handling:** Fundamentals of exception handling, Exception types, Checked and Unchecked Exceptions, using try and catch, try with resource, multiple catch clauses, nested try statements, throw, rethrow, throws and finally, built- in exceptions, creating own

exception sub classes, chained exceptions.

Inner classes: Introduction to inner classes, Types of Inner classes.
WRAPPER CLASSES, Generics & COLLECTION FRAMEWORK:
Wrapper classes: Introduction to wrapper classes, Auto Boxing and Auto unboxing, types of wrapper classes
Generic Programming and Collections: Generic Classes; Generic Methods; Type Bounds; Type Variance and Wildcards; Restrictions on Generics;
Collections Framework: An Overview of the Collections Framework interfaces: Sets & Maps
java cursors, enum class & Iterators

TEXTBOOKS:

1. HerbertSchildt, "Java™:The Complete Reference",McGraw-Hill,Twelfth Edition,2021.
2. Understanding OOP with Java, up dated edition, T.Budd, Pearson education.

REFERENCEBOOKS:

1. Kathy Sierra and Bert Bates Head First Java, O reilly, 2nd Edition, 2020.
2. Cay S. Horstmann, "Core Java™ Volume I—Fundamentals", Prentice Hall, Tenth Edition, 2015.
3. Joshua Bloch, "Effective Java",Addison-WesleyProfessional,ThirdEdition,2017.
4. David Gallardo, Ed Burnette, Robert Mcgovern,"Eclipse in Action a guide for java developers", Manning Publications,2003.
5. Ed Burnette, "Eclipse IDE Pocket Guide: Using the Full-Featured IDE", O'Reilly Media, Inc, USA,2005.
6. Ken Kousen, "Modern Java Recipes", O'Reilly Media, Inc.,2017.
7. Oracle Java Documentation. (https://docs.oracle.com/javase/tutorial/)

JOURNALS/MAGAZINES:

1. https://www.javadevjournal.com/
2. https://blogs.oracle.com/javamagazine/
3. https://ieeexplore.ieee.org/document/5464387
4. https://files.eric.ed.gov/fulltext/EJ1075126.pdf
5. https://www.sciencedirect.com/science/article/pii/S0167642304000590
6. https://www.informingscience.org/Publications/4322?Source=%2FJournals%2FJITEIIP%2FArticles%3FVolume%3D0-0

SWAYAM/NPTEL/MOOCs:

1. https://onlinecourses.nptel.ac.in/noc19_cs84/preview
2. https://www.classcentral.com/course/swayam-programming-in-java-12930
3. https://swayam.gov.in/explorer?searchText=java

SELF-LEARNINGEXERCISES:

1. The Eclipse-IDE
2. Streams
3. Multithreading
4. JavaFX
5. Networking- JDBC

Course Title	Formal Language and Automation				Course Type		SC	
Course Code	B24EHS312		Credits		Class		III Semester	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**0**	**50**	**50**

COURSE OVERVIEW:

The course covers essential concepts in automata theory and formal languages, such as finite automata, regular expressions, formal languages, grammars, pushdown automata, and Turing machines. These serve as fundamental models of computation, forming the cornerstone of various branches of computer science.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Compare the concepts of Deterministic and Non-Deterministic Finite Automata.
2. Demonstrate the use of regular expressions for constructing DFA and NFA.
3. Illustrate the construction of context free grammar for a given language.
4. Explain computing Machine including PDA and Turing Machine.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Construct the Finite State Machine by applying the concepts of DFA andNFA.	1 to 5, 12	1
CO2	Make use of regular expressions for constructing DFA and NFA.	1 to 5,12	2
CO3	Identify ambiguity in grammar and Construct CFG for the given language in Normal Forms.	1 to 5,12	3
CO4	Apply the concepts of Push down Automata and Turing machine for a given Language.	1 to 5,12	2
CO5	Outline the notions of computation, such as algorithm, computability, decidability,reducibility, and complexity, through problem solving.	1 to 5,12	1,2
CO6	Choose mathematical foundations, algorithmic principles and computer science theory to model and design of computer based systems.	1 to 5,12	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			

CO5		√				
CO6	√					

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	1	1	1							1	3		
CO2	2	2	1	1	1							1		3	
CO3	2	2	1	1	1							1			3
CO4	2	2	1	1	1							1		3	
CO5	3	2	3	3	2								3		3
CO6	3	3	2	3	3									3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to finite automata: Introduction of Automata theory, Central concepts of Automata theory, Alphabets; Languages; strings; Deterministic and non-deterministic finite automata (with and without epsilon transitions) and their applications; Equivalence of finite automata; Minimization of Finite Automata.

Regular Expressions and Regular languages and their properties: Regular Expressions; Finite Automata and Regular Expressions; Equivalence of finite automata and regular expressions; Pumping lemma for regular languages;

Context free Grammars and Normal forms: Introduction to Context Free Grammars; Classification of Grammars; Parse Trees; Ambiguous Grammar; Normal forms – Simplifications: Left Recursion, Left Factoring, Removal of Useless Production, Elimination of Ɛ productions , Elimination of Unit-productions, Conversion to CNF and GNF.

Push Down Automata and Turing Machine: Push down automata (PDA); Working principle of PDA, Languages of a PDA; Instantaneous description of PDA, Construction of PDA, Deterministic and Non-Deterministic PDA; Turing Machine – Definition, Working principle of Turing machine; Construction of TM.

Self-Learning Component:

Applications of Finite Automata and Applications of Regular Expressions. (JFLAP tool for assignments)

TEXT BOOKS:

1. John E Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, “Introduction to Automata Theory, Languages and Computation”, 3rd Edition, Pearson Education, 2016.

2. Peter Linz, “An Introduction to formal Languages and Automata”, 4/ E, Jones and Bartlett Publishers, 2006.

REFERENCE BOOKS:

1. Kamala Krithivasan, Rama R, “Introduction to Formal Languages, Automata Theory and Computation”, Pearson, 2009.

2. B N Srinivasa Murthy, “Formal Languages and Automata Theory”, Sanguine Publishers, 2006.

JOURNALS/MAGAZINES:

1. https://theoryofcomputing.org/

2. https://www.journals.elsevier.com/theoretical-computer-science
3. https://www.springer.com/journal/224

SWAYAMNPTEL/MOOCs

1. https://www.edx.org/course/automata-theory
2. https://nptel.ac.in/courses/106/104/106104028/
3. https://ocw.mit.edu/courses/mathematics/18-404j-theory-of-computation-fall-2006/syllabus/

Course Title	Computer Networks				Course Type		SC	
Course Code	B24EHS213	Credits	3		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	-	-	-				
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:

The main emphasis of this course is on the organization and management of local area networks (LANs). The course description includes learning about computer network organization and implementation, obtaining a theoretical understanding of data communication and computer networks, and about Open Systems Interconnection (OSI) communication model with TCP/IP protocol; This course provides knowledge of error detection and recovery; local area networks; bridges, routers and gateways; network naming and addressing; and local and remote procedures. This course also emphasis on User Datagram Protocol, TCP Congestion Control; DNS Message Formatting and Remote Login Protocols.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the protocol stacks (OSI and TCP/IP) for data communication.
2. Discuss the MAC protocols, error detection & correction strategies for data transmission over the networking devices.
3. Describe the standards for data communication with routing protocols.
4. Illustrate the client server communication using TCP or UDP protocols and other application level protocols.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Make use of the architectural principles of computer networking and compare different approaches to organizing networks.	1 to 6,9,10, 12	2
CO2	Discover the good network design with simplicity, scalability, performance and the end-to-end principle	1 to 6,9,10, 12	2
CO3	Appraise the working principles of Internet.	1 to 6,9,10, 12	3
CO4	Compile the effectiveness of existing or similar network protocols.	1 to 6,9,10, 12	2, 3
CO5	Summarize the key components of the Network.	1 to 6,9,10, 12	2, 3
CO6	Demonstrate different protocols used in real world application.	1 to 6,9,10, 12	2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√	√		
CO3			√	√		
CO4			√	√		
CO5		√				
CO6		√	√	√		

COURSE ARTICULATION MATRIX

O# /	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	3	3	3	2	1			3	3		1		3	
CO 2	3	3	3	3	2	2			3	3		1		3	
CO 3	3	3	3	3	2	1			3	3		2			3
CO 4	3	3	3	2	2	1			3	3		1		3	3
CO 5	3	3	3	3	2	2			3	3		2		3	3
CO 6	3	3	3	3	2	2			3	3		2		3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Data Communication and Networking: Internet history and Internet today, Data Communications, Networks, Network Topologies, Classification of Networks, Protocols & Standards. Layered Architectures: Tasks, The OSI model, Layers in OSI model, TCP/IP Protocol suite, Addressing. Introduction to switching: Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks, Physical Layer: Introduction to Transmission Media, Periodic Analog signals, Digital signals, Nyquist bit rate, shanon capacity, performance, PCM, Parallel transmission, serial transmission.

Coding: Line Coding, Error Detection and Correction: Introduction, cyclic Codes: CRC, Internet checksum. Framing, Data Link Protocols: Point-to-Point Protocol. MAC Protocols: classification of MAC protocols, Random access (ALOHA, CSMA/CD, CSMA/CA), Controlled Access (Reservation, Polling, Token passing), Channelization Protocols (FDMA, TDMA, CDMA). Introduction to Networking Devices: Repeaters, Hubs, Bridges, Routers, and High layered switches, Gateways, Virtual LAN.

Network Layer: IPv4 addresses, IP Datagram format, ICMP Messages, Introduction to Mobile IP for mobility management, IPv6 addresses, IPv6 Packet Format, Transition from IPv4 to IPv6, Routing algorithms (Distance Vector, Link State and Path vector).

Transport Layer: Introduction to Stop and Wait, GoBack-N, Selective repeat N, Piggybacking. Services and port numbers, User Datagram Protocol (UDP): UDP Segment, Transmission Control. Protocol (TCP): TCP Segment, TCP Connection Set up, Application of TCP and UDP. TCP flow control, TCP error control, TCP Congestion Control. Application Layer: DNS, SMTP. Introduction to Remote Login Protocols: TELNET Protocol and SSH Protocol.

Self-Learning Component:
Wifi, WiMAX, 4G, 5G, Satellite Networks, MPLS, VPN, ATM, Bluetooth Architecture, World Wide Web (WWW).
TEXT BOOKS:

1. Behrouz A Forouzan, "Data Communications and Networking", 5th Edition, McGraw – Hill, 2016.
2. Nader F. Mir, "Computer and Communication Networks", Pearson Education, 2009

REFERENCE BOOKS:

1. Alberto Leon-Garcia and Indra idjaja, "Communication Networks – Fundamental Concepts and Key Architectures", 2nd Edition Tata McGraw – Hill, 2004.
2. Andrew S. Tanenbaum, "Computer Networks", 4th Edition, Pearson Education, 2005.
3. Larry L. Peterson and Bruce S. Davie, "Computer Networks- A system Approach", 5th Edition, Elsevier, 2012.
4. William Stallings, "Data and Computer Communications", 10th Edition, Pearson Education, 2008.
5. Douglas E. Comer, "Internetworking with TCP/IP Vol.1", 6th Edition, Pearson, 1995.

JOURNALS/MAGAZINES

1. IEEE Transactions on Networking.
2. Elsevier Journal of Computer Networks
3. Springer Journal of communications and Information networks.

SWAYAM/NPTEL/MOOCs:

1. https://www.udemy.com/topic/computer-network/
2. https://www.coursera.org/courses?query=computer%20network
3. https://nptel.ac.in/courses/106/105/106105183/
4. https://www.edx.org/learn/computer-networking

Course Title	Data Structures lab using C				Course Type		HC	
Course Code	B24EF0304	Credits	1		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

This course, "Data Structures Using C," provides a comprehensive introduction to fundamental data structures and their applications, using the C programming language as the primary tool for implementation. Designed for students with basic programming knowledge, the course aims to equip them with the skills needed to efficiently store, manage, and manipulate data. Through a series of hands-on experiments and mini-projects, students will explore a variety of data structures, including linked lists, stacks, queues, trees, heaps, and graphs. They will learn to implement these structures, understand their underlying principles, and apply them to solve complex problems. Overall, this course not only teaches the technical aspects of data structures but also encourages analytical thinking and effective programming practices, preparing students for advanced studies and careers in computer science and software engineering.

COURSE OBJECTIVES:

The objectives of this course are to

1. Master fundamental C programming concepts such as pointers, dynamic memory allocation, and structures.
2. Gain a deep understanding of various data structures, including arrays, linked lists (singly, doubly, and circular), stacks, and queues.
3. Learn to implement and manipulate advanced tree structures like Binary Search Trees (BSTs), AVL trees, Red-Black trees, Splay trees, and Threaded Binary trees.
4. Implement and analyze algorithms for managing priority queues using binary heaps, and perform file compression using Huffman coding.
5. Utilize learned data structures to design and develop real-world applications such as a student enrollment system, library management system, music playlist management, and text editor auto-complete features.
6. Undertake mini-projects that synthesize course concepts to solve complex problems, enhancing problem-solving and project management skills.

COURSE OUTCOMES:

On successful completion of this course; the student will be able to

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate proficiency in advanced C programming concepts, including pointers, dynamic memory allocation, and structures.	1, 5, 8, 12	1, 3
CO2	Implement and manipulate essential data structures such as arrays, linked lists, stacks, and queues in C.	1, 2, 4, 5, 9	1, 2, 3
CO3	Develop and apply advanced tree and graph structures, including various trees (BST, AVL, Red-Black) and graph traversal algorithms (BFS, DFS).	1, 2, 4, 5, 8	1, 2, 3

CO4	Design and implement algorithms for managing data efficiently, such as priority queues, Huffman coding, and database indexing.	1, 2, 4, 5, 11	1, 2, 3
CO5	Create practical applications, leveraging learned data structures to solve real-world problems in domains like enrollment systems, library management, and more.	1, 2, 4, 5, 6, 11	1, 2, 3
CO6	Enhance the problem-solving and project management abilities through mini-projects that integrate course concepts to address complex challenges.	1, 4, 9, 11, 12	1, 2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	ember(L1)	erstand(L2)	pply(L3)	halyze(L4)	aluate(L5)	reate(L6)
CO1	✓	✓				
CO2	✓	✓	✓			
CO3	✓	✓	✓			
CO4	✓	✓	✓			
CO5	✓	✓	✓			
CO6	✓	✓	✓			

COURSE ARTICULATION MATRIX

CO #/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	2	3	2				3	3		3	3	3	3
CO2	3	2	3	3	2				3	3		3	3	3	3
CO3	3	1	2	3	1				3	3		3	3	3	3
CO4	3	1	3	3	2				3	3		3	3	3	3
CO5	3	3	3	3	1				3	3		3	3	3	3
CO6			3	2	3				2			3	3	3	3

Note:1-Low,2-Medium,3-High **PRACTICE:**

No	Title of the Experiment	Tools and Techniques	Expected Skill / Ability
C Programming (Practice Experiments 1 to 5 - Not to be Examined)			
Experiment 1: Implementation of Pointer Variables and Pointer Arithmetic			
a	**Swapping Two Numbers Using Pointers** Write a C program to swap two integers using pointers. The program should prompt the user to enter two numbers and then display the numbers before and after swapping.	C programming language, Pointer manipulation	Pointers
b	**Calculating the Average of Array Elements Using Pointers** Write a C program to calculate the average of an array of integers	C programming language,	Dynamic memory

	using pointers. The program should dynamically allocate memory for an array, accept input from the user, and then compute and display the average using pointer arithmetic.	dynamic memory allocation	allocation
Experiment 2: Working with Null Pointers, Generic Pointers, Pointers to Pointers, and Pointer Arrays			
a	**Handling Null Pointers** Write a C program that demonstrates the use of null pointers. The program should attempt to dereference a null pointer and handle the resulting error gracefully by checking for null before dereferencing.	C programming language, Null pointer handling, error checking	Error handling, and conditional checks
b	**Using Generic Pointers (void pointers)** Write a C program that uses a generic pointer (void pointer) to store and print different types of data (int, float, char). The program should use type casting to handle the void pointer appropriately.	C programming language, Void pointers, type casting	Working with different data types using void pointers
c	**Pointers to Pointers** Write a C program that demonstrates the use of pointers to pointers by dynamically allocating a two-dimensional array using pointers to pointers. The program should accept input to fill the array and then display the array.	C programming language, dynamic memory allocation, Pointers to pointers, two-dimensional arrays	Working with multi-dimensional arrays using pointers.
d	**Array of Pointers to Strings** Write a C program that creates an array of pointers to strings (character arrays). The program should store a list of strings (names), allow the user to input names, and then display the list of names using pointer notation.	C programming language, string manipulation, Array of pointers, string handling	String manipulation, and pointer notation for accessing elements
Experiment 3: Implementing and Manipulating Two-dimensional and Three-dimensional Arrays Using Pointers			
a	**Two-dimensional Array Using Pointers** Write a C program that dynamically allocates memory for a two-dimensional array using pointers. The program should prompt the user to enter the number of rows and columns, accept input to fill the array, and then display the array.	C programming language, dynamic memory allocation, Pointers, dynamic memory management, array manipulation	Dynamic memory allocation for two-dimensional arrays
b	**Matrix Multiplication Using Pointers** Write a C program to perform matrix multiplication using dynamically allocated two-dimensional arrays. The program should prompt the user for the dimensions and elements of two matrices, perform the multiplication, and display the resulting matrix.	C programming language, dynamic memory allocation, Pointers, dynamic memory management, matrix manipulation	Dynamic memory allocation for matrices
c	**Three-dimensional Array Using Pointers** Write a C program that dynamically allocates memory for a three-dimensional array using pointers. The program should prompt the user to enter the dimensions, accept input to fill the array, and	C programming language, dynamic memory	Dynamic memory allocation for three-

	then display the array.	allocation, Pointers, dynamic memory management, array manipulation	dimensional arrays
Experiment 4: Creating and Using Structures			
a	**Employee Management System** Write a C program to create a structure called Employee with fields for employee ID, name, department, and salary. Use an array of structures to store information for multiple employees. The program should include functions to: Accept details for multiple employees. Display details of all employees. Modify the salary of an employee given their ID.	C programming language, structures, arrays, Structure manipulation, array handling, input/output	Structure definition, array of structures, accessing and modifying structure members.
b	**Library System with Nested Structures** Write a C program to create a structure called Book with fields for book ID, title, author, and a nested structure Date (with fields for day, month, and year) to store the publication date. Use an array of Book structures to manage a collection of books. The program should: Accept details for multiple books. Display details of all books. Find and display books published in a particular year.	C programming language, structures, arrays, Nested structures, array manipulation, search algorithms	Nested structures, array of structures, accessing nested structure members.
c	**Self-Referential Structures for Linked List** Write a C program to implement a singly linked list using a self-referential structure Node with fields for an integer data and a pointer to the next node. The program should include functions to: Insert a node at the beginning. Insert a node at the end. Display all nodes in the linked list.	C programming language, structures, pointers, Linked list implementation, pointer manipulation, memory management	Self-referential structures, dynamic memory allocation, linked list operations.
d	**Student Records System** Write a C program to create a structure called Student with fields for student ID, name, and an array of marks for 5 subjects. Use an array of structures to store records for multiple students. The program should: Accept details for multiple students. Calculate and display the average marks for each student. Find and display the student with the highest average marks.	C programming language, structures, arrays, Array of structures, input/output, data manipulation	Arrays within structures, array of structures, accessing and modifying structure members.
Experiment 5: Passing Structures to Functions			
a	**Bank Account Management** Write a C program to create a structure called BankAccount with fields for account number, account holder's name, and balance. Implement functions to: Accept account details (by value). Display account details (by value). Deposit money (by reference). Withdraw money (by reference).	C programming language, structures, functions, Structure manipulation, input/output, pass by value/reference	Passing structures by value and by reference, accessing structure members within functions.
b	**Comparing Dates** Write a C program to create a structure called Date with fields for day, month, and year. Implement functions to: Accept a date (by value).	C programming language, structures, functions,	Passing structures by value, structure comparison,

	Display a date (by value). Compare two dates (by value) and return -1, 0, or 1 if the first date is earlier than, equal to, or later than the second date respectively.	Structure manipulation, input/output, conditional statements	returning values from functions.
c	**Copying and Comparing Student Records** Write a C program to create a structure called Student with fields for student ID, name, and an array of marks for 5 subjects. Implement functions to: Accept student details (by reference). Copy the details of one student to another (by reference). Compare two students based on their total marks (by reference) and return the student with higher marks.	C programming language, structures, functions, Structure manipulation, array handling, conditional statements	Passing structures by reference, structure copying, structure comparison.
d	**Rectangle Operations** Write a C program to create a structure called Rectangle with fields for length and width. Implement functions to: Accept rectangle dimensions (by value). Display rectangle dimensions (by value). Calculate and return the area of the rectangle (by value). Calculate and return the perimeter of the rectangle (by reference).	C programming language, structures, functions, Structure manipulation, input/output, pass by value/reference, mathematical operations	Passing structures by value and by reference, returning values from functions.
PART - A Data Structures (Experiments 6 to 17 – To be Examined)			
Experiment 6: Implementing Singly Linked Lists			
a	**Student Enrollment System** Write a C program to manage student enrollments using a singly linked list. Each node in the list should represent a student with fields for student ID, name, and age. The program should include functions to: Create the linked list by adding new students. Traverse and display the list of students. Insert a new student at a specific position. Delete a student by their ID.	C programming language, singly linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Node creation, linked list traversal, insertion at a specific position, deletion by key.
b	**To-Do List Management** Write a C program to manage a to-do list using a singly linked list. Each node should represent a task with fields for task ID, description, and priority. The program should include functions to: Add a new task to the list. Display all tasks in the list. Insert a task at the beginning of the list. Delete a task from the end of the list.	C programming language, singly linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Node creation, insertion at the beginning, deletion from the end, traversal.
c	**Library Book Inventory** Write a C program to manage a library's book inventory using a singly linked list. Each node should represent a book with fields for book ID, title, and author. The program should include functions to: Add a new book to the inventory. Traverse and display the list of books. Insert a book at the end of the list. Delete a book by its title.	C programming language, singly linked list, structures, Linked list manipulation, dynamic memory allocation,	Node creation, traversal, insertion at the end, deletion by key.

		input/output	
Experiment 7: Implementing Doubly Linked Lists and Circular Linked Lists			
a	**Music Playlist Management (Doubly Linked List)** Write a C program to manage a music playlist using a doubly linked list. Each node should represent a song with fields for song ID, title, and artist. The program should include functions to: Create the playlist by adding new songs. Traverse and display the playlist from start to end and from end to start. Insert a new song at a specific position. Delete a song by its ID.	C programming language, doubly linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Node creation, bidirectional traversal, insertion at a specific position, deletion by key.
b	**Circular Queue of Customers (Circular Linked List)** Write a C program to manage a queue of customers using a circular linked list. Each node should represent a customer with fields for customer ID, name, and service type. The program should include functions to: Add a new customer to the queue. Display the queue of customers. Insert a customer at the beginning of the queue. Delete a customer from the end of the queue.	C programming language, circular linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Circular structure, traversal, insertion at the beginning, deletion from the end.
c	**Train Compartment Management (Circular Doubly Linked List)** Write a C program to manage train compartments using a circular doubly linked list. Each node should represent a compartment with fields for compartment number, capacity, and occupancy. The program should include functions to: Add a new compartment to the train. Traverse and display the list of compartments in both directions. Insert a new compartment at a specific position. Delete a compartment by its number.	C programming language, circular doubly linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Circular doubly linked structure, bidirectional traversal, insertion at a specific position, deletion by key.
d	**Double-Ended Queue (Doubly Linked List)** Write a C program to implement a double-ended queue (deque) using a doubly linked list. Each node should represent an element with an integer value. The program should include functions to: Add an element at the front of the deque. Add an element at the rear of the deque. Remove an element from the front of the deque. Remove an element from the rear of the deque. Display the elements in the deque from front to rear and rear to front.	C programming language, doubly linked list, structures, Linked list manipulation, dynamic memory allocation, input/output	Node creation, insertion and deletion at both ends, bidirectional traversal.
Experiment 8: Implementing Stack Operations and Applications			
a	**Web Browser History (Array-based Stack)** Write a C program to simulate the back button functionality in a web browser using a stack implemented with arrays. Each entry in the stack should represent a URL visited by the user. The program should include functions to: Push a new URL onto the stack when a new page is visited. Pop the top URL from the stack when the back button is pressed. Peek at the current URL without removing it from the stack. Display the current stack of URLs.	C programming language, array-based stack, functions, Stack implementation using arrays, push and pop operations, input/output	Stack operations (push, pop, peek), array implementation, simulation of real-world application.

b	**Expression Evaluation (Linked List-based Stack)** Write a C program to evaluate a postfix expression using a stack implemented with linked lists. The program should: Read a postfix expression from the user.	C programming language, linked list-based stack, functions, Stack	Stack operations (push, pop), linked list implementation,
	Use stack operations to evaluate the expression. Display the result of the evaluation.	implementation using linked lists, postfix expression evaluation, input/output	postfix expression evaluation.
c	**Infix to Postfix Conversion** Write a C program to convert an infix expression to a postfix expression using a stack implemented with arrays. The program should: Read an infix expression from the user. Use stack operations to convert the infix expression to a postfix expression. Display the converted postfix expression.	C programming language, array-based stack, functions, Stack implementation using arrays, infix to postfix conversion algorithm, input/output	Stack operations (push, pop, peek), precedence and associativity of operators, infix to postfix conversion algorithm.
Experiment 9: Implementing Queue Operations			
a	**Customer Service Queue (Array-based Queue)** Write a C program to manage a customer service queue using a queue implemented with arrays. Each entry in the queue should represent a customer with fields for customer ID and name. The program should include functions to: Enqueue a new customer. Dequeue a customer when they are served. Display the current queue of customers.	C programming language, array-based queue, functions, Queue implementation using arrays, enqueue and dequeue operations, input/output	Queue operations (enqueue, dequeue), array implementation, simulation of customer service.
b	**Printer Job Queue (Linked List-based Queue)** Write a C program to manage a printer job queue using a queue implemented with linked lists. Each entry in the queue should represent a print job with fields for job ID and number of pages. The program should include functions to: Enqueue a new print job. Dequeue a print job when it is completed. Display the current queue of print jobs.	C programming language, linked list-based queue, functions, Queue implementation using linked lists, enqueue and dequeue operations, input/output	Queue operations (enqueue, dequeue), linked list implementation, simulation of a print queue.
c	**Circular Queue for Ticket Booking** Write a C program to manage a circular queue for booking tickets using an array. Each entry in the queue should represent a ticket request with fields for request ID and number of tickets. The program should include functions to: Enqueue a new ticket request. Dequeue a ticket request when it is processed. Display the current queue of ticket requests.	C programming language, circular queue, functions, Queue implementation using circular arrays, enqueue and dequeue operations, input/output	Circular queue operations (enqueue, dequeue), array implementation, circular buffer concept.

d	**Priority Queue for Task Scheduling** Write a C program to manage a priority queue for task scheduling using a linked list. Each entry in the queue should represent a task with fields for task ID, task description, and priority level. The program should include functions to: Enqueue a new task based on its priority.	C programming language, priority queue, linked list, functions, Priority queue	Priority queue operations (enqueue with priority, dequeue), linked list
	Dequeue the highest priority task. Display the current queue of tasks.	implementation using linked lists, enqueue and dequeue operations based on priority, input/output	implementation, task scheduling based on priority.
Experiment 10: Implementing Heaps			
a	**Priority Queue for Task Management (Binary Heap)** Write a C program to manage a priority queue for task management using a binary heap. Each task should have a priority level and a description. The program should include functions to: Insert a new task into the heap. Delete the highest priority task from the heap. Display the current tasks in the heap.	C programming language, binary heap data structure, Priority queue implementation,	Binary heap properties, insertion (heapify-up), deletion (heapify-down), priority queue management.
b	**Comparing Binary Heaps, Binomial Heaps, and Fibonacci Heaps** Write a C program to compare the performance of binary heaps, binomial heaps, and Fibonacci heaps. The program should: Implement insertion and deletion operations for all three types of heaps. Measure the time taken for a series of random insertions and deletions in each type of heap. Display and compare the results.	binary heap operations (insertion, deletion), memory management	Understanding different heap structures, implementing heap operations, performance analysis.
Experiment 11: Implementing Graphs			
a	**Social Network Graph (Adjacency Matrix)** Write a C program to represent a social network using an adjacency matrix. Each node represents a user, and an edge represents a friendship. The program should include functions to: Add a new user. Add a friendship between two users. Display the adjacency matrix. Perform BFS to find the shortest path between two users.	C programming language, adjacency matrix representation, Matrix operations, graph representation, Breadth-First Search (BFS), dynamic memory allocation	Graph representation using adjacency matrix, graph traversal (BFS), shortest path finding.

b	**Road Network Graph (Adjacency List)** Write a C program to represent a road network using an adjacency list. Each node represents a city, and an edge represents a road between cities with a certain distance. The program should include functions to: Add a new city. Add a road between two cities. Display the adjacency list. Perform DFS to find all cities reachable from a given city.	C programming language, adjacency list representation, Linked list operations, graph representation, Depth-First Search (DFS), dynamic memory allocation	Graph representation using adjacency list, graph traversal (DFS), reachability in graphs.
c	**Graph Traversal Algorithms** Write a C program to implement basic graph traversal algorithms (BFS and DFS) on a given graph. The program should:	C programming language, graph representation	BFS and DFS algorithms, graph traversal
	Allow the user to input a graph. Perform BFS starting from a specified node and display the traversal order. Perform DFS starting from a specified node and display the traversal order.	(adjacency matrix or list), Graph input handling, Breadth-First Search (BFS), Depth-First Search (DFS), traversal algorithms, dynamic memory allocation	techniques, user input handling.
Experiment 12: Binary Search Trees (BST)			
	Implementing a Binary Search Tree Write a C program to implement a Binary Search Tree (BST). The program should include functions to: Insert a new node. Delete a node. Search for a node. Traverse the tree in-order, pre-order, and post-order. Find the minimum and maximum value nodes.	C programming language, pointers, structs, Tree data structure implementation Binary search tree operations (insertion, deletion, search)	BST properties, insertion, deletion, search, tree traversal, minimum and maximum node retrieval.
Experiment 13: Operations on Binary Search Trees			
	Advanced BST Operations Write a C program to perform the following operations on a BST: Determine the height of the tree. Count the number of nodes in the tree. Find the mirror image of the tree. Check if the tree is balanced.	C programming language, pointers, structs, Binary Search Tree (BST) operations	Tree height calculation, node count, tree mirroring, balance checking.
Experiment 14: Threaded Binary Trees			

	Implementing a Threaded Binary Tree Write a C program to implement a Threaded Binary Tree. The program should include functions to: Insert nodes in the threaded binary tree. Perform in-order traversal using threads. Perform pre-order and post-order traversals without recursion or stack.	C programming language, pointers, structs, Threaded Binary Tree implementation	Threaded binary tree structure, threaded in-order traversal, non-recursive traversals.
Experiment 15: AVL Trees			
	Implementing an AVL Tree Write a C program to implement an AVL Tree. The program should include functions to: Insert nodes and maintain balance using rotations (single and double). Delete nodes and maintain balance. Perform in-order, pre-order, and post-order traversals. Check the balance factor of each node.	C programming language, pointers, structs, AVL Tree implementation Node insertion with balancing (single and double rotations: left, right, left-right, right-left)	AVL tree properties, rotations (left, right, left-right, right-left), balanced tree maintenance, balance factor calculation.
Experiment 16: Red-Black Trees			
	Implementing a Red-Black Tree Write a C program to implement a Red-Black Tree. The program should include functions to: Insert nodes and maintain Red-Black properties. Delete nodes and maintain Red-Black properties. Perform in-order, pre-order, and post-order traversals. Verify the properties of the Red-Black Tree.	C programming language, pointers, structs, Red-Black Tree implementation	Red-Black Tree properties, color handling, rotations, insertion and deletion with property maintenance.
Experiment 17: Splay Trees			
	Implementing a Splay Tree Write a C program to implement a Splay Tree. The program should include functions to: Insert nodes into the splay tree. Delete nodes from the splay tree. Search for nodes and splay them to the root. Perform in-order, pre-order, and post-order traversals.	C programming language, pointers, structs, Splay Tree implementation	Splay tree operations, splaying mechanism, self-adjusting properties, tree traversal.
PART - B **Data Structures (Mini-Project – To be Examined)** **Note: Each student shall be assigned any one of the following mini-projects**			
1	**Priority Queue Management System using Binary Heap**: Implement a priority queue management system using a binary heap. Users can enqueue tasks with priorities and dequeue them according to their priorities. Operations: Insertion (enqueue), Deletion (dequeue), Peek (retrieve highest priority task).	C programming language, binary heap data structure, Priority queue implementation, binary heap operations (insertion, deletion), memory management	

2	**Shortest Path Finding using Dijkstra's Algorithm with Fibonacci Heap**: Implement Dijkstra's shortest path algorithm using a Fibonacci heap for efficient priority queue management. Input: Weighted directed graph. Output: Shortest path from a source vertex to all other vertices.	C programming language, graph data structure, Fibonacci heap, Dijkstra's shortest path algorithm, graph traversal, Fibonacci heap operations
3	**File Compression using Huffman Coding**: Implement file compression using Huffman coding with a binary tree. Input: Text file. Output: Compressed file and Huffman tree.	C programming language, binary tree data structure, Huffman coding algorithm, binary tree traversal, file I/O
4	**Social Network Analysis using Graphs**: Create a social network analysis tool using graph data structures. Features: Add users as vertices, add friendships as edges, find mutual friends, suggest friends based on common connections, identify communities using graph algorithms.	C programming language, graph data structure, Graph manipulation, community detection algorithms, friend suggestion algorithms
5	**Expression Parsing using Binary Trees**: Implement an expression parser using binary trees to represent arithmetic expressions. Features: Parse infix expressions into binary expression trees, evaluate expressions, and perform basic arithmetic operations.	C programming language, binary tree data structure, Expression parsing, infix to binary expression tree conversion, arithmetic operations
6	**Database Indexing using AVL Trees**: Create a database indexing system using AVL trees to efficiently store and retrieve data.	C programming language, AVL tree data structure, Indexing, balanced tree operations, database operations
	Features: Insertion of records, deletion of records, searching records, maintaining balanced AVL trees for efficient indexing.	
7	**Text Editor Auto-complete using Trie**: Develop a text editor auto-complete feature using a trie data structure. Features: Insertion of words, searching for words with prefix, suggesting auto-complete options based on user input.	C programming language, trie data structure, Text processing, auto- complete feature, trie operations
8	**Symbol Table Implementation using Red-Black Trees**: Implement a symbol table for a compiler using red-black trees. Features: Insertion of identifiers, searching for identifiers, maintaining balanced red-black trees for efficient symbol lookup.	C programming language, red-black tree data structure, Symbol table implementation, balanced tree operations, compiler optimization
9	**Cache Optimization using Splay Trees**: Implement a cache optimization algorithm using splay trees. Features: Cache storage, access tracking, splay tree operations to promote frequently accessed items for faster retrieval.	C programming language, splay tree data structure, Cache optimization, access tracking, splay tree operations
10	**Network Routing using Bi-connected Components**: Develop a network routing algorithm using bi-connected components in graphs. Features: Identifying critical nodes and edges, ensuring redundancy in network paths, handling network failures.	C programming language, graph data structure, Network routing algorithms, bi-connected components, redundancy handling

TEXTBOOKS:

1. Thareja, Reema. "Data structures using C." (2014).
2. Richard Reese, Understanding and Using C Pointers, First Edition, O'Reilly, 2013.
3. Langsam, Yedidyah, Moshe J. Augenstein, and Aaron M. Tenenbaum. Data Structures using C and C++. Prentice Hall Press, 2000.
4. Seymour Lipschutz, Data Structures With C, Schaum's Outlines, McGraw Hill Education, 2017.
5. Data Structures using C, E Balagurusamy ; McGraw-Hill Education (India), 2013

REFERENCE BOOKS:

1. Richard Gilberg, Behrouz Forouzan, “DataStructures: A Pseudocode Approach with C”, Cengage Learning,2004.
2. DebasisSamanta, “Classic DataStructures”, second edition, PHI Learning Private Limited,2011.

JOURNALS/MAGAZINES:

1. https://www.imedpub.com/scholarly/data-structure-journals-articles-ppts-list.php
2. https://www.mdpi.com/journal/algorithms/special_issues/Efficient_Data_Structures
3. https://ieeexplore.ieee.org/document/4055607
4. https://ieeexplore.ieee.org/abstract/document/6312216
5. https://www.sciencedirect.com/science/article/pii/0022000083900065
6. https://www.sciencedirect.com/journal/journal-of-algorithms

SWAYAM/NPTEL/MOOCs:

1. Coursera – Data Structures and Algorithms Specialization
2. Coursera – Data Structures, University of California San Diego
3. Data Structures and Algorithms, National Research University Higher School of Economics
4. https://nptel.ac.in/courses/106/102/106102064/
5. https://nptel.ac.in/courses/106/106/106106127/
6. https://nptel.ac.in/courses/106/103/106103069/

Course Title	Linux Operating Systems Lab				Course Type		HC	
Course Code	B24EH0303	Credits	1		Class		III Semester	
Course Per Semester Structure	LTP	Credits	Contact	Workload	Total Number of Classes		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	**1**	**2**	**2**	**-**	**28**	**25**	**25**

COURSE DESCRIPTION:

This Linux Operating System Lab is designed to provide students with hands-on experience in navigating, managing, and utilizing Linux systems effectively. Through a series of practical experiments, students will learn how to navigate the file system, perform file operations, manage system processes, configure network settings, write shell scripts, handle system administration tasks, and develop programs using C that interact with the operating system. By the end of the course, students will have a solid foundation in using Linux for various administrative and development tasks.

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Navigate and manipulate the Linux file system using basic commands and directory structures.
2. Perform file operations and manage permissions to control access and ownership of files and directories.
3. Retrieve and analyze system information and processes using system monitoring commands and tools.
4. Configure network settings and test connectivity to ensure proper network functionality.
5. Develop and execute scripts to automate tasks, manage files, and interact with users.
6. Implement C programs to interact with the operating system, demonstrating process creation, inter-process communication, and network programming.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate the use of system calls to perform file operations, including opening, reading, and writing, navigating using lseek, and closing files in a C program.	1,2,3,4,5,10,12	1,2,3
CO2	Illustrate directory management by using opendir, readdir, and closedir system calls to list file names within a directory.	1,2,3,4,5,10,12	1,2,3
CO3	Construct a parent-child process relationship using fork, where the parent and child processes correctly print their respective PIDs, and utilize exec to execute a different program within the child process.	1,2,3,4,5,9,10,11,12	1,2,3
C04	Apply inter-process communication mechanisms such as pipes, message queues, and shared memory to enable data exchange between parent and child processes in C programs.	1,2,3,4,5,9,10,11,12	1,2,3
CO5	Develop signal handling in C programs by setting up custom signal handlers using signal or sigaction, and illustrate signal communication between processes using kill.	1,2,3,4,5,8,9,10,11,12	1,2,3
CO6	Make use of gprof to evaluate the performance of computationally intensive tasks in C programs, and analyze debugging issues using gdb to identify and resolve intentional bugs.	1,2,3,4,5,10,11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1		√				
CO2		√				
CO3			√			
C04			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	2	1	3	-	-	-	-	2	-	1	3	2	3
CO2	2	2	1	1	3	-	-	-	-	2	-	1	2	1	3
CO3	3	3	3	2	3	-	-	-	2	3	2	2	3	3	3
CO4	3	3	2	2	3	-	-	-	2	2	2	2	3	3	3
CO5	3	2	2	1	3	-	-	3	2	3	1	1	2	2	3
CO6	3	3	2	3	3	-	-	-	-	2	2	2	3	2	3

Note: 1-Low, 2-Medium, 3-High

Practice:

No	**Title of the Experiment**	**Tools and Techniques**	**Expected Skill/Ability**
Note: Experiments 1 to 6 are Introductory to allow students to have operational knowledge about Linux Operating System Students need to practice them as self-study on their Laptops, The experiments must be monitored by Lab faculty members and shall not be included in examinations. However they must be included in Lab records.			
1	Experiment the following Linux commands and Use the man command to learn about the options and detailed usage of each command. Navigation: I. Use ls, cd, pwd to navigate the file system. II. Create a directory structure: /home/user/lab1/subdir1/subdir2 using mkdir -p. **File** Operations: I. Create files using touch and use nano or vi to edit them.	Command Line Interface (CLI), Manual Pages (man)	Applying Fundamental Linux Commands

	II. Copy (cp), move (mv), and remove (rm) files within the created directory structure in (a) Permissions: I. Change file and directory permissions using chmod. II. Modify ownership using chown and chgrp. System Info: I. Retrieve system information using uname, top, df, and du. Process Management: I. List processes using ps, terminate a process with kill. II. Adjust process priority with nice and schedule a task with cron and at. Networking: I. Display network configuration using ifconfig. II. Test connectivity with ping. III. View network connections with netstat. IV. Securely connect to another system using ssh.		
2	(a) Create a script to display "Hello, World!" (b) Write a script to list all files in a directory and save the output to a file. (c) Develop a script to back up a directory (e.g., /home/user/lab1) to another location.	Shell scripting	Basic shell scripting skills.
3	(a) Write a script that takes a user input and prints a customized greeting message. (b) Create a script that iterates over a list of files and directories, displaying their names. (c) Develop a script that checks if a given directory exists and creates it if it does not.	Shell scripting	Basic shell scripting skills.
4	(a) Write a script that defines a function to calculate the factorial of a number. (b) Create a script that uses the function to display the factorial of numbers from 1 to 5. (c) Develop a script that calls another script and passes arguments to it.	Shell scripting	Basic shell scripting skills.

5.	(a) Create a new user named "student1" and assign them to a new group called "students". Set a password for "student1" and ensure it is part of the "sudo" group. (b) Create another user "student2" and add them to the "students" group. Verify that both "student1" and "student2" have appropriate permissions by creating a shared directory and setting the correct group ownership and permissions to allow group members to read and write. (c) Delete the user "student2" and ensure all their files are also deleted.	Linux commands	User and group management, Sudo configuration
6	(a) Install a new package called "htop" using both apt (for Debian-based systems) and yum or dnf (for Red Hat-based systems). Verify the installation and check the version of the installed package. Also, install a package using snap and verify its installation. (b) Use "htop" to monitor system processes and identify the top three processes consuming the most CPU and memory. Use "vmstat" to gather system performance statistics and save the output to a file for analysis. (c) Uninstall the package “htop” from the system. (d) Check the system logs in /var/log, specifically syslog and auth.log. Identify any failed login attempts and note the timestamps. Use "dstat" to monitor real-time system performance and generate a report.	Package Managers, Monitoring Tools	Package installation, System monitoring and logging
Note: Experiment 7 to 18 must be carried out in the Lab and Must be examined			
7	Write a C program to open a file, write some text, read it back, and then close the file using system calls (open, read, write, close). Also, use lseek to navigate through the file.	Linux System Calls	File handling
8	Implement a C program that opens a directory, reads all the file names in it, and displays them. Use the opendir, readdir, and closedir system calls.	Linux System Calls	Directory management
9	Write a C program that demonstrates process creation using fork. The parent process should print its PID and the child process should print its own PID and its parent’s PID.	Linux System Calls	Process creation
10	Create a C program where a child process, created using fork, executes a different program using one of the exec family of functions.	Linux System Calls	Program execution
11	Develop a C program that uses fork to create a child process and uses wait in the parent process to ensure it waits until the child process finishes execution.	Linux System Calls	Process synchronization
12	Write a C program that sets up a custom signal handler for SIGINT using signal or sigaction. The handler should print a message when the signal is received.	Linux System Calls	Signal handling
13	Create two C programs where one sends a signal to the other using kill and the receiver handles it appropriately.	Linux System Calls	Inter-process communication
14	Write a C program to create a pipe between a parent and child process. The parent should send a string to the child through the pipe, and the child should read and print it.	Pipe	Inter-process communication
15	Implement a C program using System V message queues. The program should send and receive messages between two processes.	Message queues	Inter-process communication
16	Develop a C program where two processes share a memory segment. One process should write data to the segment, and the other process should read and print it.	Shared memory	Inter-process communication
17	Create a simple client-server application using TCP sockets in C. The server should accept connections and echo back any message sent by the client.	TCP sockets	Network communication
18	Write a C program for a simple client-server model using UDP sockets. The client should send a message to the server, and the server should respond back.	UDP sockets	Datagram communication

Note: Experiments 19 and 20 are additional experiments for performance and debugging not be included in exams. However they must be conducted and executed and written in Lab records.			
19	Develop a C program that performs a computationally intensive task. Use gprof to profile the program and identify the functions consuming the most time.	gprof	Performance profiling
20	Write a C program with intentional bugs (e.g., segmentation fault, logical errors). Use gdb to debug and find the issues in the program.	gdb	Debugging

Project

No	Title of the Experiment	Tools and Technics	Expected Skill/Ability
1	Develop a multi-threaded file server in C that handles multiple client requests concurrently. Use pthread_create to manage threads, and employ mutexes or semaphores for synchronization. The server should be able to handle basic file operations (read, write) from clients.	Threads, mutexes, semaphores	Concurrency, synchronization
2	Create a process monitoring tool in C that periodically checks the status of specified processes. Use fork, exec, and wait to manage processes, and signals (kill, signal) to interact with them. The tool should provide a user interface to start, stop, and monitor processes.	Linux System Calls	Process management, signal handling
3	Build a simple chat application using TCP sockets in C. The application should support multiple clients connecting to a server and exchanging messages. Implement features like broadcasting messages to all clients and private messaging between clients.	TCP sockets	Network programming, concurrency
4	Design an IPC system using shared memory in C where multiple processes can read from and write to a shared memory segment. Implement synchronization mechanisms to ensure data consistency and handle concurrent access using semaphores.	Shared memory, semaphores	IPC, synchronization

TEXT BOOKS:

1. "The Linux Programming Interface: A Linux and UNIX System Programming Handbook", Michael Kerrisk, 1st Edition, No Starch Press, 2010.
2. "Linux Command Line and Shell Scripting Bible", Richard Blum and Christine Bresnahan, 3rd Edition, Wiley, 2015.
3. "Linux System Administration", Tom Adelstein and Bill Lubanovic, 1st Edition, O'Reilly Media, 2007.

REFERENCE BOOKS:

1. "UNIX and Linux System Administration Handbook", Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Dan Mackin, 5th Edition, Pearson, 2017
2. "Linux System Programming: Talking Directly to the Kernel and C Library", Robert Love, 2nd Edition, O'Reilly Media, 2013
3. "Linux Bible", Christopher Negus, 9th Edition, Wiley, 2015
4. "Advanced Programming in the UNIX Environment", W. Richard Stevens, Stephen A. Rago, 3rd Edition, Addison-Wesley Professional, 2013
5. "Pro Bash Programming: Scripting the GNU/Linux Shell", Chris F.A. Johnson, Jayant Varma, 2nd Edition, Apress, 2015.
6. "Beginning Linux Programming", Neil Matthew, Richard Stones, 4th Edition, Wrox, 2007

JOURNALS/MAGAZINES:

Magazines

1. Linux Journal
2. Linux Magazine (Linux Pro Magazine)
3. Linux Format

Journals

1. The Linux Gazette
2. IEEE Transactions on Software Engineering

3. ACM Transactions on Computer Systems (TOCS)
4. Journal of Systems and Software

SWAYAM/NPTEL/MOOCs

1. Introduction to Linux – Platform NPTEL
2. Operating Systems – Platform NPTEL
3. Linux Programming & Scripting – Platform NPTEL
4. Linux Basics and Shell Programming – Platform SWAYAM
5. Linux Foundation's Introduction to Linux – Platform edX
6. Linux Server Management and Security – Platform Coursera
7. Open Source Software Development, Linux, and Git – Platform Coursera
8. Linux Command Line Basics – Platform Udacity
9. Linux for Developers – Platform Coursera

Course Title	Object Oriented Programming using C++ Lab				Course Type		HC	
Course Code	B24EH0304	Credits	1		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

This laboratory course supplements the material taught in the theory course Object Oriented Programming. The objective of this lab is to get hands-on experience in Object Oriented Concepts using C++ programming language. Laboratory exercises will normally be conducted on Windows Operating system. The students will master the basic syntax of classes, objects and features of C++ like Abstraction, Polymorphism, Inheritance and Dynamic Binding.

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Practice basic and fundamental object-oriented programming concepts.
2. Write and practice C++ program that contains classes and objects.
3. Write and practice C++ program that contains constructors and destructors.
4. Implement the concepts of inheritance and polymorphism in building applications.
5. Implement the concepts of generic classes and exception handling to solve real world problems.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Design and develop C++ programs for solving simple problems.	1,2,3,4,12	1,2,3
CO2	Understand the concept of constructors and inline functions.	1,2,3,12	1,2
CO3	Develop the skillset to solve real world problems using the inheritance concept.	1,2,3,4,12	1,3
C04	Illustrate the polymorphism concept to implement early binding and late binding.	1,2,3,4,12	1,3
CO5	Demonstrate the applications of generic programming in solving real world problems.	1,2,3,4,12	1,3
CO6	Design and develop complex applications using the OOPs concepts to address needs of society, industry and others.	1,2,3,4,6,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1	√	√	√			
CO2	√	√	√			
CO3	√	√	√			
C04	√	√	√			
CO5	√	√	√			
CO6	√	√	√			

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	3	3	2	2				3			3	3	3	3
CO2	3	3	2									1	1	3	3
CO3	3	3	2									2	3	3	3
CO4	3	3	2									1	3	3	3
CO5	3	3	3									2	2	3	3
CO6	3	3	3	2	2				3			3	3	3	3

PRACTICE:

No	**Title of the Experiment**	**Tools and Techniques**	**Expected Skill / Ability**
C++ Programming (Practice Experiments 1 to 4 - Not to be Examined)			
Experiment 1: Simple Programs using the cin, cout objects			
a	**Printing the name, college name and address** Rama wants to print her name, college name, and her college address on the console using cout object. Design a C++ program to help Rama in printing the above details.	C++ programming language, input and output objects	Input and output statements
b	**Perform the basic arithmetic operations** Salman wants to perform simulation of basic calculator which includes all arithmetic operations such as +, - ,*, / and %. Design a C++ program to perform the simulation of basic calculator that includes all operations suggested by Salman.	C++ programming language, input and output objects	Input and output statements
c	**Printing the number read from the user without including namespace std** C++ programs should include the namespace std. std namespaces includes the cin and cout objects defined under it. Design a C++ program to read and print the data of all datatypes	C++ programming language, input and output objects	Input and output statements
	without including the namespace std.		

Experiment 2: Working with inline functions, recursive functions			
a	**To find square of a number** Inline functions are the special type of functions where body of the function is replaced in the function call. Design such inline functions to find square of a number	C++ programming language, inline function	Inline functions
b	**To find factorial of a number** Factorial of a number can be found using both iterative method and recursive method. Design a C++ program where we make a recursive call to find factorial of a number.	C++ programming language, recursive function	Recursive functions
c	**To convert temperatures between ^{0}C and ^{0}F.** Temperature can be converted between ^{0}C and ^{0}F. This conversion can be implemented using normal functions and inline functions. Design a C++ program to implement inline functions to convert temperatures between Celsius and Fahrenheit.	C++ programming language, inline function	Inline functions
Experiment 3: Implementing static or early binding through function overloading			
a	**To perform mathematical operations on different types of data.** Design a C++ program to perform addition and multiplication of data of all data types implementing the function overloading concept.	C++ programming language, function overloading	function overloading
b	**Calculate area of various shapes.** Develop a C++ program that calculates the area of various shapes (rectangle, circle, triangle) using function overloading. Create functions calculateArea() for each shape with appropriate parameters.	C++ programming language, function overloading	function overloading
PART - A C++ (Experiments 5 to 12 – To be Examined)			
Experiment 5: Programs on classes			
a	**Employee Management System in C++** Sanchit wants to maintain the details of his employee like empid, name, qualification, department and salary. Write a C++ program to create a class called Employee with the above details. Implement the following functions ○ Accept details for multiple employees. ○ Display details of all employees. ○ Modify the salary of an employee given their ID. by creating an object of type employee.	C++ programming language, classes, creating objects and accessing members of class	classes, creating objects and accessing members of class
b	**Student grade using array of objects** University maintains a database of student which contains Name, age, year, semester and marks scored. The grade is calculated according to their percentage of marks scored. a. 90 % and above – O grade b. 80% to 89% -- A grade c. 70% to 79% -- B grade d. 60% to 69% -- C grade Design a C++ program to create an array of objects of type student and print the grade of the student given the percentage of marks.	C++ programming language, classes, creating array of objects and accessing members of class	classes, creating array of objects and accessing members of class using objects
Experiment 6: Implementing Constructors			
a	**Employee Management System** Savitha wants Siri to design a class to represent employees in a company. She wants to implement overloaded constructor functions to initialize employee objects with different sets of attributes such as name, ID, department, and salary. Additionally, she also wants to define overloaded functions for	C++ programming language, Constructors, Implementing the three types of constructors	Constructors, Implementing the three types of constructors

	updating employee details based on various parameters. Write a C++ program to test these overloaded constructor and member functions by creating and modifying employee objects with different sets of attributes created by Siri.	input/output	
b	**String operations** The HR of the company decides to store the concatenated names (first name and last name) of the employees in database. Design a C++ program to use copy constructor for concatenation of names. ○ String S1=" Samhith", S2 = "Patil". S3=S1+S2.	C++ programming language, Constructors, Implementing the copy constructor input/output	Constructors, Implementing the copy constructor
Experiment 7: Implementing operator overloading			
a	**Overloading multiplication and division operators** The arithmetic operations can be implemented on the set of complex numbers. Design a C++ program where we overload the '-' operator to perform subtraction and overload the '*' operator to perform multiplication of any two complex numbers	C++ programming language, operator overloading, overloading * and / operators, input/output	operator overloading, overloading * and / operators
b	**String manipulation** Shyam is developing a string manipulation library and want to overload the + operator to concatenate two strings. Implement a C++ class to represent strings and overload the + operator to concatenate strings.	C++ programming language, operator overloading, overloading * and / operators, input/output	operator overloading, overloading + operator, strings
Experiment 8: Implementing dynamic objects			
a	**Car rental system** Design a class called Car with attributes like car make, model, year, rental price per day, etc. Implement methods for renting a car, returning a car, and calculating the total rental cost. Write a C++ program to simulate a car rental system by creating a dynamic object and calling appropriate functions using the pointer to object.	C++ programming language, dynamic objects input/output	dynamic objects
b	**Banking system** Write a C++ program using classes and array of dynamic objects (array of pointer to objects) to demonstrate the working of the banking system, consisting of the following functionalities a. Display all account details b. Search by account number c. Deposit the amount ○ d. Withdraw the amount	C++ programming language, array of dynamic objects input/output	Dynamic objects
Experiment 9: Implementing Inheritance			
a	**Student test management system** Write a C++ Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports using virtual base classes. The sports and test class should be inherited by a result class Write appropriate functions to add the scores and display the final result for a student.	C++ programming language, classes and objects, inheritance	Inheritance and its types

b	**Vehicle rental system** Design a vehicle rental system in C++ with a base class Vehicle having common attributes and a virtual function displayDetails (). Derive classes Car, Truck, and Motorcycle with specific attributes. Implement constructors and override displayDetails () in each derived class. Write a main program to input vehicle details, display them, and calculate rental cost.	C++ programming language, classes and objects, inheritance	Inheritance and its types

Experiment 10: Implementing late binding or run time polymorphism			
a	**Ecosystem with different types of animals** You are simulating an ecosystem with different types of animals, including mammals, birds, and reptiles. Create a base class named Animal with a virtual function named makeSound () to represent the sound each animal makes. Derive classes for specific types of animals (e.g., Mammal, Bird, Reptile) from the Animal class and implement the makeSound () function in each derived class. Write a main program to demonstrate virtual polymorphism by calling the makeSound () function on objects of different animal types.	C++ programming language, classes and objects, inheritance, virtual polymorphism	Inheritance, run time polymorphism
b	**Geometric shapes** Implement a base class Transaction with virtual functions processTransaction() and displayDetails(). Derive classes DepositTransaction and WithdrawalTransaction that override processTransaction() to deposit or withdraw funds, and displayDetails() to show transaction details. Write a program that processes a series of transactions using an array of Transaction* pointers and demonstrates late binding.	C++ programming language, classes and objects, inheritance, virtual polymorphism	Inheritance, run time polymorphism
Experiment 11: Exception handling			
a	**Simple calculator** You are tasked with developing a module for a banking application in C++. The module handles the transfer of funds between accounts. Describe how you would implement exception handling to ensure the reliability and integrity of transactions, considering potential issues such as insufficient funds. Explain your approach to handle these exceptions effectively, ensuring that transactions are either completed successfully or rolled back to maintain data consistency. Additionally, discuss any strategies you would employ to log errors, notify users, and retry transactions.	C++ programming language, inline function and exception handling	Inline function and exception handling
b	**Flight reservation system** Design a C++ program to implement a flight reservation system where users can book flights, view available seats, and cancel reservations. Implement classes for flights, passengers, and reservations. Use inheritance to implement the relation between the above classes and implement run time polymorphism to book seats. Implement exception handling if the input is invalid and also for seat availability.	C++ programming language, inline function and exception handling	Exception handling
Experiment 12: Templates			
a	**Implementing Sorting using templates** Sorting is arranging the elements in either ascending order or descending order. There are many sorting techniques like	C++ programming language, generic functions, any	generic functions

	merge sort, quick sort, selection sort etc. These sorting techniques can be used to sort data of any data types. Design a C++ program to implement bubble sort technique to sort data of any datatype using generic class.	sorting technique	
b	**Implement a generic stack data structure** Stack is LIFO data structure where elements are inserted and deleted from the same end. Write a C++ program to implement stack operations where the stack here is a generic stack and can hold data of any data type. (in this case it is int, float and char	C++ programming language, generic functions, any sorting technique	generic functions

	data type)		
PART - B C++ Lab (Mini-Project – To be Examined) Note: Each student shall be assigned any one of the following mini-projects			
1	**Encryption and Decryption Tool** Develop a C++ program that can encrypt and decrypt text using common cryptographic algorithms like Caesar cipher, Vigenère cipher, or RSA encryption. You can also explore more advanced encryption techniques like AES.	C++ programming language, concepts or features of OOPS	
2	**School Management System** School Management System project can be used to manage and track student performance, attendance, and grades. The School Management System project aims to develop a School Management System in C++ that can be used by schools to manage their daily operations. The system will enable the school to store, manage, and access student records, teacher records, course information, exam schedules, and other important data related to the activities in the school.	C++ programming language, concepts or features of OOPS	
3	**Hotel Reservation system** Create an application to manage hotel rooms, reservations, check-ins/check-outs, and billing. The application should include the below features: 1. **Room Management:** Add, update, and display hotel rooms with details like room number, type, availability, and price. 2. **Reservation System:** Reserve rooms for guests, check room availability, and handle reservation cancellations. 3. **Check-In/Check-Out:** Assign rooms to guests, capture guest information, and process check-ins/check-outs. 4. **Billing and Invoicing:** Calculate charges for guest stays, generate invoices, and handle payment processing. 5. **Guest Management:** Maintain guest records, update guest information, and track check-in/check-out dates.	C++ programming language, concepts or features of OOPS	

4	**Online shopping platform** Develop a basic online shopping platform allowing users to browse items, add them to a cart, and make purchases. The application should include the below features: **Features:** 1. **User Authentication:** Users can log in or create accounts. 2. **Inventory Management:** Database of products with details and real-time updates. 3. **Browsing Items:** Display items, organized by categories or search. 4. **Adding Items to Cart:** Users can add items to a dynamic shopping cart. 5. **Making Purchases:** Checkout process with payment integration. 6. **Order Tracking:** Users can track order status.	C++ programming language, concepts or features of OOPS
5	**Library management system** Design a C++ program for a library management system. Implement classes for books, users, and library. Use inheritance to model different types of users (like students, faculty), and polymorphism to handle operations like borrowing and returning books.	C++ programming language, concepts or features of OOPS
6	**Hangman Game** The hangman project game consists of guessing a secret word of a given length by entering one letter at a time. The game ends if the user does not correctly guess the word within the allotted number of guesses. The Hangman game will utilize classes to store the hidden word and guesses a class containing member methods used to track progress, and a class to represent the hangman figure. The game logic will be written using a variety of features such as structs, switch statements, and loops of C++ programming language.	C++ programming language, concepts or features of OOPS
7	**Phonebook Application** The Phonebook Application is made for storing and organizing contact information for people and businesses. Users can manage their contact list and save and search for contact information using this project. This project will offer a lot of functions, including name or phone number searching and sorting. Users can easily save and manage contact information with the C++ phonebook application. It provides users with an easy way to organize and retain contact information. The system is additionally created to be secure, ensuring that all user data is kept private.	C++ programming language, concepts or features of OOPS
8	**Finding Nearby Cabs** This project in C++ will use the great circle distance formula to find the nearest cab to a user location. The great circle distance formula is used for calculating the shortest distance between two points on a sphere, such as Earth or the moon. In this project, the user will give their current location as input and the distance of the cab along with its location will be sent to the user.	C++ programming language, concepts or features of OOPS

9	**Online Exam System** This project is a complete Online Exam System written in C++. It is a web-based system that enables users to set up tests, administer tests, view and grade results, and analyse performance data. Users can set up tests with multiple questions, including objective and essay questions. A Timer for each test can be generated and can randomly select questions from a large pool of questions. It also allows users to leave comments on the test results for further analysis. The system also allows the administrator to publish test results and print out a summary. It records exam answers, multiple choice, and short answer questions, time spent, and marks obtained. With this system, users will be able to take the exams with ease and accuracy.	C++ programming language, concepts or features of OOPS
10	**Tic-Tac-Toe Game** Implement the classic game of Tic-Tac-Toe where two players take turns marking spaces in a 3x3 grid.	C++ programming language, concepts or features of OOPS

TEXT BOOKS:

1.HerbertSchildt," The Complete Reference C++",4thEdition, TataMcGrawHill,2003.
2.SanleyB.Lippmann, Josee Lajore,"C++Primer",4th Edition, Pearson Education,2005.

REFERENCE BOOKS:

1. Paul J Deitel, Harvey M Deitel, "C++ for Programmers", Pearson Education, 2009.
2. K R Venugopal, Rajkumar Buyya,T Ravi Shankar, "Mastering C++",TataMcGrawHill, 1999.
3. ACM, ACM Transactions on Programming Languages and Systems (TOPLAS).

4. ACM Journal on Object-Oriented Programming

JOURNALS/MAGAZINES:

1. **https://www.geeksforgeeks.org/introduction-of-object-oriented-programming/**
2. https://www.geeksforgeeks.org/best-practices-of-object-oriented-programming-oop/
3. https://www.britannica.com/technology/object-oriented-programming
4. https://www.codeproject.com/Articles/22769/Introduction-to-Object-Oriented-Programming-Concept

SWAYAMNPTEL/MOOCs

1. **https://www.coursera.org/specializations/object-oriented-programming**
2. https://www.coursera.org/lecture/writing-running-fixing-code/introduction-to-the-programming-environment-sXizU
3. https://www.coursera.org/learn/object-oriented-design

Course Title	Course based Project				Course Type		SDC	
Course Code	B24EH0305	Credits	1		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

This course emphasizes hands-on learning through a series of projects. Students will apply theoretical knowledge to practical tasks, developing skills in problem-solving, critical thinking, collaboration, and communication. The projects will cover various topics within the subject and will require students to use tools and techniques relevant to the field.

COURSE OBJECTIVES:

The objectives of this course are to

1. Apply theoretical concepts to practical scenarios.
2. Develop project management and organizational skills.
3. Enhance problem-solving and critical thinking abilities.
4. Collaborate effectively with peers.
5. Communicate findings and results clearly.

COURSE OUTCOMES:

After the completion of this course, students will be able to do

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project work to attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing the project work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3

CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysis required for carrying out the project.	11	1,2,3
CO12	Engage in lifelong learning to improve knowledge and competence in the chosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

	Bloom's Level					
CO#	**Remember (L1)**	**Understand (L2)**	**Apply (L3)**	**Analyze (L4)**	**Evaluate (L5)**	**Create (L6)**
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3

CO12												3	3	3	3

Program description:

The objective of this course-based project is to implement and analyse several fundamental data structures and associated algorithms. This project aims to deepen understanding of data structures, their operations, time complexities, and practical applications.

List of Sample Projects:

1. Network Routing

Algorithm Problem

Statement:

Develop a network routing algorithm to determine the most efficient path between nodes in a network, considering factors like distance, cost, and traffic congestion.

Implementation Description:

- **Graph Representation:** Represent the network as a graph where nodes represent routers or switches, and edges represent the connections between them. Use an adjacency list or matrix for the graph representation.
- **Data Structures:**
  - **Priority Queue:** Use a priority queue to manage nodes to be explored, with the priority being the cost or distance.
  - **Hash Maps:** Maintain a hash map for the shortest path estimates and another for the predecessor nodes to reconstruct the path.
- **Algorithm:** Implement Dijkstra's or A* algorithm to find the shortest path. The priority queue will help efficiently select the next node to explore based on the current shortest path estimate.

2. Information Retrieval System

Problem Statement:

Create an information retrieval system that indexes a collection of documents and allows for efficient search and retrieval of information based on user queries.

Implementation Description:

- **Inverted Index:** Build an inverted index to map terms to the documents containing them. This index will facilitate fast lookups.
- **Data Structures:**
  - **Trie:** Use a trie for efficient prefix searches and auto-completion of queries.
  - **Hash Map:** Store the inverted index using a hash map where keys are terms and values are lists of document IDs.
- **Search Algorithm:** Implement Boolean or vector space models for querying. Use TF-IDF for ranking the relevance of documents to the query.

3. Traffic Light Simulation at Cross

Road Problem Statement:

Simulate the operation of traffic lights at a busy crossroad, considering traffic flow and optimizing wait

times.

Implementation Description:

- **Graph Representation:** Represent the crossroad as a graph with nodes as intersections and edges as roads.
- **Data Structures:**
  - **Queue:** Use queues to model the lanes and manage the flow of vehicles.
  - **Finite State Machine (FSM):** Use FSM to manage the states of traffic lights (red, yellow, green).
- **Algorithm:** Implement a traffic light control algorithm that adjusts light durations based on the traffic density in each direction.

4. Natural Language Understanding

Problem Statement:

Develop a system to understand and process natural language text, extracting meaningful information and responding to user queries.

Implementation Description:

- **Text Representation:** Use tokenization and parsing to represent the text.
- **Data Structures:**
  - **Trie:** For efficient storage and retrieval of vocabulary and phrases.
  - **Parse Tree:** Use parse trees to represent the grammatical structure of sentences.
- **Algorithm:** Implement natural language processing (NLP) techniques such as named entity recognition (NER), part-of-speech tagging, and sentiment analysis.

Course Title	Introduction to Design Thinking				Course Type		FC	
Course Code	B24CI0309	Credit	02		Class		III SEMESTER	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	2	2				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	1	1				
	Total	**2**	**3**	**3**	**28**	**14**	**50**	**50**

COURSE OVERVIEW:

Today, innovation is everyone's business. At every level, in every kind of organization, design thinking provides the tools that one needs to become an innovative thinker and uncover creative opportunities. For example, companies like Procter, Gamble and GE have incorporated Design Thinking into their strategy and marketing. The course draws on methods from engineering and design, and combines them with ideas from the arts, tools from the social sciences, and insights from the business world. In this course, students start in the field, where they discover the needs of the target audience. They then iterate ideas on teams to develop a range of promising possible solutions, create rough prototypes to take back out into the field, and learn to test with real people in the target audience.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To impart knowledge on design thinking process for understanding designs.
2. To provide design skills to analyze design thinking issues and apply the tools and techniques of design.
3. To inculcate attitude to solve societal problems using design thinking tools.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Develop problem solving skills.	1,2,9,10,12	1,2
CO2	Students will develop human-centric mindset while designing, innovating, developing, and testing solutions for new products, services, and processes	1,2,9,10,12	1,2
CO3	Enhance communication and understanding between the team members.	1,2,9,10,12	1,2
CO4	Enhance creative thinking and apply for solving real world problems.	1,2,9,10,12	1,2
CO5	Understand the role of innovation in the digital era and drive disruptive innovation.	1,2,3,5,9,10	1,2
CO6	Develop the ability to create and test prototypes that are customer-centric and innovative	1,2,3,4,5,9,10	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	✓	✓				
CO2	✓	✓	✓			
CO3	✓	✓				
CO4	✓	✓	✓			✓
CO5	✓	✓				✓
CO6	✓	✓			✓	

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2							2	2		2	3	2	
CO2	1	3							2	3		2	1	2	
CO3	1	2							3	2		3	1	2	
CO4	1	2							3	2		2	1	2	
CO5	2	2	3		2				3	3		2	2	3	
CO6	2	2	2	2	2				3	2		2	2	3	

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Design Thinking Process: Types of the thinking process, Design thinking: Definition, Origin of design thinking, Importance of design thinking, Design vs Design thinking. Problem Exploration, Case Studies from Embrace-Stanford Innovation Challenge, IDEO, GE Healthcare, The Good Kitchen- Denmark Program etc, identifying the target users for the problem selected, Survey on existing solutions for the problem identified.

Empathizing: Powerful Visualizing tool – a method to connect to the user, Creating Empathy maps – Case studies. Defining the problems: POV statements from User perspective. Idea generation: Methods to spark the innovative ideas – Brainstorming, Mind map, Story board, Provocation etc

What is a prototype? - Prototyping as a mind set, prototype examples, prototyping for products; Why we prototype? Fidelity for prototypes, Process of prototyping- Minimum Viable prototype

Prototyping for digital products: What's unique for digital, Preparation; Prototyping for physical products: What's unique for physical products, Preparation; Testing prototypes with users.

PRACTICE:

Sl.No	**Name of the Practice Session**	**Tools and Techniques**	**Expected Skill /Ability**
1	Identifying the problem that can be solved using Design Thinking approach	Observation and survey	Develop identifying human cantered Problems
2	Build the empathy maps for simple problems like single user	Visualization	Develop ability to understand other's emotions
3	Build the detailed empathy maps for problem identified in the teams formed	Visualization	Develop ability to understand other's emotions
4	Presentation by student teams	PPT	Develop ability to express their views
5	Obtain the insights into user's problems and make PoV statement	Understanding	Develop ma perception king problem statements from user
6	Presentation by student teams	PPT	Develop ability to express their views

Course Title	Technical Documentation				Course Type		FC	
Course Code	B24EN0308	Credits	1		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	1	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	**1**	**1**	**1**	**14**	**-**	**25**	**25**

COURSE OVERVIEW:

Technical writing is all about strategically placing facts and figures in a sensible and user-understandable way. A structured approach encourages creating a better output, all the while considering available resources and objectives. This course focusses on various factors to improve the skills of Technical documentation.

COURSE OBJECTIVES:

The objectives of this course are to:

1. Acquire language skills
2. Develop linguistic and communicative competencies
3. Study academic subjects more effectively using the theoretical and practical components of English syllabus, and hence will develop study skills and communication skills in formal and informal situations.

COURSE OUTCOMES (COs):

On successful completion of this course; the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Produce effective engineering documents that enable readers to access relevant information.	6,8,9,10,12	**1**
CO2	Learn to avoid communication problems that distract the readers, causing confusions, distrust, or misunderstanding.	6 8,9,10,12	**1**
CO3	Practice various verbal reasoning and grammar practice.	6,8,9,10,12	**1**
CO4	Search engineering information, both in traditional ways and online.	6,8,9,10,12	1
CO5	Write research/design reports with special emphasis on content and style.	6,8,9,10,12	1
CO6	Improve the art of presentations in team	6,8,9,10,12	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√					
CO2	√	√	√			
CO3		√				
CO4		√				
CO5		√	√			
CO6		√	√			

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						1		1	2	2		2	1		
CO2						1		1	2	2		2	1		
CO3						1		1	2	2		2	1		
CO4						1		1	2	2		2	1		
CO5						1		1	2	2		2	1		
CO6						1		1	2	2		2	1		

Note:1-Low,2-Medium,3-High

COURSE CONTENTS

Information Design and Development- Different kinds of technical documents, Information development life cycle, factors affecting information and document design, Technical Writing, Grammar and Editing- Technical writing process, forms of discourse, Writing drafts and revising, Collaborative writing, creating indexes, technical writing style and language. Basics of grammar, study of advanced grammar, editing strategies to achieve appropriate technical style.

Advanced Technical Communication : Introduction to advanced technical communication, Usability, Managing technical communication projects, time estimation, Single sourcing, Localization, Writing reports, project proposals, brochures, newsletters, technical articles, manuals, official notes, business letters, memos, progress reports, minutes of meetings, event report.

TEXT BOOKS:

1. David F. Beer and David McMurrey, Guide to writing as an Engineer, John Willey. New York, 2004
2. Diane Hacker, Pocket Style Manual, Bedford Publication, New York, 2003. (ISBN 0312406843)
3. Shiv Khera, You Can Win, Macmillan Books, New York, 2003.
4. Raman Sharma, Technical Communications, Oxford Publication, London, 2004.

Course Title	Indian Constitution and Cyber Law				Course Type		MC	
Course Code	B24MEM301	Credits	0		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	0	1	1				
	Tutorial	0	0	0	Theory	Practical	IA	SEE
	Practice	0	0	0				
	Total	0	1	1	14	0		

COURSE OVERVIEW

The Constitution of India lays down in defining fundamental political principles, establishes the structure, procedures, powers and duties of government institutions and sets out fundamental rights, directive principles and duties of citizen. It helps to know and understand state executive and elections system of India.

COURSE OBJECTIVES

The objectives of this course are to:

1. Know about the basic structure of Indian Constitution.
2. Know the Fundamental Rights (FR's), DPSP's and Fundamental Duties (FD's) of our constitution.
3. Know about our Union Government, political structure & codes, procedures.
4. Know the State Executive & Elections system of India.
5. Learn the Amendments and Emergency Provisions, other important provisions given by the constitution

COURSE OUTCOMES (Cos)

After the completion of the course, the student will be able to:

CO	Course Outcomes	Pos	PSOs
CO1	Analyse the basic structure of Indian Constitution	6,8,9, 12	
CO2	Remember their Fundamental Rights, DPSP's and Fundamental Duties (FD's) of our constitution	6,8,9, 12	
CO3	Know about Indian Union Government, political structure & codes, procedures.	6,8,9, 12	
CO4	Understand our State Executive & Elections system of India	6,8,9, 12	
CO5	Understand the Amendments and Emergency Provisions, other important provisions given by the constitution	6,8,9, 12	
CO6	Understand constitutional amendments till today	6,8,9, 12	

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1				√		
CO2		√				
CO3		√				
CO4		√				
CO5		√				
CO6		√				

COURSE ARTICULATION MATRIX

CO / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						**2**		**1**	**1**			**1**			
CO2						**2**		**1**	**1**			**1**			
CO3						**2**		**1**	**1**			**1**			
CO4						**2**		**1**	**1**			**1**			
CO5						**2**		**1**	**1**			**1**			
CO6						**2**		**1**	**1**			**1**			

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Indian Constitution: Necessity of the constitution, societies before and after the constitution adoption, introduction to the Indian constitution, making of the constitution, role of the constituent assembly.

Salient features of India Constitution: Preamble of Indian constitution and key concepts of the preamble, fundamental rights and its restriction and limitations in different complex situations.

DPSP's and Fundamental Duties: Directive Principles of State Policy (DPSP's) and its present relevance in Indian society, fundamental duties and its scope and significance in nation, union executive: parliamentary system, union executive – president, prime minister, union cabinet.

Executive and Elections system of India: Parliament - LS and RS, parliamentary committees, important parliamentary terminologies, judicial system of India, supreme court of India and other courts, judicial reviews and judicial activism, state Executive and Governer, CM, state cabinet, legislature - VS & VP, election commission, elections and electoral process, amendment to constitution, and important constitutional amendments till today, emergency provisions.

TEXT BOOKS

1. Kapoor, S.K., "Human rights under International Law and Indian Law", Prentice Hall of India, New Delhi, 2002.
2. Basu, D.D., "Indian Constitution", Oxford University Press, New Delhi,
2002. REFERENCES BOOKS
1. M V Pylee, "An Introduction to Constitution of India", S Chand & Company, 5th Edition.
2. Durga Das Basu, "Introduction to constitution of India", LexisNexis, 23rd Edition.

Course Title	AEC-1 (Ability Enhancement Course Placement Training)				Course Type		AEC	
Course Code	B24EF0308	Credits	1		Class		III Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	IA	SEE
	Practice	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW

Placement training classes can effectively be conducted as a part of an Ability Enhancement Course (AEC). These courses are designed to enhance students' skills, making them more prepared and competitive for the job market.

IV Semester												
Sl. No	Course Code	Title of the Course	HC/FC/SC/OE/MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0401	Optimization Techniques	FC	3	0	0	3	3	50	50	100	BSC
2	B24EF0401	Design and Analysis of Algorithms	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0402	Artificial Intelligence and Machine Learning	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0403	Data Engineering	HC	3	0	0	3	3	50	50	100	PCC
5	B24EH0404	Distributed Systems	HC	3	0	0	3	3	50	50	100	PCC
6	B24EHS41X	Professional Elective-2	SC	3	0	0	3	3	50	50	100	PEC
7	B24EF0405	Design and Analysis of Algorithms Lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0405	Artificial Intelligence and Machine Learning Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0406	Data Engineering Lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EH0407	Course-based Project (SDP) AI	SDC	0	0	1	1	2	25	25	50	Proj
11	B24CS0401	Professional Ethics	FC	1	0	0	1	1	25	25	50	HSMC
12	B24EE0310	Universal Human Values	HC	1	0	0	1	1	25	25	50	HSMC
13	B24AS0403	Environmental Science	MC	1	0	0	0	1				HSMC
14	B24EF0409	AEC-2 (Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				21	0	5	25	31	475	475	950	
TOTAL SEMESTER CREDITS				25								
TOTAL CUMULATIVE CREDITS				89								
TOTAL CONTACT HOURS				30								
TOTAL MARKS				950								

Course Title	Optimization Techniques				Course Type		FC	
Course Code	B24EH0401	Credits	3		Class		IV SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	0	0	0				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW:
Students will comprehend optimization techniques and applications, analyze characteristics of linear programming problems, and apply basic concepts of mathematics to formulate optimization problems

COURSE OBJECTIVE (S):
The objectives of this course are to

1. Establish a solid foundation in the basic concepts and types of optimization problems.
2. Acquire proficiency in classical optimization techniques, including linear and nonlinear programming.
3. Understand and analyze the mathematical properties of optimization problems, focusing on convexity and concavity.
4. Investigate and apply modern optimization techniques to solve complex real-world problems.

COURSE OUTCOMES (COs)
After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify and classify various optimization problems.	1,2,5,6	1,2,3
CO2	Formulate linear and nonlinear optimization models for practical problems	1,2,5,6	1,2,3
CO3	Solve linear programming problems using graphical methods, the Simplex method, and the Two Phase method	1,2,5,6	1,2,3
CO4	Analyze the properties of convex and concave functions using the Hessian matrix and eigenvalues	1,2,5,6	1,2,3
CO5	Apply appropriate techniques to solve single and multivariable nonlinear optimization problems	1,2,5,6	1,2,3
CO6	Utilize modern optimization techniques such as genetic algorithms and simulated annealing for solving complex problems	1,2,5,6	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**pply (L3)**	**nalyze (L4)**	**aluate (L5)**	**reate (L6)**
CO1		√				
CO2		√				
CO3			√			
CO4				√		
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2			3	2							3	3	3
CO2	3	2			3	2							3	3	3
CO3	3	2			3	2							3	3	3
CO4	3	2			3	2							3	3	3
CO5	3	2			3	2							3	3	3
CO6	3	2			3	2							3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Optimization Techniques: Overview of optimization problems- unconstrained vs. constrained, Single variable vs multivariable Optimization. Types of optimizations - linear, nonlinear, discrete, continuous. Linear Programming: Formulation of linear programming problems, Graphical method for solving LP problems, Simplex method and its applications, Two Phase method.

Convex Functions, Definition and Basic Properties, First-order and Second-order Conditions, Epigraph and Sublevel Sets. Concave Functions, Definition and Basic Properties, First-order and Second-order Conditions, Hypograph and Superlevel Sets. Hessian Matrix and Eigenvalues: Computation of Hessian Matrix, Eigenvalues and Definiteness, Using Eigenvalues to Determine Convexity or Concavity. Bordered Hessian Matrix- Second-order Conditions for Constrained Optimization. convex optimization problems, Quadratic Programming.

Introduction to nonlinear optimization problems, Types of nonlinear programming problems, Single variable optimization: Unimodal Function, Elimination Methods- Unrestricted search, Exhaustive Search. Multivariable optimization with no constraints, Multivariable Optimization with Equality Constraints, Multivariable Optimization with Inequality constraints.

Unconstrained Optimization Techniques: Introduction, Direct Search Methods, Random Search Methods, Indirect search(Descent) methods, Gradient of a function, Steepest Descent(Cauchy) Method, Newton's Method.

Modern Methods of Optimization: Introduction, Genetic Algorithms-Use case, Simulated Annealing- Use case, Particle Swarm Optimization- Use case, Ant Colony Optimization- Use case, Optimization of Fuzzy Systems- Use case, Neural-Network-Based Optimization- Use case.

TEXT BOOKS:

1. Singiresu S. Rao," Engineering Optimization – Theory and Practices", JohnWiley & Sons, 4th Edition, 2009
2. Operations research, theory and applications: J K Sharma
3. Convex Optimization: Stephen Boyd, Lieven Vandenberghe

REFERENCE BOOKS:

1. Operations Research Theory, Methods and Applications, S D Sharma, 2020
2. Introduction to Operations Research, 11th Edition, by Frederick S. Hillier, Gerald J. Lieberman, et al., 2021
3. Operations Research: An Introduction, 10e by Hamdy A. Taha
4. Multi-Objective Optimization using Evolutionary Algorithms, Kalyanmoy Deb, ISBN: 978-0-471-87339-6, July 2001

JOURNALS/MAGAZINES:

1. https://link.springer.com/book/10.1007/978-1-84996-129-5

2. https://link.springer.com/book/10.1007/978-0-387-36797-2

3. **https://link.springer.com/book/10.1007/978-3-319-91578-4**

SWAYAMNPTEL/MOOCs

1. **https://nptel.ac.in/courses/111105039**
2. **https://archive.nptel.ac.in/courses/106/108/106108056/**
3. **https://nptel.ac.in/courses/106106245**

Course Title	Design and Analysis of Algorithms				Course Type		HC	
Course Code	B24EF0401	Credits	3					
	LTP	Credits	ontact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	50

Pre-requisites: Programming with C , Data Structures.

COURSE OVERVIEW:

It is a conceptual course which gives insight to the students about different algorithm design techniques. Understanding the fundamental concepts of writing algorithms for the given problem using appropriate algorithmic design technique and analyzing for its time complexity makes students to develop problem solving skill sets and coding methodology. This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. Deals with analyzing Time and Space complexity of algorithms and to evaluate trade-offs between different algorithms.

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Develop a solid understanding of fundamental concepts in algorithm design, analysis, and data structures.
2. Enhance problem-solving skill sets and strategies for designing efficient algorithms.
3. Learn techniques for analysing the efficiency of algorithms, including Time and Space complexity analysis.
4. Evaluate the learnt concepts and techniques to solve a variety of complex problems using different algorithms.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand of fundamental algorithmic concepts	1, 2, 3, 5, 9,12	1,2,3
CO2	Design efficient algorithms and solve a variety of problems	1, 2, 3, 5, 9,12	1,2,3
CO3	Develop the skills to analyze the time complexity and space complexity of algorithms.	1, 2, 3, 5, 6, 9, 12	1,2,3
CO4	Analyze different algorithmic techniques and data structures to solve the problems efficiently.	1, 2, 3, 5, 6, 9, 12	1,2,3
CO5	Apply the knowledge and skills to solve problems that require algorithmic solutions	1, 2, 3, 5, 6, 9,12	1,2,3
CO6	Evaluate the algorithms for real time complex problems.	1, 2, 3,4, 5, 6, 9,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√	√		
CO3			√	√		

CO4				√		
CO5			√		√	
CO6			√	√	√	

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	2			1								3	3	
CO2		3			2							2	3	3	
CO3		2	3		3							2	3	3	3
CO4		2	2		2							2	3	3	
CO5			3		3				2			2	3		3
CO6			3	2	2				2			3		3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT:

Introduction to Analysis of Algorithm Efficiency: Notion of an Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types. Fundamentals of the Analysis of Algorithm Efficiency- The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive Algorithms, Mathematical Analysis of Recursive Algorithms.

Brute Force Approach: Bubble Sort, Selection Sort, Sequential Search and Brute-Force String Matching, Exhaustive Search, Depth-First Search and Breadth-First Search. Decrease-and-Conquer: Topological Sorting. Divide-and-Conquer: Merge sort, Quick sort. Transform and Conquer: Heaps and Heap sort.

Space and Time Trade-Offs: Sorting by Counting, Input Enhancement in String Matching Hashing: Open and Closed Hashing. Dynamic Programming: The Knapsack Problem and Memory Functions, Optimal Binar Search Trees, Warshall's and Floyd's Algorithms. Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes.

Backtracking: N-Queens Problem, Hamiltonian Circuit Problem, Subset-Sum Problem, Branch-and-Bound-Assignment Problem, Knapsack Problem, Travelling Salesman Problem, Basic concepts of deterministic and non-deterministic algorithms, P, NP, NP-Complete and NP-Hard problems.

Self Learning Component:

Bellman Ford Algorithm, Longest Common Subsequence, Multi-Stage Graph, Matrix Chain Multiplication, Naïve String Matching, Rabin Karp Algorithm, Knuth-Morris-Pratt (KMP) Algorithm.

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd edition, Pearson Education, 2017.
2. Ellis Horowitz, Satraj Sahni and Rajasekaran,Computer Algorithms/C++, Universities Press, 2nd Edition, 2014.

REFERENCE BOOKS:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 3rd edition, PHI Learning Private Limited, 2017.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson.
3. Donald E. Knuth, "The Art of Computer Programming", Volumes 1and 3 Pearson.

JOURNALS/MAGAZINES:

1. ACM Transactions on Algorithms.
2. ACM Journal of Algorithms and Computational Technology.

SWAYAMNPTEL/MOOCs

1. NPTEL Course on “Design and Analysis of Algorithms”, Prof. Abhiram G Ranade, Prof. Ajit A Diwan and Prof. Sundar Vishwanathan

2. https://nptel.ac.in/courses/106101060

3. Udemy Course on “Design and Analysis of Algorithms”, shravan Kumar Manthri

https://www.udemy.com/course/design-and-analysis-of-algorithms/?couponCode=IND21PM

Course Title	Artificial Intelligence and Machine Learning				Course Type		HC	
Course Code	B24EH0402	Credits	3		Class		IV SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

The course "Artificial Intelligence and Machine Learning" provides an in-depth introduction to the fundamental concepts, theories, and techniques of artificial intelligence (AI) and machine learning (ML). This course aims to equip students with a solid foundation in AI and ML and empower them to apply these principles to real-world problems. Throughout the course, students will gain a comprehensive understanding of the key components of AI and ML, including data preprocessing, feature engineering, model selection, and evaluation. They will learn about different types of machine learning algorithms, such as supervised learning, unsupervised learning, and reinforcement learning, and explore their applications across various domains. The course will start by laying the groundwork with an exploration of the basic principles and theories behind AI and ML. Students will examine the mathematical foundations and concepts underlying machine learning algorithms, deep learning enabling them to grasp the fundamental mechanisms behind these models.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Gain a solid understanding of the fundamental principles and theories of AI and ML.
2. Develop proficiency in applying different machine learning algorithms to real-world datasets.
3. Acquire the skills to preprocess and engineer data effectively for machine learning tasks.
4. Evaluate and compare the performance of different machine learning models using appropriate metrics.
5. Explore the ethical implications and considerations inherent to AI and ML systems.
6. Apply critical thinking and problem-solving skills to design and develop effective machine learning solutions

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain the fundamental concepts and terminology of artificial intelligence and machine learning.	1,2,3,5,6,9 10	1,2,3
CO2	Make use of different search strategies and compare their effectiveness.	1 to 8, 12	1,2,3
CO3	Demonstrate an understanding of the mathematical foundations of machine learning algorithms.	1 to 6,9,10,12	1,2,3
CO4	Apply machine learning algorithms to real-world datasets for solving classification and regression problems.	1 to 8,11	1,2,3
CO5	Interpret the performance of machine learning models using appropriate metrics and techniques.	1 to 10	1,2,3
CO6	.Build a model based on deep learning	1 to 6, 9,10	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3		√				
CO4			√			
CO5		√				
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	3	3	1		3	3			3	3			3	3	3
CO2	3	3	3	3	3	3	3	1				2	3	3	3
CO3	3	3	2	2	3	3			3	3		1	3	3	3
CO4	3	3	3	3	3	3	3	3			1		3	3	3
CO5	3	2	1	3	3	3	2	3	3	3			3	3	3
CO6	3	3	3	3	3	3			3	3			3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction: What Is AI ?, The Foundations of Artificial Intelligence, The State of the Art, Fundamental use cases of Artificial Intelligence, I**ntelligent Agents, Solving Problems by Searching:** Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Informed Search Strategies. Knowledge Representation and Reasoning, Knowledge Representation Issues, Representation and Mapping, Approaches to Knowledge Representation, Issues in Knowledge Representation

Machine Learning Basics : Learning Algorithms, Capacity, Overfitting and Underfitting, Hyperparameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning Algorithms, Unsupervised Learning Algorithms

Stochastic Gradient Descent, Building a Machine Learning Algorithm, Challenges Motivating Deep Learning, Deep Feedforward Networks: Example: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-Propagation and Other Differentiation Algorithms, Bagging and Other Ensemble Methods

Convolutional Networks: The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features , The Neuroscientific Basis for Convolutional Networks

Self Learning Component:

Regularization for Deep Learning and RNN

Online Tutorials and Courses, Coursera: "Machine Learning" by Andrew Ng, edX: "Artificial Intelligence (AI)" by Columbia University, Udacity: "AI for Robotics" by Sebastian Thrun,
Participation in AI Competitions, Kaggle: Participate in machine learning competitions, DrivenData: Engage in data science competitions focused on social impact.
To develop practical AI applications: Developing a chatbot using natural language processing, Creating a recommendation system, Implementing a game-playing AI (e.g., Tic-Tac-Toe, Chess).

TEXT BOOKS:

1. Russell, S. J., & Norvig, P. (2016). Artificial intelligence: a modern approach. Pearson

2. Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning – 29 copies available

3. Alberto Artasanchez, Prateek Joshi . Artificial Intelligence with Python – 13 copies available

REFERENCE BOOKS:

1. Rich, E. Kevin Knight y Shivashankar B. Nair. *Artificial Intelligence*.

2. David L. Poole and Alan K. Mackworth. Artificial Intelligence: Foundations of Computational Agents

3. N.P.Padhy. Artificial Intelligence and Intelligent Systems. Oxford

4. John D. Kelleher, Brian Mac Namee and Aoife D'Arcy. Fundamentals of Machine Learning for Predictive Data Analytics - Algorithms, Worked Examples, and Case Studies. The MIT Press

JOURNALS/MAGAZINES:

1. Journal of Artificial Intelligence Research (JAIR)
2. Artificial Intelligence (Elsevier)
3. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)

SWAYAMNPTEL/MOOCs

1. Coursera: "Machine Learning" by Andrew Ng
2. edX: "Artificial Intelligence (AI)" by Columbia University
3. Udacity: "AI for Robotics" by Sebastian Thrun

Course Title	Data Engineering				Course Type		HC	
Course Code	B24EH0403	Credits	3		Class		IV SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice			-	Theory	Practical	CIE	SEE
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

Presents techniques for designing, building, and managing information with relational databases, NoSQL databases, and big data infrastructure. Data Engineers are responsible for the creation and maintenance of the data infrastructure that facilitates the analysis of data by Data Scientists.

COURSE OBJECTIVE (S):

The Objectives of this course are to:

1. Introduce the basics of data mining, data types, similarity and dissimilarity measures
2. Illustrate the use of different SQL statements.
3. Discuss the topics such as Database Design and Normalization
4. The purpose of this course is to provide the students with the knowledge of Big data Analytics principles and techniques.
5. To learn the security of databases, design techniques of database security and learn the secure software design

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To understand the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system	1,2	1
CO2	To examine Secure Operating System Design and Secure Database Management System (DBMS)	1,2,3	1,2
CO3	To demonstrate the database for given real world application and analyze queries.	2,3,5,8,12	1,2
CO4	To present an optimized database using design guidelines and normalization techniques.	3,4,5,8,12	2,3
CO5	To experiment some real time applications using HADOOP and Map reduce.	2,3,4,5,8,12	2,3
CO6	To demonstrate a comprehensive understanding of Transaction Management in database systems.	2,3,4,6,8,12	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√	√				
CO2		√	√			
CO3			√	√		
CO4			√	√		
CO5		√		√		
CO6			√	√		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2												1		
CO2	1	2	2										2	2	
CO3		2	2		2			3				3	2	2	
CO4			2	2	2			3				3		2	2
CO5		2	2	2	2			3				3		2	2
CO6		2	2	2		2		3				3		2	1

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Data, Types of Data, Data Preprocessing, characteristics of the database approach, data models. Data quality and testing methodology. Data Warehouses: Concepts, star and snowflake schemas, OLAP, and data marts.
Security Software Design: Introduction A Methodological Approach to Security Software Design, Secure Operating System Design, Secure DBMS Design Security Packages Database Security Design.

SQL data definition and data types, specifying constraints in SQL, basic retrieval queries in SQL, insert, delete, update statements in SQL, additional features of SQL, schema change statements in SQL, Retrieving data using the SQL Select Statement.
Normal Forms: Functional dependencies, Normal forms based on primary keys, General definitions of second and third normal forms.
NoSQL Databases: Document, key-value, column-family, and graph databases.

Introduction to Big Data, Hadoop Architecture, Hadoop distributors, HDFS, HDFS Daemons, Anatomy of File Write and Read, Name Node, Secondary Name Node, and Data Node, Map Reduce Framework.
Discussion on case studies with sample examples.

Data Integration and Extract, Transform, and Load (ETL) processes, Data Modelling, Data Pipeline Orchestration.
Transaction Management: Transaction Concept, Transaction Atomicity and Durability.
Concurrency Control: Lock-Based Protocols, Deadlock Handling, Multiple Granularity.

Self Learning Component:
Database Management systems, Data Mining

TEXT BOOKS:

1. Database Security by Castano, Pearson Edition
2. Database Security and Auditing: Protecting Data Integrity and Accessibility, 1st Edition, Hassan Afyouni, THOMSON Edition.
3. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, “Introduction to Data Mining”, Pearson Education, 2007.
4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan: “Database System Concepts”, 6th Edition, McGraw Hill, 2010.
5. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O‟Reilly Media, 2012

REFERENCE BOOKS:

1. Database security by Alfred Basta, melissazgola, CENGAGE learning.
2. Jure Leskovec, Anand Rajaraman, Jeff Ullman, Mining of Massive Datasets, 3rd Edition.
3. Data Mining – Concepts and Techniques - JIAWEI HAN & MICHELINE KAMBER, Elsevier.
4. Big Data Analytics, Seema Acharya, Subhasini Chellappan, Wiley 2015.
5. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Business, Michael Minelli, Michehe Chambers, 1st Edition, Ambiga Dhiraj, Wiely CIO Series, 2013.
6. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O‟Reilly Media, 2012.
7. Big Data Analytics: Disruptive Technologies for Changing the Game, Arvind Sathi, 1st Edition, IBM Corporation, 2012

Course Title	Distributed Systems				Course Type		HC	
Course Code	B24EH0404	Credits	3		Class		IV Semester	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42	-	50	100

COURSE OVERVIEW:

This course provides a comprehensive introduction to the concepts, principles, and techniques of distributed systems. The course covers the basic concepts, including system models and communication protocols, as well as advanced topics such as distributed objects, process management, security, transaction processing, and distributed algorithms. Through this course, students will gain a deep understanding of how distributed systems work, the challenges involved, and the methods to address these challenges.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. To understand the fundamental concepts and architectures of distributed systems.
2. To explore the communication mechanisms and protocols used in distributed systems.
3. To identify the methods and algorithms for distributed transaction processing and concurrency control.
4. To analyze and implement distributed algorithms in various distributed system models.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To understand the core concepts and architecture of distributed systems.	1-6,12	1,2,3
CO2	Describe the mechanisms and protocols for interprocess communication and distributed objects	1-5,12	1,2,3
CO3	Develop the basic client-server and group communication systems	1-5,9,10,12	1,2,3
CO4	Identify the security requirements and apply cryptographic algorithms to secure distributed systems by differentiating between various security techniques	1-5, 8,9, 12	1,2,3
CO5	Inspect the mechanisms for distributed transaction processing and concurrency control by organizing and comparing different protocols.	1-5, 12	1,2,3
CO6	Compare and Contrast the distributed algorithms for synchronization, mutual exclusion, and leader election in various models, attributing specific algorithm choices to different network models.	1-5,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2		√				

CO3			√			
CO4			√			
CO5				√		
CO6				√		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	2	1	1	2	1						2	2	2	2
CO2	3	3	2	2	2							2	3	3	2
CO3	2	2	3	2	3				3	2		2	3	3	3
CO4	2	2	2	2	2			3	2			2	3	3	2
CO5	2	3	3	3	2							2	3	3	2
CO6	2	3	3	3	2							2	3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Basic Concepts:

Definition of a distributed systems, Examples, Resource sharing and the Web, Challenges, System models, Architectural and fundamental models, Networking Interprocess communication, External data representation and marshalling, Client-server and Group communication.

Distributed Objects and Process:

Distributed objects and remote invocation, Communication between distributed objects, Remote procedure call, Events and notifications - The operating system layer, Protection, Processes and Threads, Communication and invocation, OS Architecture. Security techniques, Cryptographic algorithms, Access control, Digital signatures, Cryptography pragmatics, Needham-Schroeder, Kerberos, Securing electronics transaction, IEEE
802.11 WiFi.

Distributed Transaction Processing:

Transactions - Nested transactions - Locks - Optimistic concurrency control - Timestamp ordering - Flat and nested distributed transactions - Atomic commit protocols - Concurrency control in distributed transactions - Distributed deadlocks - Transaction recovery - Overview of replication, Distributed shared memory and Web services.

Distributed Algorithms:

Synchronous network model - Algorithms: leader election, maximal independent set -
Asynchronous system model: I/O automata, operations on automata, fairness - Asynchronous shared memory model - Mutual exclusion: model, the problem, stronger conditions, lockout-free mutual exclusion algorithms, lower bound on the number of registers - Asynchronous network model - Asynchronous network algorithms: leader election in a ring and an arbitrary network.

Self-Learning Component:

Distributed file systems - Name services, Domain name system, Directory and discovery services, Peer to peer systems, Napster file sharing system, Peer to peer middleware routing overlays – Clocks, Events and process states Clock

Synchronization - Logical clocks Global states - Distributed debugging - Distributed mutual exclusion - Elections - Multicast communication

TEXT BOOKS:

1. George Coulouris, Jean Dollimore, and Tim Kindberg, " Distributed Systems Concepts and Design", 5th ed., Pearson Education, 2011.
2. Andrew S. Tanenbaum, Maarten van Steen, "Distributed Systems Principles and Paradigms", 2nd ed., Pearson Education, 2006.
3. Nancy A. Lynch, "Distributed Algorithms", Hardcourt Asia Pvt. Ltd., Morgan Kaufmann, 2000.

REFERENCE BOOKS:

1. Distributed Systems, An Algorithm Approach, Sikumar Ghosh, Chapman & Hall/CRC, Taylor & Francis Group, 2007.
2. Andrew S Tanenbaum: Distributed Operating Systems, 3rd edition, Pearson publication, 2007
3. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2008
4. Sunita Mahajan, Seema Shan, " Distributed Computing", Oxford University Press,2015

SWAYAMNPTEL/MOOCs

1. **https://onlinecourses.nptel.ac.in/noc21_cs87/preview**
2. **https://onlinecourses.nptel.ac.in/noc21_cs15/preview**

Professional Elective 2

Course Title	Agile Software Development and DevOps				Course Type	SC	
Course Code	B24EHS411	Credits	3		HC	IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester	Assessment in Weightage	
	Lecture	3	3	3			
	Tutorial	-	-	-	Theory	CIE	SEE
	Practical	-	-	-			
	Total	3	3	3	42	50	50

COURSE OVERVIEW:

The course provides students with a knowledge on the basic principles of software development life cycle, activities involved in software requirements engineering, software development, testing, evolution and maintenance. It introduces concepts such as software processes and agile methods and essential software development activities. New methods to approach old development problems with DevOps.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Discuss the importance of the software development process using agile.
2. Demonstrate the workflow of Automating process.
3. Explain the development of a software product using Agile method.
4. Illustrate with the introduction and importance of DevOps concepts with a case study.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Apply software development process to solve complex problems in software engineering.	1,2	1,2
CO2	Make use of Agile principles for rapid software development and automation.	1,3	3
CO3	Summarize the basic principles of agile approach and need of integration, delivery and deployment.	1,3,5	1
CO4	Apply the concepts of agile life cycle modeling and automation in real world applications.	1,3,5,8	2,3
CO5	Distinguish between the traditional SDLC and agile ALM model for efficient and effective product delivery.	1,3,4	1
CO6	Develop the real-world applications using DevOps tools for continuous business improvement.	1,3,4,5,8,9	2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3		√				
CO4			√			
CO5				√		
CO6			√			

COURSE ARTICULATION MATRIX:

CO #/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	3											3	3	
CO 2	3		3												3
CO 3	3		3		2								3		
CO 4	3		3		3			2						3	3
CO 5	3		3	3									3		
CO 6	3		3	3	2			3	3					3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Requirements and Development process: Software Processes: Models, Process iteration, Process activities. Software Requirements: Functional and Non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements Engineering Processes: Feasibility studies, Requirements elicitation and analysis.

Agile Application Life Cycle Management: Goals of Agile Application Life Cycle Management, Why Is Agile ALM Important? Understanding the Paradigm Shift, Rapid Iterative Development, Focus on12 Agile Principles, Agile Manifesto, scrum, XP, Fixed Time box Sprints, Customer Collaboration, Requirements and Documentation, Control the Flow: Sprint iteration, overview, Sprint Planning, Sprint Reviews, Sprint Retrospectives, Sprint Planning.

Automating the agile ALM: Goals of Automating the Agile ALM, Why Automating the ALMIs Important, Tools, Do Tools Matter? Process over Tools, Understanding Tools in the Scope of ALM, Staying Tools Agnostic, Commercial versus Open Source, Automating the Workflow, Process Modeling Automation, Managing the Lifecycle with ALM, Broad Scope of ALM Tools, Achieving Seamless Integration, Managing Requirements of the ALM, Creating Epics and Stories, Systems and Driven Development, Environment Management, Gold Copies, Supporting the CMDB, Supporting Operations, Incident Management, Project Management, Planning the PMO, Planning for Implementation, Evaluating and Selecting the Right Tools, Defining the Use Case, Training Is Essential, Vendor Relationships, Keeping Tools Current.

DevOps: Goals of DevOps, Why Is DevOps Important? Where Do I Start? How Do I Implement DevOps? Developers and Operations Conflicts, Developers and Operations Collaboration, Need for Rapid Change, Knowledge Management, the Cross-Functional Team, Is DevOps Agile? The DevOps Ecosystem, Moving the Process Upstream, Left-Shift, Right-Shift, DevOps in Dev, Fully Automated Deployment, Deployment Pipeline, Dependency Control, Configuration Control, Configuration Audits, Continuous Integration and Continuous Delivery, Automated testing using Selenium, UFT, version control using Git, QA and DevOps, Information Security, Infrastructure as Code, Docker Containerization, Taming Complexity, Automate Everything, Disaster Recovery and Business Continuity, Continuous Process Improvement, DevOps on Cloud.

TEXTBOOKS:

1. BobAiello and LeslieSachs, “Agile Application Life Cycle Management Using DevOps to Drive Process Improvement”, Addison Wesly, First printing, 2016.
2. Martin, R.C., Agile Software Development: Principles, Patterns, and Practices, Pearson Education Publisher, 2011.

REFERENCE BOOKS:

1. Ken Schwaber, Agile Project Management with Scrum, 2014,1st edition, Microsoft Press US.
2. Roman Pichler, Agile Product Management with Scrum: Creating Products that Customers Love, 2011, First edition, Addison-Wesley.
3. Learning Agile: Understanding Scrum, XP, Lean, and Kanban, By Andrew Stellman, Jennifer Greene, 2015, O Reilly.
4. DevOps: Puppet, Docker, and Kubernetes By Thomas Uphill, John Arundel, Neependra Khare, Hideto Saito, Hui-Chuan Chloe Lee, Ke-Jou Carol Hsu, Packt, 2017.
5. Roger S, “Software Engineering-A Practitioner’s Approach”, seventh edition, Pressman, 2010.
6. DevOps: Continuous Delivery, Integration, and Deployment with DevOps: By Sricharan Vadapalli, Packt, 2018.

JOURNALS/MAGAZINES:

1. IEEE Transactions on Software Engineering.
2. Journal of Software Engineering Research and Development.
3. International Journal of Agile and Extreme Software Development.
4. A decade of agile methodologies: Towards explaining agile software development.
5. Journal of Systems and Software.
6. IEEE Transactions on Software Engineering.

SWAYAM NPTEL/MOOCs:

1. https://www.mooc-list.com/course/agile-project-management-coursera

2. https://openclassrooms.com/en/courses/4544621-learn-about-agile-project-management-and-scrum
3. https://www.coursera.org/learn/devops-culture-and-mindset
4. https://www.coursera.org/learn/uva-darden-continous-delivery-devops

Course Title	Game Theory				Course Type		SC	
Course Code	B24EHS412	Credits	3		Class		IV Semester	
Course Structure	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:
This course offers a comprehensive introduction to game theory, covering the basics of strategic and extensive games, Bayesian games, and repeated games. Students will explore applications across various disciplines and address ethical considerations and limitations. Through lectures, case studies, and projects, students will learn to analyze, model, and solve real-world strategic interactions.

COURSE OBJECTIVE (S):
The objectives of this course are to:

1. Comprehend the basics of strategic gaming and mixed strategic equilibrium.
2. Enable students to develop skills on extensive gaming strategies.
3. Analyze and discuss various gaming models.
4. Illustrate some real-time situations.

COURSE OUTCOMES (COs)
After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To interpret the basics of strategic gaming and extensive games.	1, 2, 5	1, 3
CO2	To analyze gaming strategies on real-time incidences.	2, 4, 9	1, 2
CO3	To analyze the models of gaming on real-time incidences.	3, 4, 11	1, 3
CO4	To apply game theory in the real-world problems.	1, 6, 7	1, 2
CO5	To demonstrate Interdisciplinary applications of Game Theory.	1, 6, 10	1, 3
CO6	To interpret Ethical Implications and Limitations of Game Theory.	8, 12	1, 2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	nalyze (L4)	valuate (L5)	Create (L6)
CO1		√				
CO2				√		
CO3				√		
CO4			√			
CO5		√				
CO6		√				

COURSE ARTICULATION MATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	2			2								3		3
CO2		3		2					2				3	2	3
CO3	3	3	2	3						2	2		3	3	3
CO4	3		2			2	2					3	3	3	3
CO5	3					3				2		2	3		3
CO6								3				2	3		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Strategic Games: What is game theory? The theory of rational choice, Strategic games; Examples: The prisoner's dilemma, Bach or Stravinsky, Matching pennies; Nash equilibrium; Examples of Nash equilibrium; Best response functions; Dominated actions.

Strategic games in which players may randomize; Mixed strategy Nash equilibrium; Dominated actions; Pure equilibrium when randomization is allowed.
Use cases: Expert Diagnosis; Equilibrium in a single population.

Bayesian Games, Motivational examples; General definitions; Two examples concerning information; Use cases: Cournot's duopoly game with imperfect information, Providing a public good; Auctions: Auctions with an arbitrary distribution of valuations.

Competitive Games: Strictly competitive games and maximization.
Repeated games: The main idea; Preferences; Repeated games; Finitely and infinitely repeated Prisoner's dilemma; Strategies in an infinitely repeated Prisoner's dilemma; Nash equilibrium of an infinitely repeated Prisoner's dilemma, Nash equilibrium payoffs of an infinitely repeated Prisoner's dilemma.

Self Learning Component:

Extensive games with perfect information; Strategies and outcomes; Sub- game perfect equilibrium;
Finding sub-game perfect equilibria of finite horizon games: Backward induction;
Use cases: The ultimatum game, Stackelberg's model of duopoly.

TEXT BOOKS:

1. Martin Osborne: "An Introduction to Game Theory", Oxford University Press, First Indian Edition, 2009, 7th impression, ISBN – 0195128958.

REFERENCE BOOKS:

1. Roger B. Myerson: "Analysis of Conflict Game Theory", Re-print Edition, Harvard University Press, 2008, ISBN – 978- 0674341166.

2. Frederick S. Hillier and Gerald J. Lieberman: "Introduction to Operations Research, Concepts and Cases", 9th Edition; Tata McGraw Hill, 2010, ISBN – 0073376299.

3. Joel Watson: "An Introduction to Game Theory" Strategy, 2nd Edition, W.W. Norton &Company, 2007, ISBN – 9780393929348.

JOURNALS/MAGAZINES:

1. International Journal of Game Theory

   https://link.springer.com/journal/182
2. Game Theory Magazine

   https://www.quantamagazine.org/tag/game-theory/

SWAYAM,NPTEL/MOOCs:

1. http://nptel.ac.in/courses.php?disciplineID=111
2. http://www.class-central.com/subject/math(MOOCs)
3. http://academicearth.org/

Course Title	Cryptography and Network Security				Course Type		SC	
Course Code	B24EHS413	Credits	3		Class		IV Semester	
Course Structure	LTP	Credits	ontact Hours	Workloa d	Total Number of Classes Per Semester		ssessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

This course will emphasize on principles and practice of cryptography and network security: classical systems, symmetric block ciphers (DES, AES, other contemporary symmetric ciphers), linear and differential cryptanalysis, perfect secrecy, public-key cryptography algorithms for factoring and discrete logarithms, cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes and email security, viruses, firewalls, digital right management, and other topics. In this course students will learn as aspects of network security and cryptography

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the fundamental concepts of cryptography.
2. Describe public key cryptography and message authentication.
3. Demonstrate the key distribution using Symmetric or Asymmetric encryption.
4. Discuss security applications in the field of Information technology.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify the fundamental concepts of cryptography	1 to 5	1
CO2	Use symmetric and asymmetric key algorithms for cryptography	1 to 5	2
CO3	Apply the key distribution technique using Symmetric or Asymmetric encryption	1 to 5	2,3
CO4	Develop real-world security applications in the field of Information technology.	1 to 5	2,3
CO5	Understanding of Authentication functions the manner in which Message Authentication Codes and Hash Functions works.	1 to 5	1,2
CO6	Apply the System security concepts for real-time secure applications	1 to 5	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3			√			
CO4						√
CO5		√				
CO6		√				

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	2	1								3		
CO2	3	2	3	1	2									3	
CO3	3	1	2	2	3									3	3
CO4	2	2	2	1	1									3	3
CO5	3	2	3	2	1								3	3	
CO6	2	1	2	2	2									3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT :

INTRODUCTION: Security trends, The OSI Security Architecture, Security Attacks, Security Services and Security Mechanisms, A model for Network security.
CLASSICAL ENCRYPTION TECHNIQUES: Overview of Symmetric Cipher Modes, Substitute Techniques, Transposition Techniques, and Stenography.

INTRODUCTION TO NUMBER THEORY: Prime Numbers, Fermat's and Euler's Theorem, Testing for Primality, The Chinese Remainder Theorem, Discrete logarithms.

TOOLS DEMONSTRATION: Wireshark, Metasploit, Snort, Nmap, Kali Linux.

SYMMETRIC KEY CRYPTOGRAPHY: Block Cipher Principles, Data Encryption Standards, Advanced Encryption Standards, Block Cipher Modes of Operation, Stream Cipher and RC4.
PUBLIC KEY CRYPTOGRAPHY: Principles Public key crypto Systems, RSA algorithm, Key Management, Diffie Hellman Key Exchange, Overview of Elliptic Curve Cryptography.
MESSAGE AUTHENTICATION AND HASH FUNCTIONS: Authentication Requirement, Authentication Function, Message Authentication Code, List of Hash Function, Secure Hash Algorithm, HMAC, CMAC, an overview of Digital Signature.
AUTHENTICATION APPLICATION: Kerberos, X.509 Authentication Service, Public Key Infrastructure.

EMAIL SECURITY: Pretty Good Privacy (PGP) and S/MIME**. IP SECURITY:** Overview, IP Security Architecture.

SYSTEM SECURITY: Intrusion Detection, Overview of Packet Sniffing, Penetration testing, Database testing, DoS attacks. Password Management, Virus, and threats, Virus Countermeasures.
FIREWALLS: The Need for Firewalls, Firewall Characteristics, Types of Firewalls.

SELF-LEARNINGEXERCISES:
The student is expected to study more about the following topics: IoT complexity leads to security issues and further DDoS attacks, Malware, Cloud security, Roles of AI and machine learning in cyber security.

TEXT BOOKS:

1. William Stallings, "Cryptography and Network Security Principles and Practice", Pearson, Sixth edition, 2013
2. Behrouz A. Forouzan, "Cryptography and Network Security", McGraw Hill, 2007.

REFERENCE BOOKS:

1. William Stallings, "Network Security Essentials Applications and Standards", Fourth edition, Prentice Hall, 2011.
2. Joseph MiggaKizza," Guide to Computer Security, Springer Science & Media Inc., Third edition, 2015
3. Andrew S.Tanenbaum, "Computer Networks", Pearson, Fifth edition, 2015.
4. AtulKahate, "Cryptography and Network Security", McGraw Hill, 2013.

JOURNALS/MAGAZINES:

1. Springer Journal of Cryptographic Engineering, ISSN 2190-8508
2. ACM, ACM- International Journal of Applied Cryptography, ISSN:1753-0563
3. IEEE, IEEE Transactions on Information Forensics and

4. Elsevier, Journal of Information Security and Applications.

SWAYAM/NPTEL/MOOCs:

1. Foundations of Cryptography: https://nptel.ac.in/courses/106/106/106106221/

2. Cryptography and Network Security: https://nptel.ac.in/courses/106/105/106105162/

Course Title	Design and Analysis of Algorithms Lab				Course Type		HC	
Course Code	B24EF0405	Credits	1		Class		IV Semester	
Course Structure	LTP	Credits	Contact	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

This course gives insight to the students about different algorithm design techniques. Understanding the fundamental concepts of writing algorithms for the given problem using appropriate algorithmic design technique and analyzing for its time complexity makes students to develop problem solving skill sets and coding methodology. This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. Deals with analyzing Time and Space complexity of algorithms and to evaluate trade-offs between different algorithms.

COURSE OBJECTIVE (S):

The objective of this course are to

1. Demonstrate performance of algorithms with respect to time and space complexity.
2. Explain Graph and Tree traversals techniques.
3. Understand the concepts of greedy method and dynamic programming for different applications.
4. Illustrate the methods of Backtracking, Branch and bound techniques.
5. Familiarize the concepts of deterministic and non-deterministic algorithms.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Implement sorting and searching techniques using different algorithmic techniques.	1, 2, 3, 5, 9,12	1,2,3
CO2	Implement Tree Traversal method and Greedy Algorithms	1, 2, 3, 5, 9,12	1,2,3
CO3	Develop the skill set to solve problems using Dynamic Programming concepts.	1, 2, 3, 5, 6, 9,12	1,2,3
CO4	Illustrate Backtracking, Branch and Bound concept to solve variousproblems	1, 2, 3, 5, 6, 9,12	1,2,3
CO5	Demonstrate Time and Space complexities of various algorithms	1, 2, 3, 5, 6, 9,12	1,2,3
CO6	Analyze and evaluate different performance analysis methods for non-deterministic algorithms	1, 2, 3,4, 5, 6,9, 12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1		√				
CO2			√	√		
CO3			√	√		
C04				√		

CO5			√		√	
CO6			√	√	√	

COURSE ARTICULATION MATRIX:

CO# /	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	2	3		1				2			2	3	3	3
CO2	3	3	3		2				2			2	3	3	3
CO3	3	2	3		3	2			2			2	3	3	3
CO4	3	2	3		3	2			2			2	3	3	3
CO5	3	2	3		3	2			2			2	3	3	3
CO6	3	2	3		3	2			2			2	3	3	3

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.	Text Editor/ C IDE, C Compiler	C programming
2	a. Obtain the Topological ordering of vertices in a given digraph. b. Compute the transitive closure of a given directed graph using Warshall's algorithm.	Text Editor/ C IDE, C Compiler	C programming
3	Implement 0/1 Knapsack problem using Dynamic Programming.	Text Editor/ C IDE, C Compiler	C programming
4	From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.	Text Editor/ C IDE, C Compiler	C programming
5	Find the Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.	Text Editor/ C IDE, C Compiler	C programming
6	Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.	Text Editor/ C IDE, C Compiler	C programming
7	Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.	Text Editor/ C IDE, C Compiler	C programming
8	For a given graph a. Print all the nodes reachable from a given starting node in a digraph using BFS method. b. Check whether a given graph is connected or not using DFS method.	Text Editor/ C IDE, C Compiler	C programming
9	Implement N Queen's problem using Back Tracking.	Text Editor/ C IDE, C Compiler	C programming
10	Implement All-Pairs Shortest Paths Problem using Floyd's algorithm.	Text Editor/ C IDE, C Compiler	C programming

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Implement and analyze the performance of algorithms like Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, and Quick Sort. The project should generate the random datasets of varying sizes, apply each sorting algorithm to these datasets, and measure the time complexity. Students should present their findings, highlighting the strengths and weaknesses of each algorithm.	Text Editor/ C IDE, C Compiler	C programming
2	Students will choose problems that exhibit overlapping subproblems and optimal substructure. Examples include the 0/1 Knapsack problem, Longest Common Subsequence, and Matrix Chain Multiplication. The project involves implementing Dynamic Programming solutions, analyzing time and space complexities, and presenting the optimized solutions.	Text Editor/ C IDE, C Compiler	C programming
3	In this mini-project, students will explore various pathfinding algorithms applied to graphs or grids. Problems may involve finding the shortest path between two points, maze solving, or routing in a network. Common algorithms to be considered are Dijkstra's Algorithm, A* Algorithm, and Depth-First Search. Students will implement these algorithms, analyze their performance, and present their findings.	Text Editor/ C IDE, C Compiler	C programming
4	Implement various graph traversal techniques. Students will choose problems that require Breadth-First Search (BFS), Depth-First Search (DFS), or both. Examples include connectivity analysis, cycle detection, and node reachability. The project should involve implementing traversal algorithms, analyzing their applications, and presenting findings.	Text Editor/ C IDE, C Compiler	C programming

TEXT BOOKS:

1. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, 3rd edition, Pearson Education, 2017.
2. Ellis Horowitz, Satraj Sahni and Rajasekaran,Computer Algorithms/C++, Universities Press, 2nd Edition, 2014.

REFERENCE BOOKS:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, 3rd edition, PHI Learning Private Limited, 2017.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, “Data Structures and Algorithms”, Pearson.
3. Donald E. Knuth, “The Art of Computer Programming”, Volumes 1and 3 Pearson.

JOURNALS/MAGAZINES:

1. ACM Transactions on Algorithms.
2. ACM Journal of Algorithms and Computational Technology.

Course Title	Artificial Intelligence and Machine Learning Lab				Course Type		HC	
Course Code	B24EH0405	Credits	1		Class		IV Semester	
Course Structure Per Semester	LTP	Credits	Contact	Workload	Total Number of Classes		Assessment in Weightage	
	Lecture	-	-	-				
	Theory	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	**1**	**2**	**2**	**0**	**28**	**25**	**25**

COURSE OVERVIEW:

The lab component of the "Artificial Intelligence and Machine Learning" course provides an in-depth, hands-on introduction to the fundamental concepts, theories, and techniques of artificial intelligence (AI) and machine learning (ML). This lab is designed to equip students with a solid foundation in AI and ML, empowering them to apply these principles to real-world problems through practical experience.

Throughout the lab sessions, students will gain comprehensive knowledge of key AI and ML components, including data preprocessing, feature engineering, model selection, and evaluation. They will explore various types of machine learning algorithms, such as supervised learning, unsupervised learning, and deep learning, and investigate their applications across different domains.

COURSE OBJECTIVE (S):

The objectives of this course are to :

1. Gain a solid understanding of the fundamental principles and theories of AI and ML.
2. Develop proficiency in applying different machine learning algorithms to real-world datasets.
3. Acquire the skills to preprocess and engineer data effectively for machine learning tasks.
4. Evaluate and compare the performance of different machine learning models using appropriate metrics.
5. Apply critical thinking and problem-solving skills to design and develop effective machine learning solutions

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Illustrate to implement different search algorithms for problem solving	1 to 8, 12	1,2,3
CO2	Apply machine learning algorithms to real-world datasets for solving classification and regression problems.	1 to 8, 12	1,2,3
CO3	Compare and experiment with different hyperparameters of machine learning algorithms	1,2,4,5,6, 9 ,10	1,2,3
C04	Interpret the performance of machine learning models using appropriate metrics and techniques.	1 to 10	1,2,3
CO5	Choose different packages of python used for AI concepts machine learning	1,2,4,5,6, 9 ,10	1,2,3
CO6	Build models based on deep learning	1 to 6, 9,10	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1		√				
CO2			√			
CO3			√			
C04		√				
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	3	3	3	3	3	3	1				2	3	3	3
CO2	3	3	3	3	3	3	3	3			1		3	3	3
CO3	3	3		3	3			3	3	3			3	3	3
CO4	3	2	1	3	3	3	2	3	3	3			3	3	3
CO5	2	3		3	3	2			3	3			3	3	3
CO6	3	3	3	3	3	3			3	3			3	3	3

Practice:

No	**Title of the Experiment**	**Tools and Techniques**	**Expected Skill/Ability**
1	Uniform Cost Search (UCS) is used to find the solution to a problem along with minimizing the cost of the path from the initial state to the goal state. UCS works by selecting the path with the minimum cost at each step after exploring all the possible paths. Write a program to implement Uninformed search algorithms – a) Breadth-First Search (BFS) for Tic- Tac-Toe problem b) Depth-First Search (DFS) for Water Jug problem	Python queues, stacks	Design and implement search algorithms
2	Implement Informed search algorithms A* and memory-bounded A* algorithms for an autonomous vehicle navigating complex urban environments. These algorithms are crucial for finding optimal routes considering factors like traffic congestion and road conditions. How would you approach implementing them to ensure both accuracy and efficiency in route planning for the vehicle?	Python Priority queues, Graph	Capability to analyze and optimize the time

		representations	and space complexity of search algorithms.
3	In the traveling salesman problem each state is typically defined as a permutation of the cities to be visited, and the neighbors of any state are the set of permutations produced by swapping any two of these cities. The well-defined way in which the states are altered to produce neighboring states is called a "move", and different moves give different sets of neighboring states. These moves usually result in minimal alterations of the last state, in an attempt to progressively improve the solution through iteratively improving its parts. Write a program to implement Simulated Annealing Algorithm	Python Simulated Annealing (SA) Traveling Salesman Problem (TSP) Arrays or lists	Ability to design a suitable move function for generating neighboring states in TSP.
4	The Hill Climbing technique is mainly used for solving computationally hard problems. It looks only at the current state and immediate future state. Hence, this technique is memory efficient as it does not maintain a search tree. Write a program to implement Hill Climbing Algorithm.	Python Arrays or lists	Ability to select appropriate variants of Hill Climbing based on the problem.
5.	As a data scientist working for a car dealership, you need to build a regression model to predict car prices based on various features like mileage, year, make, model, and engine size. Illustrate to a) Compare three regression models to predict car prices using the Car Price Prediction dataset for the analysis. Which regression model would you start with to predict car prices? b) How to handle categorical variables? c) How to evaluate the model Consider fewer features: Numeric features: wheelbase, curbweight, enginesize, horsepower, highwaympg Categorical features: fueltype, carbody, drivewheel	Python NumPy, pandas, scikit-learn, statsmodels, matplotlib, seaborn.	Ability to implement and tune various regression models.
6	Machine learning model performance can be enhanced by performing feature engineering, cross-validation, and hyperparameter tuning. Could you illustrate how to enhance car price prediction regression models by performing feature engineering, cross-validation, and hyperparameter tuning on the Lasso Regression model?	Scikit-learn, NumPy, pandas, matplotlib, seaborn feature scaling	Proficiency in implementing and interpreting cross-validation techniques.

7	Build a classification model to identify different species of iris flowers based on their features such as sepal length, sepal width, petal length, and petal width. Illustrate: a) Comparison of three classification algorithms using the Iris dataset. Which algorithm would be your initial choice for classifying iris species? b) Techniques for evaluating the classification model's performance.	Scikit-learn, NumPy, pandas, matplotlib, seaborn	Proficiency in implementing various classification algorithms, normalisation
8	Imagine you're working on a social media platform that wants to implement a background propagation algorithm to enhance user engagement and content discovery. This algorithm aims to analyze users' interests, preferences, and behaviors to subtly introduce relevant content in the background while they browse their feed. Build an Artificial Neural Network by implementing the Back-propagation algorithm and test the same using appropriate data sets.	Python	Knowledge of activation functions, loss functions, and optimization algorithm
9	Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering.	Python Clustering Algorithms	Ability to preprocess and prepare data for clustering,
10	Imagine you're a data scientist working for a financial institution that wants to improve its credit risk assessment model. The institution wants to reduce the number of defaults on loans by accurately predicting the creditworthiness of applicants. You decide to use ensemble methods to tackle this problem. Implement following ensemble methods on a given dataset. a) Random Forest ensemble b) XGBoost	Python XGBoost	Ensemble methods
11	Develop a convolutional neural network (CNN) for image classification on a dataset of handwritten digits (such as MNIST). Illustrate to : a. Design and train a CNN architecture suitable for this task, considering factors like convolutional layers, pooling layers, and fully connected layers? b. Split the dataset into training, validation, and testing sets, as well as the choice of loss function, optimizer, and evaluation metrics c. Evaluate the performance of the CNN model using metrics like accuracy, precision, recall, and F1-score, and discuss strategies for fine-tuning the model for better results.	Tensor flow Keras	Knowledge of strategies for fine-tuning the model to improve performance, such as adjusting learning rates and adding dropout.
12	Build a LSTM model to predict the stock price of any company. Evaluate the performance of the LSTM model and compare it to a traditional machine learning model.	Tensor flow Keras	Handling Time-Series Data

Course Title	Data Engineering Lab				Course Type		HC	
Course Code	B24EH0406	Credits	1		Class		IV SEMESTER	
Course Structure	LTP	Credits	Contact	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	**1**	**2**	**2**		**28**	**25**	**25**

Prerequisites:

1. Knowledge of programming concepts in Python, Sql
2. Understanding of basic Nosql Concept.

COURSE OVERVIEW:

This lab course aims to provide hands-on experience in designing, programming, and implementing projects using python by designing, building, and managing information with relational databases, NoSQL databases, and big data infrastructure. Data Engineers are responsible for the creation and maintenance of the data infrastructure that facilitates the analysis of data by Data Scientists.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Introduce the basics of data mining, data types, similarity and dissimilarity measures

2. Illustrate the use of different SQL statements.

3. Discuss the topics such as Database Design and Normalization

4. The purpose of this course is to provide the students with the knowledge of Big data Analytics principles and techniques.

5. To learn the security of databases, design techniques of database security and learn the secure software design

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Ability to understand the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system	1,23,4,5,6,9,11,12	1,2,3
CO2	Analyze and implement Secure Operating System Design and Secure Database Management System (DBMS) Design strategies, ensuring the confidentiality, integrity, and availability of data and system resources.	1,2,3,4,5,6,8,11,12	1,2,3
CO3	Construct the database for given real world application and solve queries over it using SQL commands.	1,2,3,4,5,6,9,11,12	1,2,3
CO4	Develop an optimized database using design guidelines and normalization technique.	1,2,3,4,5,6,9,11,12	1,2,3
CO5	Ability to program using HADOOP and Map reduce, NOSQL	1,2,3,4,5,6,9,11,12	1,2,3
CO6	Demonstrate a comprehensive understanding of Transaction Management in database systems and proficient in implementing and analyzing Transaction Isolation levels.	1,2,3,4,5,6,9,11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1			✓			
CO2			✓			
CO3			✓	✓		
C04			✓	✓	✓	✓
CO5		✓	✓			
CO6			✓			

COURSE ARTICULATION MATRIX

CO# /	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PS O1**	**PS O2**	**PS O3**
CO1	3	3	3	3	3	3			3		3	3	3	2	2
CO2	3	3	3	3	3	3		3			3	3	3	2	2
CO3	3	3	3	3	3	3			3		2	3	3	2	2
CO4	2	2	2	2	2	2			3		2	2	3	2	2
CO5	2	2	2	2	2	2			2		2	2	3	2	2
CO6	3	2	3	2	2	2			2		2	3	3	2	2

Practice:

No	**Title of the Experiment**	**Tools and Techniques**	**Expected Skill/Ability**
1	Normalization in Relational Databases:Create a sample relational database schema.Normalize the schema to 3rd Normal Form,Implement the normalized schema in a SQL database	Jupyter notebook	sql/python
2	Denormalization Techniques :Analyze the performance of queries on the normalized database. Identify denormalization opportunities. Apply denormalization techniques (e.g., adding redundant columns, materialized views). Measure the performance improvement of denormalized queries.	Jupyter notebook	sql/python
3	Structured Data Processing with SQL :Write SQL queries to perform aggregation, grouping, and filtering on a dataset. Use window functions and CTEs to solve complex querying tasks. Perform joins between multiple tables to retrieve desired information. Optimize SQL queries using indexes and query tuning techniques.	Jupyter notebook	sql/python

4	Unstructured Data Processing with Python : Fetch unstructured text data from online sources or files. Use Python libraries (e.g., NLTK, spaCy) for text preprocessing, tokenization, and stemming. Perform sentiment analysis or topic modeling on the text data. Visualize the results using matplotlib or seaborn.	Jupyter notebook	python
5	XML and JSON Processing: Parse XML and JSON files using Python libraries (e.g., xml.etree.ElementTree, json). Extract relevant information from the XML/JSON data structure.Transform the data into a structured format (e.g., pandas DataFrame). Store the transformed data in a relational database or a NoSQL database.	Jupyter notebook	html/python
6	Data Integration and ETL Processes : Identify multiple data sources for integration. Develop ETL pipelines using Python (e.g., using pandas, SQLAlchemy). Extract data from the sources, transform it to fit the target schema, and load it into the destination. Schedule and automate the ETL process using tools like Apache Airflow or cron jobs.	Jupyter notebook	Sql/python
7	Data Warehousing Concepts: Design a dimensional model for a data warehousing scenario. Implement the dimensional model in a SQL database. Populate the data warehouse with sample data. Write SQL queries to perform OLAP operations (e.g., slice, dice, roll-up, drill-down).	Jupyter notebook	sql/python
8	Introduction to Big Data Technologies : Set up a Hadoop cluster using tools like Cloudera or Hortonworks. Store and process large datasets using Hadoop Distributed File System (HDFS) and MapReduce. Perform data analysis tasks using Hive or Pig. Compare the performance of Big Data processing with traditional database systems.	Jupyter notebook	Cloud computing/python
9	Capstone Project Preparation: Brainstorm project ideas based on course learnings and personal interests. Write a project proposal outlining the objectives, methodology, and expected outcomes. Receive feedback from the instructor and peers on the proposal. Refine the proposal based on the feedback received.	Jupyter notebook	python
10	Python code for creating a collection with documents, deleteng and modifying the collection operations using MongoDB. Install MongoDB: Download and install MongoDB from the official website Install PyMongo: PyMongo is the Python driver for MongoDB. You can install it using pip: Start MongoDB Server: Start the MongoDB server. Connect to MongoDB in Python Select a Database: Specify the name of the database you want to work with. If the database doesn't exist, MongoDB will create it when you first write data to it. Select a Collection: Choose a collection within the database. Collections are similar to tables in relational databases but do not enforce a schema. Perform Operations	Jupyter notebook	NoSql/Python

TEXT BOOKS:

1. "Data Engineering: A Gentle Introduction" by Dan Levine
2. "Database Systems: The Complete Book" by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom
3. "SQL Performance Explained" by Markus Winand
4. “NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" by Martin Fowler and Pramod J. Sadalage
5. "Natural Language Processing with Python" by Steven Bird, Ewan Klein, and Edward Loper

REFERENCE BOOKS:

1. XML and JSON Recipes for SQL Server: A Problem-Solution Approach" by Alex Grinberg

2. “Building Data Pipelines with Python" by Mohammad R. Rezae

JOURNALS/MAGAZINES:

1. A Distributed Solution For Efficient K Shortest Paths Computation Over Dynamic Road Networks.Ziqiang Yu;Xiaohui Yu;Nick Koudas;Yueting Chen;Yang Liu
2. A Neural Database for Answering Aggregate Queries on Incomplete Relational Data Sepanta Zeighami;Raghav Seshadri;Cyrus Shahabi
3. A Generic Schema Evolution Approach For NoSQL And Relational Databases Alberto Hernández Chillón;Meike Klettke;Diego Sevilla Ruiz;Jesús García Molina

Course Title	Course based Project				Course Type		SDC	
Course Code	B24EH0407	Credits	1		Class		IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	1	2	2	-	28	25	25

COURSE OVERVIEW:

This course emphasizes hands-on learning through a series of projects. Students will apply theoretical knowledge to practical tasks, developing skills in problem-solving, critical thinking, collaboration, and communication. The projects will cover various topics within the subject and will require students to use tools and techniques relevant to the field.

COURSE OBJECTIVES:

The objectives of this course are to

1. Apply theoretical concepts to practical scenarios.
2. Develop project management and organizational skills.
3. Enhance problem-solving and critical thinking abilities.
4. Collaborate effectively with peers.
5. Communicate findings and results clearly.

COURSE OUTCOMES (CO'S):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project work to attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing the project work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysis required for carrying out the project.	11	1,2,3

CO12	Engage in lifelong learning to improve knowledge and competence in the chosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATIONMATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3
CO12												3	3	3	3

Note: 1-Low, 2-Medium, 3-High

Project Description:

The objective of this project is to design, implement, and analyse a range of algorithms to solve specific computational problems. The project aims to develop a deep understanding of algorithmic design principles, efficiency analysis, and practical applications.

List of Sample Projects:

1. E-commerce Product Recommendation System Problem Statement:

Develop a product recommendation system for an e-commerce platform to suggest relevant products to users based on their browsing and purchasing history.

Implementation Description:

- **User-Item Matrix:** Represent user preferences with a matrix where rows represent users and columns represent items.
- **Data Structures:**
  - **Matrix:** Store user ratings and interactions with products.
  - **Hash Map:** Map user IDs and product IDs to their respective indices in the matrix.
- **Algorithm:** Implement collaborative filtering techniques (user-based or item-based) and matrix factorization algorithms like Singular Value Decomposition (SVD) to generate recommendations.

2. Dynamic Memory Allocation Problem Statement:

Implement a dynamic memory allocator that efficiently handles memory allocation and deallocation requests.

Implementation Description:

- **Memory Representation:** Manage a large contiguous block of memory.
- **Data Structures:**
  - **Linked List:** Use a free list to manage free memory blocks.
  - **Heap:** Implement a heap to manage allocated memory.
- **Algorithm:** Implement algorithms for first-fit, best-fit, and worst-fit memory allocation strategies. Optimize deallocation to minimize fragmentation.

3. Real-Time Chat Application Problem Statement:

Develop a real-time chat application that supports multiple users and ensures efficient message delivery and storage.

Implementation Description:

- **Message Representation:** Represent messages with unique IDs, timestamps, and user IDs.
- **Data Structures:**
  - **Queue:** Handle incoming and outgoing messages.
  - **Hash Map:** Store user sessions and message histories.
- **Algorithm:** Implement efficient message routing algorithms and data structures to handle high concurrency and ensure real-time communication.

4. Resource Allocation in Cloud Computing Problem Statement:

Create a system for efficient resource allocation and management in a cloud computing environment.

Implementation Description:

- **Resource Representation:** Model resources (CPU, memory, storage) as entities with capacities.
- **Data Structures:**
  - **Priority Queue:** Manage resource allocation requests based on priority.
  - **Hash Map:** Store resource usage statistics.
- **Algorithm:** Implement resource allocation algorithms (First-Come, First-Served, Round Robin, Priority Scheduling) to optimize resource utilization and ensure fairness.

Course Title	Professional Ethics				Course Type		FC	
Course Code	B24CS0401	Credits	1		Class		IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	1	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	-	-	-				
	Total	1	1	1	14		25	25

COURSE OVERVIEW:

The course enables e the students to imbibe and internalize the Values and Ethical Behaviour in the personal and Professional lives.

COURSE OBJECTIVE S:

1. Understand the professional Rules of conduct for Engineers.
2. Appreciate codes of conduct, professional Rules of conduct.
3. Recognize the conflict of interest and Develop strategies
4. Understand the importance of communication with all stakeholders.
5. Apply practical strategies for handling ethical dilemmas.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understanding basic purpose of profession, professional ethics and various moral and social issues.	8,9,10	
CO2	Awareness of professional rights and responsibilities of a Engineer, safety and risk benefit analysis of a Engineer	8,9,10	
CO3	Acquiring knowledge of various roles of Engineer In applying ethical principles at various professional levels	8,9,10	
CO4	Professional Ethical values and contemporary issues	8,9,10	
CO5	Apply practical strategies for handling ethical dilemmas	8,9,10	
CO6	Appreciate codes of conduct, professional Rules of conduct	8,9,10	

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√	√	√			
CO2	√	√	√			
CO3	√	√	√			
CO4	√	√	√			
CO5	√	√	√			
CO6	√	√	√			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1								3	2	2					
CO2								3	2	2					
CO3								3	2	2					
CO4								3	2	2					
CO5								3	2	2					
CO6								3	2	2					

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Introduction to Professional Ethics: Basic Concepts, Governing Ethics, Personal & Professional Ethics, Ethical Dilemmas, Life Skills, Emotional Intelligence, Thoughts of Ethics, Value Education, Dimensions of Ethics, Profession and professionalism, Professional Associations, Professional Risks, Professional Accountabilities, Professional Success, Ethics and Profession.

Basic Theories: Basic Ethical Principles, Moral Developments, Deontology, Utilitarianism, Virtue Theory, Rights Theory, Casuist Theory, Moral Absolution, Moral Rationalism, Moral Pluralism, Ethical Egoism, Feminist Consequentialism, Moral Issues, Moral Dilemmas, Moral Autonomy.

Professional Practices in Engineering: Professions and Norms of Professional Conduct, Norms of Professional Conduct vs. Profession; Responsibilities, Obligations and Moral Values in universa, Professional codes of ethics, the limits of predictability and responsibilities of the engineering profession.

Work Place Rights & Responsibilities, Ethics in changing domains of Research, Engineers and Managers; Organizational Complaint Procedure, difference of Professional Judgment within the Nuclear Regulatory Commission (NRC), the Hanford Nuclear Reservation.

TEXT BOOKS:

1. Professional Ethics: R. Subramanian, OxfordUniversityPress,2015.

2. Ethics in Engineering Practice &Research, CarolineWhitbeck,2e, Cambridge University Press 2015.

REFERENCE BOOKS:

1. Engineering Ethics, Concepts Cases: Charles E Harris Jr.,Michael S Pritchard, Michael J Rabins, 4e , Cengage learning, 2015.

2. Business Ethics concepts & Cases: ManuelGVelasquez,6e,PHI, 2008.

Course Title	Universal Human Values				Course Type		HC	
Course Code	B24EE0310	Credits	1		Class		IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	1	1	1				
	Tutorial	0	0	0	Theory	Practical	IA	SEE
	Practice	0	0	0				
	Total	1	1	1	14	0	25	25

COURSE OBJECTIVE

1. Development of a holistic perspective based on self- exploration about themselves (human being), family,society and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
3. Strengthening of self-reflection.
4. Development of commitment and courage to act

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand the significance of value inputs in a classroom and start applyingthem in their life and profession.	6,7,8	1
CO2	Distinguish between values and skills, happiness and accumulation of physical facilities, the Self and the Body, Intention and Competence of an individual, etc.	6,7,8	1
CO3	Understand the role of a human being in ensuring harmony in society and nature.	6,7,8	1
CO4	Demonstrate the role of human being in the abetment of pollution.	6,7,8	1
CO5	Describe appropriate technologies for the safety and security of the society as responsible human being.	6,7,8	1
CO6	Distinguish between ethical and unethical practices and start working out the strategy to actualize a harmonious environment wherever they work.	6,7,8	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyse (L4)	Evaluate (L5)	Create (L6)
CO1		√	√			
CO2		√	√			
CO3		√	√			
CO4		√	√			
CO5		√	√			
CO6		√	√			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						3	2	1					1	
CO2						3	2	1					1	
CO3						3	2	1					1	
CO4						3	2	1					1	
CO5						3	2	1					1	
CO6						3	2	1					1	

Note: 1-Low, 2-medium, 3-High

COURSE CONTENT

Happiness and Prosperity- A look at basic Human Aspirations. Right understanding, Relationship, basic requirements for fulfilment of aspirations of every human being with their correct priority. Understanding Happiness and Prosperity correctly, Method to fulfil human aspirations: understanding and living in harmony at various levels, Understanding human being as a co-existence of the sentient 'I' and the material 'Body'. Understanding the needs ofSelf ('I') and 'Body' - happiness and physical facility. Understanding the Body as an instrument of 'I' (I being the doer, seeker and enjoyer). Understanding the characteristics and activities of 'I' and harmony in 'I'. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail.

Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship. Understanding the meaning of Trust; Difference between intention and competence. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Understanding the harmony in the Nature. Interconnectedness and mutual fulfilment among the four orders of nature-recyclability and self-regulation in nature. Understanding Existence as Co-existence of mutually interacting units in all-pervasive space. Holistic perception of harmony at all levels of existence. Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

Natural acceptance of human values. Definitiveness of Ethical Human Conduct. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for above production systems. Case studies of typical holistic technologies, management models and production systems. Strategy for transition from the present state to Universal Human Order: a) At the level of individual: as socially and ecologically responsible engineers, technologists and managers b) At the level of society: as mutually enriching institutions and organizations

Text Book:

1. R R Gaur, R Sangal, G P Bagaria, Human Values and Professional Ethics, Excel Books, New Delhi, 2010
2. A.N Tripathy, Human Values, New Age Intl. Publishers, New Delhi, 2004.
3. R.R. Gaur, R. Sangal and G.P. Bagaria, A Foundation Course in Human Values and Professional Ethics, Excel Books, New Delhi, 2010
4. Bertrand Russell, Human Society in Ethics & Politics, Routledge Publishers, London, 1992

Reference Books:

1. Corliss Lamont, Philosophy of Humanism, Humanist Press, London, 1997

2.I.C. Sharma, Ethical Philosophy of India Nagin & co Julundhar, 1970

3.Mohandas Karamchand Gandhi, The Story of My Experiments with Truth, Navajivan Mudranalaya, Ahmadabad, 1993.

Course Title	Environmental Science				Course Type		MC	
Course Code	B24AS0403	Credits	0		Class		IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Theory	0	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	**0**	**1**	**1**	**14**	**-**		

COURSE OVERVIEW:

Environmental Science is focussed on a holistic understanding of earth systems in order to learn from the past, comprehend the present and influence the future. It is the study of how physical, chemical and biological processes maintain and interact with life, and includes the study of how humans affect nature. As environmental science is at the cross-roads of the natural sciences, it provides an enriching alternative to a single-subject honours degree, and can open the door to an exciting range of career options. This approach enables us to tackle necessary problems, such as ensuring that human needs are met in a sustainable way, so that everyone has access to clean water and air, and the resources required for agriculture and industrial activity.

COURSE OBJECTIVES:

The objectives of this course are to:

1. Familiar with current and emerging environmental engineering and global issues and have an understanding of ethical and societal responsibilities.
2. Recognize the need for engaging in life-long learning.
3. Study various types of energy (conventional & non-conventional) resources and natural resources.
4. Acquire knowledge with respect to biodiversity, threats, conservation and appreciate the concept of ecosystem.
5. Know about sources, effects and control measures of environmental pollution, degradation, and waste management.
6. Explore the ways for protecting the environment.

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand, analyse and execute favourable environmental conditions and the role of individual, government and NGO in environmental protection.	7,9,10	**1**
CO2	List the causes, effects & remedial measures of environmental pollution, degradation & find ways to overcome them by suggesting the pollution controlled products.	7,9,10	**1**
CO3	Get motivation to find new renewable energy resources with high efficiency through active research and innovation.	7,9,10	**1**
CO4	Critically analyse the ecological imbalances and provide recommendations to protect the environment.	7,9,10	**1**
CO5	Explore the condition of environmental degradation and waste management techniques and take promising measures to make our environment eco- friendly.	7,9,10	**1**
CO6	Identify new methodologies for conservation of our natural resources and ecosystem.	7,9,10	**1**

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	✓	✓				
CO2	✓	✓				
CO3	✓					
CO4	✓					
CO5	✓	✓				
CO6	✓					

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1							2		2	2			1		
CO2							2		2	2			1		
CO3							3		2	2			1		
CO4							3		2	2			1		
CO5							2		2	2			1		
CO6							3		2	2			1		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENTS:

Environment and Environmental Protection

Basics of environment: Introduction & definition to Environment, objectives and guiding principles of environmental education, Components of environment, Structure of atmosphere, Sustainable environment/Development, Impact of technology on the environment in terms of modern agricultural practices and industrialization, Environmental Impact Assessment.

Environmental protection: Role of Government - Assignments of MOEF, Functions of central and state boards, Institutions in Environment and People in Environment, Environmental Legislations.

Environmental pollution, degradation & Waste management:

Environmental Pollution: Definition, sources and types, Pollutant-Definition & classification, Concepts of air pollution, water pollution, Automobile Pollution-Causes, Effects & control measures.

Environmental degradation: Introduction, Global warming and greenhouse effect, Acid rain-formation & effects, Ozone depletion in stratosphere and its effect.

Waste management: Municipal solid waste, Bio-medical waste and Electronic waste (E-Waste).

Energy & Natural resources:

Energy: Conventional/Non-renewable sources – Fossil fuels based (Coal, petroleum & natural gas), nuclear energy, Non- conventional/renewable sources – Solar, wind, hydro, biogas, biomass, Hydrogen as an alternative as a future

source of energy.
Natural resources: Water resource - Global water resource distribution, Water conservation methods, Water quality parameters, Uses of water and its importance. Forest wealth - Importances, Deforestation-Causes, effects and controlling measures

Ecology, ecosystem & field work:
Ecology - Definition, branches, objectives and classification, Concept of an ecosystem – Structure and functions, Components of ecosystem-abiotic and biotic.
Levels of biological diversity: Genetic, species and ecosystem diversity; Biogeography zones of India; Biodiversity patterns and global biodiversity hot spots. India as a mega-biodiversity nation; Endangered and endemic species of India. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity.

Field work: Visit to waste water/sewage treatment plant (STP) and biogas plant at REVA university campus, and/or Visit to a local polluted site-Urban/Rural/Industrial/Agricultural.

TEXT BOOKS:
1. R.J. Ranjit Daniels and Jagadish Krishnaswamy, "Environmental Studies", Wiley India Private Ltd., New Delhi, Co-authored & Customised by Dr.MS Reddy & Chandrashekar, REVA University, 1st Edition, 2017.

2. R.J. Ranjit Daniels and Jagadish Krishnaswamy, "Environmental Studies", Wiley India Private Ltd., New Delhi, 2nd Edition, 2014.

3. Benny Joseph, "Environmental Studies", Tata McGraw – Hill Publishing Company Limited, 2nd Edition, 2008.
4. Dr.S.M.Prakash, "Environmental Studies", Elite Publishers, Mangalore, 2nd Edition, 2009.

5. Rajagopalan R, "Environmental Studies – from Crisis to cure", Oxford University Press, 3rd Edition, 2016.

Links for EVS Online resources

Link for online	Title of the course	Course Duration
https://www.classcentral.com/course/swayam-environmental-studies-14042	**Environmental Studies**	**12 Weeks Free Online**
https://www.edx.org/course/introduction-to-environmental-science- 2?index=product&search_index=product&webview=false&campaign=Introduction+to+Environmental+Science&source=edX&product_category=course&placement_url=https%3A%2F%2Fwww.edx.org%2Flearn%2Fenvironmental-science	**Introduction to Environmental Sciences**	**5hrs/Week For Weeks**
https://www.coursera.org/specializations/environmental- science?action=enroll	**Introduction to Environmental Science Specialization**	**5hrs/Week 12 Weeks**

Course Title	AEC-2(Ability Enhancement Course Placement Training)				Course Type		AEC	
Course Code	B24EF0409	Credits	1		Class		IV Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Theory	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	**1**	**2**	**2**	**-**	**28**	**25**	**25**

COURSE OVERVIEW

Placement training classes can effectively be conducted as a part of an Ability Enhancement Course (AEC). These courses are designed to enhance students' skills, making them more prepared and competitive for the job market.

V SEMESTER

V Semester												
Sl. No	Course Code	Title of the Course	HC/FC /SC/OE/MC	Credit Pattern				Contact Hours /Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0501	Feature Engineering with Python	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0502	Image processing	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0503	Big data Analytics	HC	3	0	0	3	3	50	50	100	PCC
4	B24EHS51X	Professional Elective-3	SC	3	0	0	3	3	50	50	100	PEC
5	B24EHS52X	Professional Elective-4	SC	3	0	0	3	3	50	50	100	PEC
6	B24EFO5XX	Open Elective-1	OE	3	0	0	3	3	50	50	100	OE
7	B24EH0504	Feature Engineering with Python Lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0505	Image processing lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0506	Big data Analytics lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EF0507	AEC-3(Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
11	B24ED0501	Indian Knowledge System	MC	1	0	0	0	1	25	25	50	HSMC
TOTAL				19	0	4	22	27	425	425	850	
TOTAL SEMESTER CREDITS				22								
TOTAL CUMULATIVE CREDITS				111								
TOTAL CONTACT HOURS				27								
TOTAL MARKS				850								

Course Title	Feature Engineering with Python				Course Type		HC	
Course Code	B24EH0501	Credits	3		Class		V SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	**3**	**3**	**3**	**42**		**50**	**50**

COURSE OVERVIEW:

It is designed to equip students with the essential techniques and tools needed to transform raw data into meaningful features for machine learning models. This course covers a wide range of topics, from statistical methods and data manipulation with pandas to advanced feature extraction techniques for text and images. Students will learn how to preprocess, engineer, and optimize features to improve model performance.

COURSE OBJECTIVE (S):

The objective of this course are to

1. Understand statistical concepts and methods
2. Apply pandas data structures for data manipulation
3. Implement machine learning feature engineering techniques
4. Extract features from text data
5. Develop image feature extraction models using deep learning
6. Build and improve recommendation models based on user requirements

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain the significance of statistical methods like PCA, ANOVA, and maximum likelihood estimators in the context of feature engineering.	1-3	1
CO2	Analyze a dataset to identify trends and relationships using correlation, covariance, and filtering techniques with pandas.	4-5	2
CO3	Apply feature scaling techniques (e.g., Bag-of-Words to Tf-Idf) and encode categorical variables to prepare data for machine learning models.	4-5	2
CO4	Implement a convolutional neural network (CNN) to extract features from image data and evaluate its performance.	7-8	2
CO5	Construct a recommendation model using feature extraction and deep learning techniques, integrating data import, cleaning, and parsing steps.	9-10	3
CO6	Assess the recommendation model's performance and make improvements based on user feedback and requirements.	10-11	3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**pply (L3)**	**nalyze (L4)**	**aluate (L5)**	**reate (L6)**
CO1	√					
CO2		√				
CO3			√			
CO4				√		
CO5						√
CO6					√	

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	3	3										3	2	1
CO2		2	2	2	2								2	2	2
CO3			3	3	3								2	2	2
CO4							2	2					3	3	3
CO5									3	3			2	2	2
CO6											2	2	1	2	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

STATISTICS: Distributions for description: Moments, Sample distributions, Using the sample distributions, non-parametric description, Principal component analysis, Discrete variables, Multilevel modeling, The Central Limit Theorem, Gaussian family, testing a hypothesis, ANOVA, Regression, Maximum likelihood estimators, missing data, Testing with likelihoods. [T1]

Introduction to pandas Data Structure: Series, Data Frame, Index Objects, Reindexing, dropping entries from an axis, Indexing, selection and filtering, sorting and ranking, correlation and covariance, Handling Missing Data [T2]

The Machine Learning pipeline, Text Data: Flattening, Filtering, and Chunking, The Effects of Feature Scaling: From Bag-of-Words to Tf-Idf, Categorical Variables, Encoding Categorical Variables, Dealing with Large Categorical Variables, Dimensionality Reduction: PCA, Nonlinear Featurization using K-Means Model Stacking [T3]

Image Feature Extraction and Deep Learning, Fully Connected Layers, Convolutional Layers, Rectified Linear Unit (ReLU) Transformation, Response Normalization Layers, Pooling Layers, Structure of AlexNet, Application: Building a Recommended Model, Data Import, Cleaning, and Feature Parsing and improvements based on the user requirement [T3]

Self Learning Component: Satellite Images, Exploratory Data Analysis, Pixels as Features, Automatic Dataset Expansion, Histograms, Error Analysis, Computer Vision, Color Manipulation, Edge Detection, Image Smoothing

TEXT BOOKS:

1. Klemens, Ben. Modeling with data: tools and techniques for scientific computing. Princeton University Press, 2008.
2. McKinney, Wes. Python for data analysis. " O'Reilly Media, Inc.", 2022.
3. Zheng, Alice, and Amanda Casari. Feature engineering for machine learning: principles and techniques for data scientists. " O'Reilly Media, Inc.", 2018.

REFERENCE BOOKS:

1. Duboue, Pablo. *The art of feature engineering: essentials for machine learning*. Cambridge University Press, 2020.

JOURNALS/MAGAZINES:

1. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)
2. Journal of Data Science
3. Data Mining and Knowledge Discovery
4. Pattern Recognition Letters
5. International Journal of Computer Vision (IJCV)

SWAYAMNPTEL/MOOCs

1. NPTEL Course: "Data Science for Engineers"
2. Instructor: Prof. Raghunathan Rengasamy, IIT Madras
3. NPTEL Course: "Python for Data Science"
4. Instructor: Prof. Ragunathan Rengasamy, IIT Madras
5. Coursera: "Feature Engineering for Machine Learning"
6. Instructor: Dr. Dan Becker

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

Course Title	Image Processing				Course Type		HC	
Course Code	3	HC	V Semester		Class		V SEMESTER	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42		50	50

Prerequisites: Basic probability theory and programming techniques.

COURSE OVERVIEW:

This course gives comprehensive study of basic image fundamentals, image enhancement using spatial domain and frequency domain. It also covers about the image restoration, image degradation model, image segmentation thresholding, region - oriented segmentation, and color image processing. Enormous applications of digital image processing are covered to make it a multidisciplinary course.

COURSE OBJECTIVE (S):

The objectives of the course are:

1. Discuss the fundamental steps of Image Processing
2. Explain different filtering operations in spatial domain for smoothening and sharpening the images
3. Describe the concept of Image restoration and image enhancement
4. Inculcate knowledge about fundamentals of color images and Image Segmentation to employ image processing techniques for various applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Make use of the fundamental steps in digital Image Processing for Zooming and Shrinking Digital Images.	1 to 5	1, 2,3
CO2	Utilize different filtering operations in spatial and frequency domain for smoothening and sharpening the images.	1 to 5	1, 2,3
CO3	Develop real world applications using Filtering techniques for image restoration and compression.	1 to 5	1, 2,3
CO4	Apply color image, Morphological and image segmentation in real world applications.	1 to 5	1, 2,3
CO5	Understand the role of image processing in different fields such as medical, engineering, space, biotechnology, ocean, agriculture, food industry, etc.	1 to 5	1, 2,3
CO6	Analyze a wide range of problems and provide solutions related to the design of image processing systems through suitable algorithms, structures, diagrams, and other appropriate methods.	1 to 5	1, 2,3

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5		√				
CO6				√		

COURSE ARTICULATION MATRIX:

CO #/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	2	2	3	3								3	2	2
CO 2	3	3	2	3	3								3	2	2
CO 3	3	1	2	3	2								3	2	2
CO 4	3	3	2	3	3								3	2	2
CO 5	3	2	3	2	3				2				3	2	1
CO 6									2				3	2	1

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction: What is Digital Image Processing, Origins of Digital Image Processing, Examples or applications of fields that use DIP, Fundamental Steps in Digital Image Processing, Components of an Image Processing System. Digital Image Fundamentals: Image Sampling and Quantization, Zooming and Shrinking Digital Images, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations. Modern computing hardware systems (GPU, cloud computing) and software libraries (OpenCV, Matlab, Python) for image processing.

Image Enhancement in Spatial and Frequency Domain, Transforms: Basic Intensity Transformation Functions, Histogram Processing, Fundamentals of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods. Introduction to the Fourier Transform and the Frequency Domain, Image Smoothing and Sharpening using Frequency Domain Filters.

Image Transforms**:** Discrete Fourier Transform, Discrete Cosine Transform, Discrete Wavelet Transformation, Singular Value Decomposition and Principal Component Analysis.

Image Restoration and Image Compression:
Image Restoration: A Model of the Image degradation/Restoration process, Noise Models, Restoration in the Presence of Noise Only-Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering.

Image Compression:
Fundamentals of image compression and types of redundancy, Lossless and lossy compression methods: Huffman coding, LZW coding, Block Transform Coding, Wavelet Coding.

Color Image, Morphological and Segmentation
Color Models: RGB, CMY, HSI, YCbCr, Color Transformations, Smoothing and Sharpening, Image Segmentation based on color, Noise in Color Images, Color Image Compression.

Morphological Image Processing: Erosion and Dilation , Opening and Closing, the Hit-or-Miss Transformation, Morphological Algorithms – Boundary Extraction, Region Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening.

Image Segmentation: Point, Line and Edge Detection, , Region-Based Segmentation.

SELF-LEARNING COMPONENT

Thresholding, Region-Based Segmentation.

TEXTBOOKS:

1. Rafael C Gonzalez, Richard E. Woods, "Digital Image Processing", Pearson Education, 4th Edition, 2019.
2. K. Jain, "Fundamentals of Digital Image Processing", Pearson, 3rd Edition, 2004.

REFERENCE BOOKS:

1. Alasdair Mc Andrew; Introduction to Digital Image Processing; Cengage learning; 2009.
2. S. Jayaraman, S. Esakkirajan, T.Veerakumar, "Digital Image Processing", McGraw-Hill Education. (India) Pvt. Ltd., 2013.
3. Scott. E. Umbaugh, "Digital Image Processing and Analysis", CRC Press, 2nd Edition, 2014.

JOURNALS/MAGAZINES:

1. IEEE transactions on Image processing
2. ACM Transactions on Image processing
3. Journal of Visual Communication and Image Representation

SWAYAM NPTEL/MOOCs:

1. **https://onlinecourses.nptel.ac.in/noc24_ee133/preview**
2. **https://www.coursera.org/learn/introduction-image-processing**
3. **https://www.coursera.org/specializations/image-processing**
4. **https://www.udemy.com/course/image-processing-from-ground-uptm-in-c/?couponCode=UPGRADE02223**

Web Sites

1. www.imageprocessingplace.com - Text Book website
2. www.mathworks.com – - (MATLAB Documentation)
3. en.wikipedia.org/wiki/Digital image processing – General Image Processing Concepts
4. www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/getstart.pdf

(MATLAB Getting Started Guide from Mathworks)

Course Title	Big Data Analytics				Course Type		HC	
Course Code	B24EH0503	Credits	3		Class		V Semester	
Course Structure	LTP	Credits	ontact Hours	Work Load	Total Number of Classes Per Semester		ssessment in Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-	Theor y	Practica l	CIE	SEE
	Practical	-	-	-				
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

This course introduces the concept of big data and provides a practical foundation level exposure that enables students to participate in big data application development. The course provides grounding in basic and advanced methods to big data technology with Apache Spark fundamentals.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the concepts of Big Data and its Business Implications.
2. Illustrate the fundamentals of functional programming for Big-Data Analytics.
3. Exemplifying the features of Apache Spark for Data Analytics.
4. Implement Machine Learning in Spark to solve real business problems.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Outline the concepts of Big Data and its Business Implications.	1 to 5,9,10,12	1
CO2	Apply the concept of functional distributed programming to Big Data Analytics applications.	1 to 5, 9,10,12	1,2, 3
CO3	Understand the fundamentals of Apache Spark and Spark Core.	1 to 5, 9,10,12	1, 3
CO4	Design a Data Analytics Framework using Apache Spark with Python.	1 to 5, 9,10,12	1,2,3
CO5	Implement distributed Machine Learning Applications for big data using Spark.	1 to 5, 9,10,12	1,2,3
CO6	Develop real world big data application using Apache Spark with Python programming.	1 to 5, 9,10,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	√					
CO2			√			
CO3		√				
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

CO #/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	3	1	3	3	-	-	-	-	-	-	-	3	1	3
CO 2	3	3	3	3	3	-	-	-	3	-	-	2	3	3	3
CO 3	3	1	3	1	3	-	-	-	3	3	3	3	3	3	3
CO 4	3	3	3	3	3	-	-	-	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	-	-	-	3	3	3	3	3	3	3
CO 6	3	3	3	3	3	1	-	1	3	3	3	3	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introducing Big Data, Hadoop, and Spark: Introduction to Big Data, Distributed Computing, and Hadoop, A Brief History of Big Data and Hadoop, Hadoop Explained; Introduction to Apache Spark: Apache SparkBackground, Uses for Spark, Programming Interfaces to Spark, Submission Types for Spark Programs, Input/Output Types for Spark Applications, The Spark RDD, Spark and Hadoop; Functional Programming Using Python: Data Structures Used in Functional Python Programming, Python Object Serialization, Python Functional Programming Basics; **NOSQL Databases and Big Data Storage Systems:** Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j.

Understanding the Spark Cluster Architecture: Anatomy of a Spark Application:Spark Driver, Spark Workers and Executors, The Spark Master and Cluster Manager; Spark Applications Using the Standalone Scheduler: Spark Applications Running on YARN Deployment; Modes for Spark Applications Running on YARN: Client Mode, Cluster Mode, Local Mode Revisited; Learning Spark Programming Basics: Introduction to RDDs,Loading Data into RDDs, Creating an RDD from a File or Files, Methods for Creating RDDs from a Text File or Files, Creating an RDD from an Object File, Creating an RDD from a Data Source, Creating RDDs from JSON Files, Creating an RDD Programmatically;

Operations on RDDs: Key RDD Concepts, Basic RDD Transformations, Basic RDD Actions, Transformations on PairRDDs, MapReduce and Word Count Exercise, Join Transformations, Joining Datasets in Spark, Transformations on Sets, Transformations on Numeric RDDs; Advanced Programming Using the Spark Core API:Shared Variables in Spark: Broadcast Variables, Accumulators, Exercise: Using Broadcast Variables and Accumulators; Partitioning Data in Spark: Partitioning Overview, Controlling Partitions, Repartitioning Functions, Partition-Specific or Partition-Aware API Methods; RDD Storage Options: RDD Lineage Revisited, RDD Storage Options, RDD Caching, Persisting RDDs, Choosing When to Persist or Cache RDDs, Checkpointing RDDs, Exercise:Checkpointing RDDs;

Processing RDDs with External Programs: Data Sampling with Spark; Understanding Spark Application and Cluster Configuration: Spark Environment Variables, Spark Configuration Properties; Optimizing Spark: Filter Early, Filter Often, Optimizing Associative Operations, Understanding the Impact of Functions and Closures, Considerations for Collecting Data, Configuration Parameters for Tuning and Optimizing Applications, Avoiding Inefficient Partitioning, Diagnosing Application Performance Issues; Machine Learning with Spark: Machine Learning Primer, Machine Learning Using Spark MLlib, Exercise: Implementing a Recommender Using Spark MLlib, Machine Learning Using Spark ML;

TEXT BOOKS:

1. Aven, Jeffrey. Data Analytics with Spark Using Python. Addison-Wesley Professional, 2018.

2. Bengfort, Benjamin, and Jenny Kim. Data analytics with Hadoop: an introduction for data scientists."O'Reilly Media, Inc.", 2016.

3. Sridhar Alla, Big Data Analytics with Hadoop 3, published by Packt Publishing Ltd, May 2018

4. SubhashiniChellappan, DharanitharanGanesan, Practical Apache Spark Using the Scala API, A Press, 2018.

REFERENCE BOOKS:

1. Michael Minelli, Michele chambers, AmbigaDhiraj: Big data, big analytics, Wiley, 2013.

2. P. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Addison-Wesley, 2005.

3. J. Han, M. Kamber, Data Mining: Concepts and Techniques, 2nd ed. Morgan Kaufmann, 2005.

JOURNALS/MAGAZINES:

1. IEEE,Introduction to the IEEE Transactions on Big Data

2. Elsevier,Big data research journal Elsevier

3. Springer, Journal on Big Data Springer.

4. ACM DL,The Journal of Machine Learning Research-ACM

SWAYAM/NPTEL/MOOCs:

1. https://nptel.ac.in/courses/106104189

2. https://onlinecourses.nptel.ac.in/noc20_cs92/preview

3. https://iisc.talentsprint.com/cds/?utm_source=googlesearch&utm_medium=tcpa&utm_campaign=ts- googlesearch-iisc-cds-tcpa-similar-audiences-roi&utm_content=data-science- program&utm_term=Analytics%20training %20courses&gclid=Cj0KCQjw8uOWBhDXARIsAOxKJ2HfWKGqq pZ7Gm0dBPDgkwWaj0BqSZBzuWvcqmbF5AOvYvxYSB5lvFcaAlO_EALw_wcB

4. https://www.simplilearn.com/pgp-data-engineering-certification-training- course? utm_source=google&utm_medium=cpc&utm_term=big%20data%20analysis%20online%20cours e&utm_content=11233548673-137256573987-586850068745&utm_device=c&utm_campaign=Search- DataCluster-PG-BigData-CDE-Purdue-IN-Main-AllDevice-adgroup-SEMR-09Mar2022-BD-Analysis-Course-Phrase&gclid=Cj0KCQjw8uOWBhDXARIsAOxKJ2GINP8K1hXBRF4Qu6cWTaZ714GQYvQR_CL4ff9qM6whNg

   _Xx2vLFmcaAjMTEALw_wcB

SELF-LEARNING EXERCISES:

1. SQL and NoSQL Programming with Spark

2. Stream Processing and Messaging Using Spark

Professional Elective 3

Course Title	REINFORCEMENT LEARNING				Course Type		SC	
Course Code	B24EHS511	Credits	3		Class		V SEMESTER	
	LTP	Credits	ontact Hours	Worklo ad	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	**3**	**3**	**3**	**42**		**50**	**50**

COURSE OVERVIEW:

The field of reinforcement learning (RL) is rapidly evolving, and numerous courses are available for learners at different levels. Here's an overview of some prominent courses that cover various aspects of reinforcement learning.

COURSE OBJECTIVE (S):

The Objective of this course are to

1. Understand the fundamental concepts of Reinforcement Learning (RL):
2. Analyze mathematical and algorithmic foundations of RL:
3. Implement policy-based learning techniques.
4. Apply Q-learning and related algorithms.
5. Utilize dynamic programming in RL.
6. Connect reinforcement learning to other machine learning paradigms.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain the basic concepts of reinforcement learning and to Formulate RL problems as MDPs	1 to 5	1
CO2	Analyze the structure and function of Markov Decision Processes, including their mathematical foundations and key components.	1 to 5	2
CO3	Implement and evaluate deterministic and stochastic policies using the Bellman equation within the RL framework.	1 to 5	1,2
CO4	Apply Q-learning algorithms to solve practical problems, demonstrating an understanding of their terms, notations, and case study applications.	1 to 5	2,3
CO5	Utilize dynamic programming methods to derive optimal policies and value functions in reinforcement learning scenarios.	1 to 5	2
CO6	Critically assess the applications, benefits, and challenges of reinforcement learning, understanding the exploration vs. exploitation trade-off.	1 to 5	2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**pply (L3)**	**nalyze (L4)**	**aluate (L5)**	**reate (L6)**
CO1		➹				
CO2		➹				
CO3			➹			
CO4			➹			
CO5				➹		
CO6				➹		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	2	2	2	2	-	-	-	-	-	-	1	3	1	-
CO2	2	3	2	1	2	-	-	-	-	-	-	1	2	3	2
CO3	2	3	3	2	2	-	-	-	-	-	-	1	3	3	-
CO4	3	3	3	2	2	-	-	-	-	-	-	1	2	3	3
CO5	2	3	2	2	2	-	-	-	-	-	-	1	3	3	-
CO6	3	3	2	2	2	-	-	-	-	-	-	1	3	3	-

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT:

Introduction to Reinforcement Learning: Early Roots of Reinforcement Learning, Need for Reinforcement Learning , Difference Between Reinforcement Learning, Supervised Learning and Unsupervised Learning, Terms Used in Reinforcement Learning: Elements of Reinforcement Learning, Types of Environments, Working of Reinforcement Learning, Approaches To Implementing Reinforcement Learning, Types of Reinforcement Learning, Exploration Vs Exploitation Trade-Off, Advantages of Reinforcement Learning, Applications of Reinforcement Learning, Challenges with Reinforcement Learning, Reinforcement Learning Vs Deep Learning Vs Machine Learning.

Markov Decision Process: Understanding MDP: Key Components Of MDP, Formal Definition Of MDP**;** Markov Property**:** Markov Chain**,** Key Concepts and Analyses Associated with Markov Chains, Markov Model**,** Markov Matrix**,** Using Markov Matrices in Machine Learning; Markov Decision Process Framework: Reward and Returns**,** Markov Reward Process; Value Function in MDP: State Value Function, Action Value Function**;** Optimal Policy In MDP**,** Examples of MDP

Policy In MDP, Q-Learning: Introduction, Types of Policy: Deterministic Policy, Stochastic Policy; Bellman Equation for Value Function, Recycling Robot: Case Study, Optimal Policy and Value Function, Backup Diagram, Overview Of Q-Learning, Terms in Q Learning with Notations, Working Of Q-Learning, Gambler in A Dungeon Example, Application of Q Learning, Case Study Examples of Q Learning

Dynamic Programming: Introduction: Top-Down Dynamic Programming, Bottom-Up Dynamic Programming, Dynamic Programming to Obtain Optimal Policy And Optimal Value Function In Reinforcement Learning: Policy Iteration, Value Iteration; Bellman-Ford Single Source Shortest Path Algorithm, Policy In Dynamic Programming, Policy Evaluation And Policy Control, Dynamic Programming For Policy Prediction / Evaluation And Policy Control, Iterative Policy Evaluation (Prediction), Policy Improvement Theorem, Generalized Policy Iteration.

Self-Learning Component:

Monte-Carlo Reinforcement Learning: Introduction, Backup Diagrams, Types of MC Algorithms: First visit Monte Carlo, Every visit Monte Carlo; Monte Carlo Methods: MC Prediction, MC Control, Epsilon- Greedy

TEXTBOOKS:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019

REFERENCE BOOKS:

1. M. Wiering and M. van Otterlo, Eds., Reinforcement learning: State-of-the-art. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012
2. “Artificial Intelligence: A Modern Approach, 4th US ed,” Berkeley.edu. [Online]. Available: https://aima.cs.berkeley.edu/.

3. N. Bhavatarini, S. T. Ahmed, and S. M. Basha, Reinforcement learning - principles, concepts and applications. MileStone Research Publications, 2024.

JOURNALS/MAGAZINES:

1. Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves et al. "Human-level control through deep reinforcement learning." Nature 518, no. 7540 (2015): 529.

2. Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep reinforcement learning with double q-learning." In Thirtieth AAAI conference on artificial intelligence. 2016.

3. Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Freitas. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015).

SWAYAMNPTEL/MOOCs

1. https://nptel.ac.in/courses/106106143

2. https://www.coursera.org/specializations/reinforcement-learning

Course Title	Cognitive Computing				Course Type		SC	
Course Code	B24EHS512	Credits	3		Class		VI SEMESTER	
	LTP	Credits	ontact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:
This course explores the area of cognitive computing and its implications for today's world of big data analytics and evidence-based decision making. Topics covered include: cognitive computing design principles, natural language processing, knowledge representation, Students will have an opportunity to build cognitive applications, as well as explore how knowledge-based artificial intelligence and deep learning are impacting the field of data science.

COURSE OBJECTIVE (S):
The objectives of this course are to :

1. To develop algorithms that use AI and machine learning along with human interaction and feedback to help humans make choices/decisions and to understand how Cognitive computing supports human reasoning by evaluating data in context and presenting relevant findings along with the evidence that justifies the answers.
2. Identify how Artificial Intelligence, Machine Learning and Cognitive Computing are related and used for application development.
3. Identify, design and assess Cognitive Computing and Artificial Intelligence challenges/ issues with social impact.
4. Visualize the data characteristics, preprocessing and Understand the Patterns for further application development.

COURSE OUTCOMES (COs)
After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate a broad knowledge of concepts and issues in Cognitive Computing.	1 to 6	1,2
CO2	Explore the technical and managerial issues in developing and using applications based on cognitive computing and Artificial Intelligence Applications techniques.	1 to 6	1,2
CO3	Apply and assess Cognitive Computing and Artificial Intelligence Applications in an organizational setting	1 to 6	1,2
CO4	Understanding on different Perception and sensing data	1 to 6	1,2
CO5	Design a Real-Time information Based Evacuation Decision support system for emergency management	1 to 6	1,2
CO6	Understand Current Trends and Research Issues and applications of Cognitive computing	1 to 6	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√	√			
CO2			√	√		

CO3			√			
CO4		√	√			
CO5			√			
CO6		√				

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	3	3	2	3	1							3	3	
CO2	3	3	3	3	3	1							2	2	
CO3	3	3	3	2	2	1							2	3	
CO4	3	3	3	3	3	1							3	3	
CO5	3	3	3	2	3	1							3	3	
CO6	2	3	2	3	3	1							3	3	

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Cognitive Computing: Definition, concepts, and historical background, Cognitive Systems and Models: Understanding cognitive architectures and systems, Cognitive Processes: Perception, attention, memory, problem-solving, and decision-making, Cognitive Task Analysis: Methods for analyzing cognitive tasks and processes.

Brain Signals and Feature extraction, Types of Brain signals, case study, Feature extraction methods and their analysis, Models of Understanding Cognition or Mind: Neuroscientific Model, Psychological Model, Representational Model, Computational Model, Isomorphic Model, Multiple realizable Model, Multiple Draft Model, Sub personal Model.

Cognitive Computing in Virtual Reality (VR) and Augmented Reality (AR): Introduction to Virtual Reality (VR) and Augmented Reality (AR), Fundamentals and applications. Cognitive Aspects of VR/AR: Understanding the impact of VR/AR on perception and cognition. VR/AR Interaction Design: Designing intuitive and effective interfaces for VR/AR applications. Immersive Experience Design: Creating immersive experiences using cognitive computing principles.
Human-Computer Interaction (HCI) and User Experience (UX) Design: Introduction to Human-Computer Interaction: Basic principles and concepts. User-Centered Design: Understanding user needs, prototyping, and evaluation. Cognitive Aspects of HCI: Mental models, human memory, and attention in interface design. Multi-modal Interfaces: Designing interfaces for multiple input/output modes.

Self Learning Component:

1. Cognitive function measurement tools and software.

2. Selected topics from research papers.

TEXT BOOKS:

1. Friedenberg, Jay, Gordon Silverman, and Michael J. Spivey. Cognitive science: an introduction to the study of mind. Sage Publications, 2021, ISBN: 978-1544380155.
2. José Luis Bermúdez, Cognitive Science: An Introduction to the Science of the Mind (3 ed.), Cambridge University Press, 2020. ISBN 978-1108440349.
3. Rob High and Tanmay Bakshi, Cognitive Computing with IBM Watson: Build Smart Applications Using Artificial Intelligence as a Service (1 ed.), missing, 2019. ISBN 978-1788478298.

REFERENCE BOOKS:

1. LaValle, Steven M. Virtual reality. Cambridge university press, 2023..

1. LaValle, Steven M. Virtual reality. Cambridge university press, 2023..

2. Still, Brian, and Kate Crane. Fundamentals of user-centered design: A practical approach. CRC press, 2017.

JOURNALS/MAGAZINES:

1. Cognitive computing: https://www.springer.com/journal/12559/
2. International Journal of Cognitive Computing in Engineering, http://www.keaipublishing.com/en/journals/international-journal-of-cognitive-computing-in-engineering/
3. IEEE Transactions on Cognitive and Developmental Systems, https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=7274989

SWAYAMNPTEL/MOOCs

1. Swayam : Course Name : Fundamentals Of Artificial Intelligence, By Prof. Shyamanta M Hazarika, IIT Guwahati.
2. NPTEL: Course Name : Human Computer Interaction (HCI), unded by -MHRD, Govt. of India National Mission on Education Through Information and Communication Technology.
3. Coursera: Mind and Machine Specialization (https://www.coursera.org/specializations/mind-machine)

Course Title	Block chain Technology and Applications				Course Type		SC	
Course Code	B24EHS513	Credits	3		Class		V SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	**3**	**3**	**3**	**42**		**50**	**50**

COURSE OVERVIEW:

Blockchain is the distributed and decentralized database technology behind this crypto currency. This course explores the fundamentals of the public, transparent, secure, immutable, and distributed database called blockchain. Blockchains can be used to record and transfer any digital asset, not just currency. This course will introduce students to the workings and applications of this potentially disruptive technology. Its potential impact on financial services, government, banking, contracting and identity management.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the emerging abstract elements of Blockchain and Distributed Ledger Technology.
2. Describe the operational and functional aspects of cryptocurrency ecosystem.
3. Demonstrate the Smart Contract Basics interaction with users.
4. Discuss the various blockchain platforms for real world applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Describe the basic principles of blockchain technology and the essential components of a blockchain network, such as nodes, miners, consensus mechanisms, and smart contracts.	1 to 8	1,2
CO2	Illustrate the cryptographic techniques used in blockchains to secure data and transactions.	1 to 8	1,2
CO3	Describe the attributes of the blockchain technology used by Ethereum and Bitcoin.	1 to 8	1,2
CO4	Compare the performance of various mining Consensus algorithm.	1 to 8	1,2
CO5	Understand the use of Hyperledger to build various application on blockchain.	1 to 8	1,2,3
CO6	Identify and apply the features of Block chain technology in real world application across various industries.	1 to 8	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	**Understand(L2)**	**pply (L3)**	**nalyze (L4)**	**aluate (L5)**	**reate (L6)**
CO1		**√**				
CO2			**√**			
CO3			**√**			

CO4			**√**			
CO5		**√**	**√**			
CO6			**√**	**√**		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	1	3	1	2	1	2					3	3	
CO2	3	3	1	3	1	2	1	2					3	3	
CO3	3	3	2	3	3	2	1	2					3	3	
CO4	3	3	2	3	3	2	1	2					3	3	
CO5	3	3	3	3	1	2	1	2					3	3	2
CO6	3	3	2	3	1	2	1	2					3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Blockchain: The History of blockchain and Bitcoin, Types of Blockchain, Consensus, CAP theorem and Blockchain, Decentralization using Bitcoin, Methods of decentralization, Blockchain and full ecosystem decentralization, Smart Contracts, Decentralized Organization, Platform for decentralization, Attacks on Blockchains.

Blockchain Cryptography: Privacy and Security on Block chain, Hash Functions, Digital Signature Algorithm. **Introduction to Bitcoin**: Bitcoin definition, Digital Keys and addresses, Transactions, Blockchain structure, Mining, Proof-of-Work(POW), Bitcoin Network, Wallets.

Introduction to Ethereum: Ethereum blockchain, The Ethereum Network, components of the Ethereum Ecosystem, Keys and addresses, Accounts, Transaction and message, Ether Cryptocurrency, The Ethereum Virtual Machine, Smart Contracts, Proof-of-Stake (POS).

Introduction to Hyperledger: Hyperledger as a protocol, the reference Architecture, Requirements and design goals of Hyperledger Fabric, Hyperledger Fabric, Blockchain services, Consensus service: Byzantine Fault Tolerance (BFT), distributed ledger, Chain code. **Use cases**: Blockchain in Financial Software and system, Blockchain for Government and Blockchain in trade supply chain.

Self Learning Component:

Development Tools and Frameworks.

Web3 for Block chain Technology

TEXT BOOKS:

1. Imran Bashir, “Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained”, Packt Publishing. 2018

2. Lorne Lantz& Daniel Cawrey ,”Mastering Blockchain,Unlocking the power of Cryptocurrencies,Smart contract and Decentralixed Application”,O’reilly Publication,2021.

3. Treccani,A.,Lipton ,A, Blockchain and Distributed Ledgers:Mathematics,Technology, and Economics-First Edition,Singapore:World Scienfific Publishing Company,2021.

REFERENCE BOOKS:

1. Aravind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder," Bitcoin and Cryptocurrency Technologies-a Comprehensive Introduction", Princeton University Press,2016.
2. Antony Lewis, "The Basics of Bitcoins and Blockchains-An Introduction to Cryptocurrencies and the Technology that powers them", Mango Publishing, 2018.
3. Andreas M. Antonopoulos, Dr. Gavin Wood, "Mastering Ethereum -Building Smart Contracts and dApps", Ethereum Book LLC,2019.
4. Narayanan, Bonneau, Felten, Miller and Goldfeder, "Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction", Princeton University Press.
5. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.

JOURNALS/MAGAZINES:

1. https://soliditylang.org/
2. https://hyperledger-fabric.readthedocs.io/en/release-1.4/prereqs.html
3. https://platon-truffle.readthedocs.io/en/v0.1.0/
4. https://trufflesuite.com/docs/
5. https://docs.metamask.io/
6. https://remix-ide.readthedocs.io/en/latest/
7. https://geth.ethereum.org/docs
8. https://readthedocs.org/projects/web3js/
9. https://web3js.readthedocs.io/en/v1.10.0/
10. https://holbrook.no/share/doc/ethereum/web3js/1.0.0/
11. https://docs.web3js.org/
12. https://framework.embarklabs.io/docs/overview.html

SWAYAMNPTEL/MOOCs

1. https://onlinecourses.nptel.ac.in/

Professional Elective 4

Course Title	**Genomics**				**Course Type**		**SC**	
Course Code	**B24EHS521**	**Credits**	**3**		**Class**		**V SEMESTER**	
	TLP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Theory	3	3	3				
	Practice	0	0	0	Theory	Practical	CIE	SEE
	-	-	-	-				
	Total	**3**	**3**	**3**	**42**	**0**	**50**	**50**

COURSE OVERVIEW:

This course introduces the basics of Cell Biology and Molecular Biology with Genomics, and the experimental tools and databases. In this course, students gain extensive knowledge and experience on Molecular Genetics and Genome Mapping. An introduction to the key concepts Genomics, Biological Databases, and Cancer Genetics are covered.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Understand the cell theory and the mechanisms of the central dogma of life.
2. Introduction to the basics of sequence analysis and molecular genetics.
3. Discuss the methods to characterise and manage the different types of biological data.
4. Introduce the types of cancer and molecular basis of cancer biology.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Outline the concepts of Cell Biology and Gene functions.	1,2,6,8,12	1,2,3
CO2	Acquire the basic concept of genomics and functional genomics with various methods for genome sequencing and assembly.	1 to 5,6,8,12	1,2,3
CO3	Analyse the molecular mechanisms of gene regulation in prokaryotes and eukaryotes with the basis of their genomes.	1 to 5,6,8,12	1,2,3
CO4	Understand the molecular basics of life with cell cycle and cell division.	1,2,6,8,12	1,3
CO5	Apply theoretical knowledge and practical skills for the analysis of biological communities, cell population, gene-bank data mining, annotation, and management	1 to 5,6,8,12	1,3
CO6	Identify, locate and retrieve specific records from various biological databases	1 to 5,6,8,12	1,2,3

BLOOM'S LEVELOF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3			√	√		
CO4		√				
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

O# / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	2			1	1		2				2	3	3	3
CO 2	3	3	2	1	2	1		2				2	3		3
CO 3	3	3	2	1	2	1		2				2	3	3	3
CO 4	3	2				1		2				2	3		3
CO 5	3	3	3	2	2	1		2				2	3		3
CO 6	3	3	3	2	2	1		2				2	3	2	3

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Introduction to Cell Biology and Gene Function: Molecular basis for life, Central dogma, Nucleic Acids, Gene, Genetic Code, Genomics, Human Genome Project, Proteins, Properties of Protein, Structure of Protein, Proteomics, Nomenclature of Protein Sequences, Functions of the genetic material, Evidence for DNA and RNA as the genetic material, The transport of molecules between the nucleus and the cytosol, The molecular mechanisms of endocytosis, and exocytosis.

Gene function at Molecular level- Replication, Transcription & Translation

Molecular Biology and Molecular Genetics: Molecular basis for life, An over view on biological organization (eg. human); Structural organization of prokaryotic and eukaryotic cells, Ultrastructure and functions of nucleus, mitochondria, plastids, endoplasmic reticulum, Golgi complex, lysosomes, microbodies, ribosomes. Cytoskeleton, Plant and animal cells, Variation in structure and function, Cell cycle and its importance, Phases of cell cycle, Genetic basis of syndromes and disorders.

Genome and Sequence Analysis with Gene Mapping: Introduction, Genome Analysis, Genome Annotation, Genome Mapping, The Sequence Assembly problem, Sequence comparison: Sequence alignment, Pair wise sequence alignment, Multiple sequence alignment and their importance, Genetic Mapping and Linkage Analysis, Physical Maps, Cloning the entire Genome, Genome Sequencing, Applications of Genetic Maps, Sequence Assembly Tools, Identification of Genes in Contigs, Basis of Gene Prediction, Gene Prediction Methods, Molecular evolution and Phylogenetic analysis

Biological Databases & Cancer Genetics: Motivation of biological database, Major Databases in Bioinformatics, Electronic Libraries, Nucleotide sequence databases - European Molecular Biology Laboratory (EMBL) - NCBI GenBank – DNA Data Bank of Japan (DDBJ), Protein amino acid sequence databases- UNIPROT, Protein Data Bank (PDB) Pfam-protein family database - GO-gene ontology, Swiss-Prot, Protein Sequence Motif Databases, Tools for Web Search, Data Retrieval Tools, Data Mining of Biological Databases.

Cancer Genetics: Definition, Types, Relationship of the cell cycle to cancer, Cancer and programmed cell death, Genetic basis for cancer, oncogenes, Tumour suppressor genes, Role of environmental factors in cancer and genetic pathways to cancer.

SELF-LEARNING COMPONENT:

1. Protein structure prediction
2. Homology and Similarity Tools
3. Biostatistics

TEXT BOOKS:

1. S C Rastogi, N Mendiratta and P Rastogi, " Bioinformatics: Methods and Applications" , ISBN : 978-81-203-4785-4, PHI Learning Private Limited, 2015.
2. David W Mount," Bioinformatics- Sequence and Genome Analysis", second Edition, CBS Publishers, 2014.
3. The Cell: A Molecular Approach, Geoffrey M. Cooper and Robert E Hausman, Oxford University press, 2015, 7th Edition
4. Genomes 4, T. A. Brown, Garland Science, 2017, 4th Edition
5. Andreas D. Baxevanis, B. F. Francis Ouellette," Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins", 2nd Edition, Wiley Inderscience, 2001.

REFERENCE BOOKS:

1. Mount D (2014) Bioinformatics: Sequence and Genome Analysis, Cold Spring Harbor
2. Lewin's Genes XII (2017) - Jocelyn E. Krebs, Elliott S. Goldstein, Stephen T. Kilpatrick. Jones and Bartlett Publishers
3. iGenetics – A Molecular approach, 3rd edition (2016) – Peter J Russell, Pearson Benjamin Cummings, San francisco, USA
4. Attwood, T.K., Parry, D.J., Smith, Introduction to Bioinformatics, Pearson Education, 2005.
5. The Analysis of Gene Expression Data: An Overview of Methods and Software, Giovanni Parmigiani, Elizabeth S. Garrett, Rafael A. Irizarry, Scott L.Zeger

JOURNALS/MAGAZINES:

1. Briefings in Functional Genomics
2. BMC Bioinformatics
3. Proteins: Structure, Function and Bioinformatics
4. IEEE/ACM Transactions on Computational Biology and Bioinformatics
5. International Journal of Genomics
6. Bentham Science- Current Bioinformatics
7. Genomic Medicine

SWAYAM/NPTEL/MOOCs:

1. https://onlinecourses.nptel.ac.in/noc19_bt24/preview - Functional Genomics
2. https://onlinecourses.swayam2.ac.in/cec20_bt03/preview- Genetics and Genomics
3. https://www.coursera.org/specializations/genomic-data-science

Course Title	High Performance and Quantum Computing				Course Type		SC	
Course Code	B24EHS522	Credits	3		Class		V SEMESTER	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number ofClasses Per Semester		Assessment in Weightage	
	Theory	3	3	3				
	Practice	0	0	0	Theory	Practical	CIE	SEE
	-	-	-	-				
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

This course provides foundations for developing parallel algorithms. It focuses on types of parallelism and Flynn's classification of Computer Architectures. It provides a practical feeling of how algorithms map to and behave on real systems and supplement algorithmic theory with hands-on exercises on modern HPC systems, such as CUDA for graphics co-processors (Graphics Processing Unit).

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Explain the basic concepts related to HPC architecture and parallel computing.
2. Illustrate the use of CUDA concepts in real world applications.
3. Describe the concepts of NVIDIA SLI.
4. Discuss different Parallel algorithms and their applications in real world examples.

COURSE OUTCOMES(COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Outline the basic concepts related to HPC architecture and parallel computing.	1 to 4, 8, 9 , 12	1
CO2	Illustrate the concepts of CUDA for scientific computations.	1 to 3, 5,9,12	1
CO3	Make use of concepts of NVIDIA SLI in real world applications.	1 to 5, 9, 12	2
CO4	Analyze performance of different parallel algorithms.	1,4,5,9,12	1
CO5	Develop and test parallel computing solutions to real world programmingproblems.	1,5,9,12	1
CO6	Demonstrate the architectural features in the GPU hardware accelerators.	1,5,8,9,12	1

BLOOM'S LEVELOF THECOURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2		√				
CO3			√			
CO4				√		
CO5			√			
CO6		√				

COURSE ARTICULATION MATRIX

CO #/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	1	1	2				1	1			1	3		
CO 2	3	2	3		2				1			1	3		
CO 3	3	1	2	1	2				1			1		3	
CO 4	3			2	2				1			1	3		
CO 5	3				2				1			1	3		
CO 6	3				2			1	1			1	3		

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Parallel Processing Concepts: Levels of parallelism –Bit, instruction; Data, transaction; task; thread;memory; function; Flynn's Classification of Computer Architectures: SISD, SIMD, MISD, MIMD. Dataflow Models, Demand- driven Computation, Introduction to superscalar architectures; multi-core; multi-threaded.

Parallel Programming with CUDA: Processor Architecture; Interconnect; Communication; Memory Organization; and Programming Models in high performance computing architecture Memory hierarchy and transaction specific memory design; Thread Organization. Heterogeneous Computing; Blocks; Threads; Indexing Shared memory sync threads (); Asynchronous operation; Handling errors; Managing devices.

NVIDIA SLI: What is SLI? , Choosing SLI Modes, Avoid CPU Bottlenecks, Disable VSync by Default, DirectX SLI Performance Tips, OpenGL SLI Performance Tips. **Multi-GPU Performance:** Moving to Multiple GPUs, Subdividing Computation across Multiple GPUs, Peer-to-Peer Communication on Multiple GPUs, Finite Difference on Multi-GPU, Scaling Applications across GPU Clusters.

QUANTUM COMPUTING: Qubits, quantum entanglement; reversible computation, quantum gates; quantum parallelism and simple quantum algorithms; quantum Fourier transforms and its applications, quantum search algorithms; introduction to quantum error correcting codes; entanglement assisted communication; elements of quantum information theory and quantum cryptography. Simulation of molecular catalysis; biomolecular simulation using hybrid QM/MM approach.

TEXTBOOKS:

1. Nielsen, Frank, "Introduction to HPC with MPI for Data Science" , Springer, 2nd Edition, 2016
2. Ananth Grama, Anshul Gupta, George Karypis, and VipinKumar, "Introduction to Parallel Computing", 2nd edition, Addison-Welsey, 2003
3. Grama, A. Gupta, G. Karypis, V. Kumar, "An Introduction to Parallel Computing, Design and Analysis of Algorithms" Addison-Wesley, 2nd Edition, 2003.
4. NVIDIA GPU Programming Guide Version 2.5.0, by NVIDIA Corporation, 2006.
5. John Cheng, Max Grossman ,Ty McKercher, "PROFESSIONAL CUDA C Programming" , John Wiley & Sons, Inc,.2014
6. Kai Hwang , "Advanced Computer Architecture: Parallelism, Scalability, Programmability", McGraw Hill 1993.
7. Michael A. Nielsen and Issac L Chuang "Quantum Computation and Quantum Information Cambridge University Press, 10th Edition 2016.

REFERENCE BOOKS:

1. J. Dongarra, I. Foster, G. Fox, W. Gropp, K. Kennedy, L. Torczon, A. White, "The Sourcebook of Parallel Computing", Morgan Kaufmann, 1st Edition, 2002.
2. G.S. Almasi and A. Gottlieb, "Highly Parallel Computing", 2ndEdition, Addison-Wesley, 1994.
3. Wilkinson and M. Allen, "Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers", Prentice Hall,2nd Edition, 2005.
4. M.J. Quinn, "Parallel Programming in C with MPI and OpenMP", McGraw-Hill, 2004.
5. G.E. Karniadakis, R.M. Kirby II, "Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and their Implementation", Cambridge University Press, 2003

JOURNALS/MAGAZINES:

1. Elsevier Journal on high performance computing for Big data.
2. Elsevier Journal on Parallel Computing

SWAYAM/NPTEL/MOOCs:

1. https://nptel.ac.in/courses/106/102/106102114/
2. https://onlinecourses.nptel.ac.in/noc19_cs45/preview

Course Title	Robotic Process Automation				Course Type		SC	
Course Code	B24EHS523	Credits	3		Class		V Semester	
Course Structure	TLP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		ssessment in Weightage	
	Theory	3	3	3				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	-	-	-	-				
	Total	**3**	**3**	**3**	**42**	**-**	**50**	**50**

COURSE OVERVIEW*:*

Robotic Process Automation (RPA) offers many challenges for software developers and scientists. This course introduces the UiPath Robotic Process Automation concepts through UiPath Studio and UiPath Orchestrator where a student gains knowledge of how to build a bot to automate required tasks.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Discuss the concepts of Robotics Process automation
2. Describe the sequence, flowchart, and control flow in automation tool
3. Demonstrate the data manipulation techniques
4. Demonstrate the usage of UI Explorer and Screen scraping

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Make use of recording features in UiPath Studio to automate the repetitivetasks.	1 to 5	1,2,3
CO2	Apply appropriate Workflow Activities in UiPath Studio to automate thecomplex tasks using Flowchart and Sequence.	1 to 5	1,2,3
CO3	Build data table and data manipulation techniques in UiPath Studio to automate CSV / Excel workbook applications.	1 to 5	1,2,3
CO4	Design and Develop bot process using UI Explorer.	1 to 5	1,2,3
CO5	Design and Develop process to automate using Screen Scraping for complexapplications.	1 to 5	1,2,3
CO6	Construct any real-world application by making use of the Automation features in RPA	1 to 5	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

CO #/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO 1	3	3	3	2	3							3	3	2
CO 2	3	3	3	2	3							3	3	2
CO 3	3	3	3	2	3							3	3	2
CO 4	3	3	3	2	3							3	3	2
CO 5	3	3	3	2	3							3	3	2
CO 6	3	3	3	2	3							3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT:

What Is Robotic Process Automation: Scope and techniques of automation, Robotic process automation, About UiPath, Future of Automation. Record and Play: UiPath stack, Downloading and installing UiPath Studio, Learning UiPath Studio, Task recorder, Step-by-step examples using the recorder.

Sequence. Flowchart and Control Flow: Sequencing the Workflow, Activities, Control Flow, various types of loops, and decision making, Step-by-step example using Sequence and Flowchart, Step-by-step example using Sequence and Control Flow.

Data Manipulation ,Exception handling and Debugging: Variables and Scope, Arguments-Purpose and use, Collections and tables- Text manipulation-Data Manipulation-Gathering and Assembling data, Data table usage and examples, Clipboard management, File operation with step-by-step example, CSV/Excel to data table and viceversa with a step- by-step example,common exceptions and ways to handle them, Logging and taking screenshots, Debugging techniques.

Taking Control of the Controls: Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls-mouse and keyboard activities, Working with UI Explorer, Handling events, Revisit recorder, Screen Scraping, When to use OCR, Types of OCR available, How to use OCR, Avoid typical failure points.

Tame that Application with Plugins: Mail plugin, PDF plugin, web integration, Excel and Word plugins, Credential management.

TEXT BOOKS:

1. Alokmani Tripathi, “Learning Robotic Process Automation”, Packt Publishing, 2018.
2. E. Turban, R. Sharda, D. Delen, David King, “Business Intelligence”, Pearson India, 2010.

REFERENCE BOOKS:

1. Marlon Dumas et. al., “Fundamentals of Business Process Management”, Springer, ebook, 2012.
2. Van der Aalst, “Process Mining: Discovery, Conformance and Enhancement of Business Processes”, Third edition, 2011.

JOURNALS/MAGAZINES

1. https://rpa-journal.org/
2. https://www.sciencedirect.com/science/article/pii/S1877050921001393
3. https://ieeexplore.ieee.org/document/9001110./authors#authors

SWAYAM/NPTEL/MOOCs:

1. https://www.uipath.com/rpa/academy
2. https://www.coursera.org/specializations/roboticprocessautomation
3. https://www.udemy.com/topic/robotic-process-automation/

SELF-LEARNING EXERCISES:

1. Handling User Events

2. Assistant Bots

Open Elective 1

Course Title	Data Science with Python				Course Type		OE	
Course Code	B24EFO511	Credits	3		Class		V Semester	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course introduces Python Programming fundamental components. Students are introduced to data structures, strings and file handling. This course includes an overview of Exception handling and Object-Oriented Programming Principles. In the course we also discuss useful python libraries such as numpy, pandas and matplotlib.

COURSE OBJECTIVE (S):

The objectives of this courses are to

1. Learn the syntax and semantics of the Python programming language.
2. Illustrate the process of structuring the data using collections.
3. Demonstrate the use of built-in functions to navigate the file system.
4. Demonstrate the use of string in-built functions to work with text data.
5. Illustrate exception handling and Object-Oriented Programming concepts in Python.
6. Outline the usage of python libraries for working with arrays, data sets and representing data in the form of graphs/plots.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Illustrate basic python syntax and semantics.	1,2,3,4,5,12	1, 2, 3
CO2	Identify and use different data types and data structures in Python, such as lists, tuples, dictionaries, and sets.	1,2,3,4,5,12	1, 2, 3
CO3	Develop programs to handle a wide variety of tasks in Python programming that involve text and string data.	1,2,3,4,5,12	1, 2, 3
CO4	Demonstrate reading from and writing to files using Python.	1,2,3,4,5,12	1, 2, 3
CO5	Illustrate exception handling and outline the principles of object-oriented programming.	1,2,3,4,5,12	1, 2, 3
CO6	Analyze retrieved data with appropriate Python visualization libraries.	1,2,3,4,5,12	1, 2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3			√			
CO4		√				
CO5		√				
CO6				√		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	2							3	3	3	3
CO2	3	3	3	3	2							3	3	3	3
CO3	3	3	3	3	2							3	3	3	3
CO4	3	2	2	3	2							3	3	3	3
CO5	3	3	3	3	2							3	3	3	3
CO6	3	3	3	3	3							3	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT:

Fundamentals of Python Programming: Introduction to Python Features, Constants, Variables, Naming conventions, simple data types, indentation, operators and expressions, conditional statements, loops, functions, user defined functions, Working with Data Structures- Lists, tuples, sets and dictionaries.

Strings and File handling: Strings - definition, about Unicode, chr( ) and ord() function, creating strings, reading strings, slicing, concatenation and comparison of strings, string library functions. File Handling - definition, file access modes, reading text files and binary files, writing text files and binary files, file object attributes, file handling functions.

Exception Handling and Object-Oriented Programming concepts: Exception Handling - Definition of error and exception, difference between syntax error and exceptions, Try and Except Statement – Catching Exceptions, Catching Specific Exception, Try with Else Clause, finally keyword, Raising exception, NameError, TypeError, ZerDivisionError. Object Oriented Programming concepts- Definition of a class and object, class and object creation, self-parameter, constructors, modifying object properties, deleting object properties, OOP features
- Encapsulation, Data abstraction, Inheritance, and polymorphism.

Python Libraries: NumPy - Basics of numpy, ndarray, dimensions in arrays, ndim attribute, ndmin attribute, accessing array elements, negative indexing, array slicing, array copy vs view, array shape, array reshaping, arange() function, matrix operations, universal array functions. Pandas - Basics of pandas, why use pandas, what can pandas do, Creation of DataFrame and Series, named indices, loc and iloc attribute, Load files into DataFrame. Matplotlib - Basics of matplotlib, features and advantages of matplotlib, Graphs/Plots.

Self-Learning Component: Python SQL Database Access, seaborn, keras.

TEXT BOOKS:

1. Eric Mathes, "Python Crash Course, A Hands-on Project-Based Introduction to Programming", 2nd Edition, 2019.
2. Bill Lubanovic, "Introducing Python Modern Computing in Simple Packages", O'Reilly, 2nd Edition, 2020.
3. Michael Dawson, "Python Programming for the absolute Beginner", Cengage, 3rd Edition, 2020.

REFERENCE BOOKS:

1. Charles Dierbach, "introduction to Computer Science using Python", Wiley India Edition, 2015.
2. Allen B. Downey, "Think Python, How to think like a Computer Scientist", O'Reilly, 2nd Edition, 2017.

3. Mark Lutz, “Learning Python”, Oreilly. 2003.

4. Mark Pilgrim, “Dive into Python 3”, Apress special edition, second edition, 20l5.

5. T.R. Padmanabhan, Programming with Python, Springer, 1st Ed., 2016.

6. Kenneth Lambert, Fundamentals of Python: First Programs, Cengage Learning,, 1st Ed., 2012.

7. Michael Dawson, “Python Programming for the Absolute Beginners”, 3rd Edition, CENAGE Learning.

8. Paul Barry, Head First Python,by O'Reilly Media, Inc 2nd Edition Released November 2016.

9. Wesley J. Chun, “Core Python Programming”, 2nd Edition, Prentice Hall.

JOURNALS/MAGAZINES:

1. IEEE Transactions on Artificial Intelligence
2. Journal of Machine Learning Research
3. Foundations and Trends in Machine Learning
4. Synthesis Lectures on Artificial Intelligence and Machine Learning
5. ACM Transactions on Intelligent Systems and Technology

SWAYAM NPTEL/MOOCs:

1. Python for Data Science, Prof. Ragunathan Rengasamy, IIT Madras.
2. The Joy of Computing using Python, Prof. Sudarshan Iyengar, IIT Ropar.

Course Title	Database Management Systems				Course Type		OE	
Course Code	B24EFO512	Credits	3		Class		V Semester	
Course Structure	TLP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Theory	3	3	3				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	-	-	-	-				
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course introduces topics such as conceptual data modelling, relational data model, relational query languages, and relational database design. It helps the students to gain fundamental concepts, techniques and applications in database.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the basics of Database Management System.
2. Demonstrate the use of Relational model and Relational algebra.
3. Illustrate the use of different SQL statements in query processing.
4. Discuss the importance of optimal Database Design with Normalization.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Summarize and exemplify fundamental nature and characteristics of database systems.	1 to 5,9 10,12	1,2
CO2	Model and design solutions for efficiently representing and querying data using relational model.	1 to 5,9.10,12	1,2
CO3	Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.	1 to 5,9,10,12	1,2,3
CO4	Construct the database for given real world application and solve queries over it using SQL commands.	1to 5,9,10,12	1,2
CO5	Develop an optimized database using design guidelines and normalization technique.	1to 5,9,10,12	1,3
CO6	Construct the physical and logical database designs, database modeling, relational, hierarchical, and network models.	1to 5,9,10,12	2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6		√				

COURSE ARTICULATIONMATRIX:

O#	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2				3	3		1	3	3	
CO2	3	3	2	3	1				3	3		1	3	3	
CO3	3	3	2	3	3				3	3		1	3	3	3
CO4	3	3	2	3	1				3	3		1	3	3	
CO5	3	2	3	3	3				3	3		1	3		3
CO6	3	3	2	3	3				3	3		1		3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to databases and Conceptual Modelling : Introduction to database, characteristics of the database approach, data models, schemas, instances, database languages and interfaces, Using high-level conceptual data models for database design, a sample database application, entity types, attributes, keys, relationship types, weak entity types, ER diagrams, naming conventions, design issues. Introduction to various database tools and framework (commercial and open source)

Relational Data Model and Relational algebra: Relational model concepts, relational model constraints and relational database schemas, update operations, transactions, dealing with constraint violations, unary relational operations, select and project, relational algebra operations from set theory, binary relational operations, join and division, additional relational operations, examples of queries in relational algebra.

SQL:SQL data definition and data types, specifying constraints in SQL, basic retrieval queries in SQL, insert, delete, update statements in SQL, additional features of SQL, schema change statements in SQL, Retrieving data using the SQL Select Statement, Restricting and sorting data, Using Single row functions, Joins, More complex SQL retrieval queries, views in SQL.

Database Design Theory and Normalization: Informal design guidelines for relation schemas, Functional dependencies, and Normal forms based on primary keys, General definitions of second and third normal forms, Other Normal forms.

Self Learning Component: Transaction, Basic terminologies, ACID properties

TEXTBOOKS:

1. Elmasri and Navathe, " Fundamentals of Database Systems", Pearson Education, 5th Edition, 2007.
2. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", McGraw-Hill, 3rd Edition, 2003.
3. Phill Pratt, "Concepts of Database Management, Cengage Learning", 8th Edition, 2014
4. Jeffrey A Hoffer, "Modern Database Management, Pearson", 12th Edition, 2015

REFERENCEBOOKS:

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan: "Database System Concepts", 6th Edition, McGraw Hill, 2010.
2. C J Date, "Database Design and Relational Theory: Normal Forms and All that Jazz", O 'Reilly, April 2012.
3. James Martin, "Principles of Database Management Systems", 1985, Prentice Hall of India, New Delhi
4. IEEE Transactions on Knowledge and Data Engineering
5. Elsevier Data and Knowledge Engineering
6. ACM Transactions on Database Systems

JOURNALS/MAGAZINES:

2. https://www.dbjournal.ro/

SWAYAM/NPTEL/MOOCs:

1. https://www.coursera.org/courses?query=database%20management
2. https://onlinecourses.swayam2.ac.in/cec19_cs05/preview
3. https://www.edx.org/learn/databases
4. https://www.classcentral.com/course/swayam-data-base-management-system-9914

Course Title	Feature Engineering with Python Lab				Course Type		HC	
Course Code	B24EH0504	Credit	1		Class		V Semester	
Course Structure	LTP	Credit	Contact	Workload	Total Number of Classes		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2	0	28	25	25

COURSE OVERVIEW:

Hands-on course designed to provide students with practical experience in transforming raw data into powerful features for machine learning models. This lab complements theoretical knowledge with real-world applications, guiding students through various techniques and tools essential for effective feature engineering. Using Python and popular libraries such as pandas, NumPy, and scikit-learn, students will learn to manipulate data, handle missing values, preprocess text and images, and apply advanced statistical methods.

COURSE OBJECTIVE (S):

The objectives of this courses are to be:

1. Understand and apply statistical methods to describe and analyze data distributions.
2. Apply PCA for dimensionality reduction and data visualization.
3. Implement encoding techniques for categorical variables in machine learning.
4. Address missing data using various imputation techniques and Preprocess text data for feature extraction.
5. Extract features from images using convolutional neural networks (CNNs).
6. Build a recommendation system using feature engineering techniques.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Compute and interpret moments, sample distributions, and Gaussian family parameters for a given dataset.	3	1
CO2	Create visualizations to illustrate the transformation and variance explained by the principal components.	4	2
CO3	Analyze the impact of different encoding methods on the performance of a machine learning model.	5	2
C04	Create a robust imputation strategy for datasets with complex missing data patterns.	8	2
CO5	Create a feature extraction model for images and apply it to a new dataset.	10	3
CO6	Evaluate the effectiveness of the recommendation system using metrics such as precision, recall, and F1-score.	11	3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply(L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1		√				
CO2		√				
CO3			√			

C04				**√**		
CO5						**√**
CO6					**√**	

COURSE ARTICULATION MATRIX

CO# /	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	3	3										3	2	1
CO2		2	2	2	2								2	2	2
CO3			3	3	3								2	2	2
CO4							2	2					3	3	3
CO5									3	3			2	2	2
CO6											2	2	1	2	3

Practice:

Sl.no	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Statistical Description and Distribution Analysis	Python, pandas, NumPy, matplotlib, seaborn	Compute and interpret statistical descriptors, understand and visualize data distributions, perform non-parametric descriptions.
2	Principal Component Analysis (PCA)	Python, scikit-learn, pandas, matplotlib	Apply PCA for dimensionality reduction, interpret principal components, visualize variance explained by components.
3	Data Manipulation with Pandas	Python, pandas	Use pandas for data manipulation tasks: reindexing, selection, filtering, sorting, handling missing data.
4	Hypothesis Testing and ANOVA	Python, SciPy, statsmodels, pandas	Perform hypothesis testing, conduct ANOVA, interpret statistical results.
5	Encoding Categorical Variables	Python, pandas, scikit-learn	Implement and evaluate encoding techniques (one-hot, label, target encoding) and their impact on machine learning models.
6	Feature Scaling and Normalization	Python, scikit-learn, pandas	Apply feature scaling techniques (standardization, min-max scaling) and understand their importance in model performance.
7	Handling Missing Data	Python, pandas, scikit-learn	Identify missing data patterns, apply imputation techniques, evaluate their effectiveness.
8	Text Data Preprocessing	Python, NLTK, spaCy, pandas	Preprocess text data (tokenization, stemming, lemmatization), convert text into features for machine learning models.
9	Image Feature Extraction using CNNs	Python, TensorFlow or PyTorch, OpenCV	Implement CNNs to extract features from images, understand CNN layer structures and functionality.
10	Building a Recommendation System	Python, pandas, scikit-learn,	Build a recommendation system using collaborative and content-based filtering,

		surprise	evaluate performance using appropriate metrics.

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Statistical Analysis of Customer Demographics	Python, pandas, NumPy, matplotlib, seaborn	Perform statistical analysis, interpret descriptive statistics, visualize demographic data distributions.
2	Dimensionality Reduction in Image Data	Python, scikit-learn, pandas, matplotlib	Apply PCA to reduce dimensions in image data, interpret principal components, create visualizations of reduced dimensions.
3	Data Cleaning and Transformation for Sales Data	Python, pandas	Clean and transform sales data using pandas, handle missing values, perform data selection, filtering, and sorting.
4	Hypothesis Testing on A/B Test Results	Python, SciPy, statsmodels, pandas	Conduct hypothesis tests to evaluate A/B test results, perform ANOVA, interpret the significance of test outcomes.
5	Encoding Categorical Data in E-commerce Dataset	Python, pandas, scikit-learn	Implement one-hot and label encoding on e-commerce data, evaluate the impact on model performance.
6	Impact of Feature Scaling on House Price Prediction	Python, scikit-learn, pandas	Apply scaling techniques to features, evaluate their effect on regression models predicting house prices.
7	Imputation Techniques for Health Survey Data	Python, pandas, scikit-learn	Identify and impute missing data in health surveys, compare imputation methods, evaluate their effectiveness.
8	Sentiment Analysis of Product Reviews	Python, NLTK, spaCy, pandas	Preprocess text data from product reviews, extract features, apply sentiment analysis, visualize results.
9	Image Classification using Convolutional Neural Networks	Python, TensorFlow or PyTorch, OpenCV	Build and train a CNN for image classification, extract and visualize feature maps, evaluate model performance.
10	Movie Recommendation System	Python, pandas, scikit-learn, surprise	Develop a recommendation system using collaborative and content-based filtering, evaluate system performance.

TEXT BOOKS:

1. Klemens, Ben. *Modeling with data: tools and techniques for scientific computing*. Princeton University Press, 2008.
2. McKinney, Wes. *Python for data analysis*. " O'Reilly Media, Inc.", 2022.
3. Zheng, Alice, and Amanda Casari. *Feature engineering for machine learning: principles and techniques for data scientists*. " O'Reilly Media, Inc.", 2018.

REFERENCE BOOKS:

1. Duboue, Pablo. *The art of feature engineering: essentials for machine learning*. Cambridge University Press, 2020.

JOURNALS/MAGAZINES:

1. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)
2. Journal of Data Science
3. Data Mining and Knowledge Discovery
4. Pattern Recognition Letters
5. International Journal of Computer Vision (IJCV)

Course Title	Image Processing Lab				Course Type		HC	
Course Code	B24EH050	Credits	1		Class		V Semester	
Course Structure	LTP	Credits	Contact	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-			CIE	SEE
	Practice	1	2	2	Theory	Practical		
	Total	2	2	2	0	28	50	50

COURSE OVERVIEW:

To understand and implement image processing techniques using open source software

COURSE OBJECTIVE (S):

The objectives of this course are to

1.Understand image acquisition and storage using a open source software

2 Analyze different image transforms on images

3 Analyze and apply different techniques and algorithms for image enhancement

4 Analyze and apply different techniques and algorithms for image restoration

5 Analyze and apply different techniques and algorithms for image compression

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand the acquisition and storage of different types of images	1,2,5, 12	
CO2	Understand Analyze and Apply different image transforms and their properties	1-5, 12	
CO3	Apply different Image Smoothening & Sharpening algorithms in time and frequency domain	1-5, 12	
C04	Apply different algorithms for image restoration	1-5, 12	
CO5	Apply different techniques for image segmentation	1-5, 12	
CO6	Apply different image compression techniques	1-5, 12	

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply(L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1		√				
CO2			√			
CO3			√			
C04			√			
CO5			√			
CO6			√			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	2	-	-	-	-	-	-	2	-	-	-
CO2	3	2	2	2	2	-	-	-	-	-	-	2	-	-	-
CO3	3	2	2	2	2	-	-	-	-	-	-	2	-	-	-
CO4	3	2	2	2	2		-	-	-	-	-	2	-	-	-
CO5	3	2	2	2	2	-	-	-	-	-	-	2	-	-	-
CO6	3	2	2	2	2	-	-	-	-	-	-	2	-	-	-
S- Strong; M-Medium; L-Low															

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	a. Display the color channels and grey level images b. implementation of spatial neighbours of pixels (N=4,8, Daigonal)	OpenCV, NumPy, Matplotlib, Scikit-learn	Programming skills in Python
2	Implementation of Transformations of an Image	OpenCV, NumPy, Pillow	Programming skills in Python
3	Contrast stretching of a low contrast image, Histogram, and Histogram Equalization	OpenCV, NumPy, Matplotlib, Scikit-learn	Programming skills in Python
4	Display of bit planes of an Image	OpenCV, NumPy, Matplotlib	Programming skills in Python
5	Display of DFT & DWT (1-D & 2-D) of an image	OpenCV, NumPy, Matplotlib, PyWavelets	Programming skills in Python
6	Implementation of Image Smoothening Filters(Mean and Median filtering of an Image)	OpenCV, NumPy, Matplotlib	Programming skills in Python
7	Implementation of image sharpening filters and Edge Detection using Gradient Filters	OpenCV, NumPy, Matplotlib	Programming skills in Python
8	Image Compression by DCT, DPCM & HUFFMAN coding	OpenCV, NumPy, Matplotlib, Bitarray	Programming skills in Python
9	Implementation of Image Intensity slicing technique for image enhancement	OpenCV, NumPy, Matplotlib	Programming skills in Python
10	Implementation of Morphological operations in Image Processing	OpenCV, NumPy, Matplotlib	Programming skills in Python

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Medical Image Analysis for Tumor Detection	Python, OpenCV, NumPy, Matplotlib, Scikit-learn	Programming skills in Python
2	Automated Plant Disease Detection	Python, OpenCV, NumPy, Matplotlib, Scikit-learn	Programming skills in Python
3	Automated License Plate Recognition (ALPR)	Python, OpenCV, NumPy, Matplotlib, Tesseract OCR	Programming skills in Python
4	Facial Recognition for Security Systems	Python, OpenCV, NumPy, Dlib	Programming skills in Python

TEXT BOOKS:

1. Rafael C Gonzalez, Richard E. Woods, "Digital Image Processing", Pearson Education, 4th Edition, 2019.
2. K. Jain, "Fundamentals of Digital Image Processing", Pearson, 3rd Edition, 2004.
3. Joseph Howse, Joe Minichino, Learning OpenCV 4 Computer Vision with Python 3 3rd ed. Edition, Packt Publishing,2020.
4. Adrian Rosebrock "Practical Python and OpenCV" 3rd ed. Edition, pyimage research,2016

REFERENCE BOOKS:

1. Alasdair Mc Andrew; Introduction to Digital Image Processing; Cengage learning; 2009.

2. S. Jayaraman, S. Esakkirajan, T.Veerakumar, "Digital Image Processing", McGraw-Hill Education. (India) Pvt. Ltd., 2013.

3. Prateek Joshi, Opencv With Python by Example Paperback – Import, Packt Pub Ltd, 22 September 2015

JOURNALS/MAGAZINES:

1. IEEE transactions on Image processing

2. ACM Transactions on Image processing

3. Journal of Visual Communication and Image Representation

<table>
<tr><td>Course Title</td><td colspan="4">Big Data Analytics Lab</td><td colspan="2">Course Type</td><td colspan="2">HC</td></tr>
<tr><td>Course Code</td><td>B24EH0506</td><td colspan="2">Credits</td><td>1</td><td colspan="2">Class</td><td colspan="2">V Semester</td></tr>
<tr><td rowspan="5">Course Structure</td><td>LTP</td><td>Credits</td><td>Contact</td><td>Workload</td><td colspan="2" rowspan="2">Total Number of Classes Per Semester</td><td colspan="2" rowspan="2">Assessment in Weightage</td></tr>
<tr><td>Lecture</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Tutorial</td><td>-</td><td>-</td><td>-</td><td rowspan="2">Theory</td><td rowspan="2">Practical</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td></tr>
<tr><td>Practice</td><td>1</td><td>2</td><td>2</td></tr>
<tr><td>Total</td><td>1</td><td>2</td><td>2</td><td>0</td><td>28</td><td>25</td><td>25</td></tr>
</table>

COURSE OVERVIEW:

This course is to familiarize the students with most important information technologies used in manipulating, storing, and analyzing big data. The Spark framework is introduced to the students to perform big data analytics using PySpark. The course demonstrates the application of PySpark in solving any big data analytics problem. The course also demonstrates the usage of machine learning with PySpark to analyse large datasets.

COURSE OBJECTIVES:

The objectives of this course are to be:

1. Discuss the fundamentals of Spark distributed system and Big Data Analytics.
2. Demonstrate Big Data Processing with Python and PySpark.
3. Describe the implementation of Real-Time Analytics with PySparkin real world Applications.
4. Illustrate the working of Machine Learning applications suitable to handle big data using PySpark.

COURSE OUTCOMES:

On successful completion of this course; the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Illustrate the fundamentals of Spark distributed framework for Big Data Analytics.	1,2,3,4,5,9 ,10,11	1,2,3
CO2	Demonstrate Big Data Processing with PySpark to solve simple real world problems.	1,2,3,4,5,9 ,10,11	1,2,3
CO3	Design Real-Time Analytics with PySparkfor real world Applications.	1,2,3,4,5,9 ,10,11	1,2,3
CO4	Develop data and processing models using Python based PySparkfor real world Big data Applications	1,2,3,4,5,9 ,10,11	1,2,3
CO5	Design Real-Time Analytics with PySpark for real world Applications.	1,2,3,4,5,9 ,10,11	1,2,3
CO6	Develop data and processing models using Python based PySpark for real world Big data Applications.	1,2,3,4,5,9 ,10,11	1,2,3

BLOOM'S LEVELOF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		✓				
CO2			✓			
CO3		✓				
CO4			✓			
CO5			✓			
CO6			✓			

COURSE ARTICULATION MATRIX

Course Outcomes	Program Outcomes														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	3	3				1	1	1	1	3	3	3
CO2	3	3	3	3	3				2	1	2	3	3	3	3
CO3	3	3	3	3	3				2	1	2	3	3	3	3
CO4	3	3	3	3	3				3	3	2	3	3	3	3
CO5	3	3	3	3	3				2	1	2	3	3	3	3
CO6	3	3	3	3	3				3	3	2	3	3	3	3

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Evaluate the performance of MapReduce program on "word count" for different file size.	Windows/Linux OS, IDE/Cloudera	Understanding the process of Installation of Java and cloud era
2	Evaluate the performance of MapReduce program on "character count" for differentfile size.	Windows/Linux OS, IDE/Cloudera	Understanding the process of Installation of Java and cloud era

3	Develop a MapReduce program to find the grades of student's.	Windows/Linux OS, IDE/Cloudera	Understanding the process of Installation of Java and cloud era
4	Develop a MapReduce program to calculate the frequency	Windows/Linux OS, IDE/Cloudera	Understanding the process of Installation of Java and cloud era
5	Perform the following operations: a. Create a Spark Context object b. Create RDD from set of words ("scala", "java", "hadoop", "spark ", "akka", "spark vs hadoop", " pyspark", "pyspark and spark") using *parallelize()* function. c. Find the total count of the words in the RDD Filter out and print the strings containing the word "spark" from the RDD	Windows/ Linux OS, IDE/Colab	Understand the skill of performing operations on RDDs in PySpark
6	Given the two RDDs: a. x created from the ordered pairs: ("spark", 1) and ("Hadoop", 4) b. y created from the ordered pairs: ("spark", 2), ("Hadoop", 5). Perform the join operation on the RDDs created above, and print the resulting RDD.	Windows/ Linux OS, IDE/Colab	Understand the skill of performing operations on RDDs in PySpark
7	Create an RDD of set of numbers and perform the sum of these numbers using an *accumulator()* function in Spark context.	Windows/ Linux OS, IDE/Colab	Understand the skill of performing operations on RDDs in PySpark
8	Create an RDD from the external text file. Find the word count in the text file using various transformation and action functions in PySpark	Windows/ Linux OS, IDE/Colab	Understand the skill of performing operations on RDDs in PySpark

9	Given the following data. Data = [("James","Sales","NY",90000,34,10000), ("Michael","Sales","NV",86000,56,20000), ("Robert","Sales","CA",81000,30,23000), ("Maria","Finance","CA",90000,24,23000) , ("Raman","Finance","DE",99000,40,24000	Windows/ Linux OS, IDE/Colab	Understand the skill of creating PySparkdataframes from RDD and then performing operations on PySpark RDDs in PySpark
	), ("Scott","Finance","NY",83000,36,19000), ("Jen","Finance","NY",79000,53,15000), ("Jeff","Marketing","NV",80000,25,18000), ("Kumar","Marketing","NJ",91000,50,210 00)], with the following schema schema = ["employee_name","department","state" ,"salary","age","bonus"] Perform the following using the aforementioned data. i. create an RDD from the above data using its schema ii. create the PySpark dataframe from the RDD created. iii. Using groupBy() function, display the salaries of the employees state-wise. iv. Display the state-wise salaries that are greater than 1 lakh Display the state-wise salaries in descending order.		
10	Implement NoSQL Database Operations: CRUD operations, Arrays using Spark).	Windows/ Linux OS, IDE/Colab	Understand the skill of performing CRUD operations on spark

Project B			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Implement and demonstrate any real life big data problem using any of the publicly available big data sets.	Windows/Linux OS, IDE/Colab	Literature Surveying, Project Implementation, Seminars, IPR Filing, Paper Publication

Note: During the examination, the student is expected to do one of the first 7 exercises from Part-A by choosing on lots. The student is also expected to develop a Mini-Project using PySpark to solve some real time data analytics application. The overall examination marks are distributed into Part-A and Part-B. Part-B includes demonstration of project and submission of project report.

TEXT BOOKS:

1. Feng, Wenqiang. "Learning Apache Spark with Python." (2019): 231.

2. Mengle, Saket SR, and Maximo Gurmendez. Mastering machine learning on Aws: advanced machine learning in Python using SageMaker, Apache Spark, and TensorFlow. Packt Publishing Ltd, 2019.

REFERENCE BOOKS:

1. Michael Minelli, Michele chambers, AmbigaDhiraj,"Big data, big analytics", Wiley, 2013.

JOURNALS/MAGAZINES

1. IEEE, Introduction to the IEEE Transactions on Big Data.
2. Elsevier, Big data research journal

Elsevier. Springer, Journal on Big Data Springer.

Course Title	AEC-3(Ability Enhancement Course Placement Training)				Course Type		AEC	
Course Code	B24EF0507	Credit	1		Class		V SEMESTER	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	1	2	2				
	Total	**1**	**2**	**2**	**-**	**28**	**25**	**25**

COURSE OVERVIEW

Placement training classes can effectively be conducted as a part of an Ability Enhancement Course (AEC). These courses are designed to enhance students' skills, making them more prepared and competitive for the job market.

Course Title	Indian Knowledge System				Course Type		MC	
Course Code	B24ED0501	Credit	0		Class		V SEMESTER	
Course Structure	LTP	Credit	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	0	1	1				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	-	-	-				
	Total	**0**	**1**	**1**	**14**	**-**	**25**	**25**

COURSE OVERVIEW

The Indian Knowledge Systems comprise of Jnan, Vignan, and Jeevan Darshan that have evolved out of experience, observation, experimentation, and rigorous analysis. This tradition of validating and putting into practice has impacted our education, ar ts, administration, law, justice, health, manufacturing, and commerce. This has influenced classical and other languages of Bharat, that were transmitted through textual, oral, and artistic traditions. “Knowledge of India” in this sense includes knowledge from ancient India and, its successes and challenges, and a sense of India’s future aspirations specific to education, health, environment and indeed all aspects of life.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To provide a roadmap for systemic study of Indian knowledge system
2. To introduce students to the science and technological advancements related to Indian tradition.
3. To help students understand the Indian architecture, fine arts and agricultural system.
4. To help learners understand India’s rich legacies influence on the world heritage

COURSE OUTCOMES (COs):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Gain conceptual understanding of Indian culture and traditions.	6,8,10	
CO2	Appreciate the science and technological advancements in ancient India.	6,8,10	
CO3	Describe various ancient theories related to health, well- being and mindfulness	6,8,10	
CO4	Comprehend the Indian architecture, town planning, art and music.	6,8,10	
CO5	Understand India as a land united by cultural diversity.	6,8,10	

BLOOM’S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom’s Level					
	Remember (L1)	**Understand (L2)**	**Apply (L3)**	**Analyze (L4)**	**Evaluate (L5)**	**Create (L6)**
CO1	✓					
CO2	✓					
CO3	✓					
CO4	✓					
CO5	✓					
CO6	✓					

COURSE ARTICULATION MATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						2		2		2					
CO2						2		2		2					
CO3						2		2		2					
CO4						2		2		2					
CO5						2		2		2					
CO6						2		2		2					

Note:1-Low,2-Medium,3-High

COURSE CONTENT

TEXT BOOKS:

1. Sundararajan K.R., Hindu Spirituality - Vedas through Vedanta, Cross Road Publications, New York, 1997.

Introduction to IKS

Bharatavarsha-A Land of Rare Natural Endowments: Largest cultivable area in the world. Protected and nurtured by Himalayas. The Sindhu-Ganga plain and great coastal plains.

The great rivers of India-Abundant rains, sunshine and warmth, vegetation, animals and mineral wealth. Most populous country in the world. India's prosperity held the world in thrall. Splendid geographical isolation of India and the uniqueness of Indian culture.

Culture – Indus Valley Civilization and early cultural practices, The Vedic culture, Influence of Buddhism and Jainism on Indian Culture, Influence of Islam and Christianity, Indian Cultural Renaissance of the 19th Century Foundational literatures-Vedas, Ramayana, Mahabharata and the Puranas

Contribution of ancient India to Mathematics, Astronomy and Health Science

Development of Science in Ancient India- Mathematics, Astronomy and Health Science.

Mathematics- Numbers, fractions and geometry in the Vedas. Decimal nomenclature of numbers in the Vedas Zero and Infinity Simple constructions from Sulba-sutras.Science- Kanad, Varahamihira, Nagarjuna. Important texts of Indian mathematics Brief introduction to the development of algebra, trigonometry and calculus.

Astronomy- Ancient records of the observation of the motion of celestial bodies in the Vedic corpus. Sun, Moon, Nakshatra & Graha. Astronomy as the science of determination of time, place and direction by observing the motion of the celestial bodies. The motion of the Sun and Moon. Motion of equinoxes and solstices Elements of Indian calendar systems as followed in different regions of India.

Health Science- Vedic foundations of Ayurveda. Ayurveda is concerned both with maintenance of good health and treatment of diseases. Basic concepts of Ayurveda. The three Gunas and Three Doshas, Pancha-mahabhuta and Sapta-dhatu. The importance of Agni (digestion). Six Rasas and their relation to Doshas Ayurvedic view of the cause of diseases. Dinacharya or daily regimen for the maintenance of good health, Ritucharya or seasonal regimen Important Texts of Ayurveda Selected extracts from Astängahrdaya (selections from Sätrasthana) and Susruta- Samhita (sections on plastic surgery, cataract surgery and anal fistula).

2. Griffiths Bede, Yoga and the Jesus Prayer Tradition, Asian Trading Corporation, Bangalore, 1992
3. Ansh Mishra, Science in Ancient India, Indian Corporation, New Delhi, 1998
4. Sen Taylor, Collen. Feasts and Fasts: A History of Food in India. Reaktion Books, New Delhi, 2014.
5. Thapar, Romila, Readings in Early Indian History. Oxford University Press. New Delhi, 2018
6. Baladev Upadhyaya, Samskrta Sastrom ka lihas, Chowkhambha, Varanasi, 2010.
7. D. M. Bose, S. N. Sen and B. V Subbarayappa, Eds, A Concise History of Science in India, 2nd Ed. Universities Press, Hyderabad, 2010.
8. Astangahrdaya, Vol 1. Sutrasthana and Sarirasthana, Translated by K. R. Srikantha Murthy.Vol. I, Krishnadas Academy. Varanasi, 1991.
9. Dharampal, Indian Science and Technology in the Eighteenth Century: Some Contemporary European Accounts, Dharampal Classics Series, Rashtrotthana Sahitya, Bengaluru, 2021.
10. JK. Bajaj and M. D Srinivas, Indian Economy and Polity in Eighteenth century Chengalpattu, in J. K. Bajaj ed., Indian Eco nomy and Polity, Centre for Policy Studies, Chennai, 1995, pp. 63-84.

VI Semester

Sl. No	Course Code	Title of the Course	HC/ FC/ SC/OE /MC	Credit Pattern				Contact Hours / Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0601	Speech and Language Processing with AI	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0602	Generative Artificial Intelligence	HC	3	0	0	3	3	50	50	100	PCC
3	B24EH0603	Cloud Computing Tools	HC	3	0	0	3	3	50	50	100	PCC
4	B24EH0604	Human-Computer Interaction	HC	3	0	0	3	3	50	50	100	PCC
5	B24EHS61X	Professional Elective-5	SC	3	0	0	3	3	50	50	100	PEC
6	B24EFO61X	Open Elective-2	OE	3	0	0	3	3	50	50	100	OE
7	B24EH0605	Generative Artificial Intelligence lab	HC	0	0	1	1	2	25	25	50	PCC
8	B24EH0606	Cloud Computing Tools Lab	HC	0	0	1	1	2	25	25	50	PCC
9	B24EH0607	Speech and Language Processing Lab	HC	0	0	1	1	2	25	25	50	PCC
10	B24EH0608	Mini Project	HC	0	0	2	2	4	50	50	100	Proj
11	B24EF0609	AEC-4(Placement Training)	AEC	0	0	1	1	2	25	25	50	AEC
TOTAL				18	0	6	24	30	450	450	900	

TOTAL SEMESTER CREDITS	24
TOTAL CUMULATIVE CREDITS	135
TOTAL CONTACT HOURS	30
TOTAL MARKS	900

Course Title	Speech and language processing with AI				Course Type		HC	
Course Code	B24EH0601	Credits	3		Class		VI SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

Speech and language processing is a broad field that focuses on the study and development of algorithms and systems capable of understanding, generating, and manipulating human language in various forms, including speech and text

COURSE OBJECTIVE (S):

The objective of this course are to

1. To understand the structure and properties of human language, including syntax, semantics, and pragmatics
2. To learn the techniques for processing speech signals, including speech recognition, speech synthesis, and
3. acoustic modeling
4. To understand machine learning approaches for Natural Language Processing tasks
5. To Explore the design and implementation of dialogue systems and conversational agents

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Describe the phonetic, lexical, syntactic, and pragmatic properties of human language and the associated challenges for automatic processing	1,2,3,5	1,2,3
CO2	Explain the algorithms and representations underlying applications such as text classification, speech recognition, speech synthesis, question answering, information extraction, machine translation, search, and conversational agents.	1,2,3,5	1,2,3
CO3	Apply basic machine learning concepts and techniques to speech and language processing problems.	1,2,3,5	1,2,3
CO4	Design and implement dialogue systems and conversational agents, including techniques for dialogue management, natural language generation, and user interaction modeling	1,2,3,5	1,2,3
CO5	Explain the nature, uses, and limitations of large language models and other pretrained models	1,2,3,5	1,2,3
CO6	Use the tools like Python, NLTK, TensorFlow, and Kaldi and able to implement and evaluate language processing systems	1,2,3,5	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2		√				
CO3			√			
CO4			√			

CO5		√				
CO6		√				

COURSE ARTICULATIONMATRIX

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	2	3		2								3	2	3
CO2	3	2	2		2								3	2	2
CO3	2	3	3		2								2	3	3
CO4	3	2	2		2								2	2	2
CO5	2	3	3		2								2	3	3
CO6	3	3	3		2								3	2	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction :
Overview of Language Applications , Rules and Tendencies, Feature Design,
Text Classification with a Linear Model ,Model Evaluation

Sequences, Context, Ngrams, and Language Modeling

Edit Distances ,The Noisy Channel Model , Bigrams and Beyond , Tagging , Sequence-to-Sequence Mapping , Language Modeling , Generative Modeling

Pattern Matching and Regular Expressions

Regular Expressions in Python ,Tokenization

Words and Representations of Meaning

Bag-of-Words , Lexical Disambiguation ,Vector-Space Similarity ,Information Retrieval

Grammatical Structures and Logical Forms

Constituency, Context-Free Grammars, and Syntactic Ambiguity, Chunks , Logic-Based and Graph-Based Meaning Representations .

Pretraining Language Models
Word Embeddings and Context Vectors, Large Language Model Training , Generative Models, revisited .

Sound, Phonetics, and Prosody
Acoustic Phonetics and Spectral Representations, Articulatory Phonetics and Phonemes (Guest Lecture: Dr. Mueller), The Noisy Channel Model, Speech Recognition, Speech Synthesis inferring Speaker States and Traits.

Dialog Structure and Dialog Flow
Finite-state Dialog Management ,Retrieval-Based Dialog and Chat bots , End pointing and Turn Taking ,Pragmatics, Dialog Acts, and User Intentions, Natural Language Generation ,Advantages and Disadvantages of Natural Language Interfaces

Self-Learning Component:

Spelling Correction, Spam Filtering, Machine Translation, Call-Center Analytics, InformatiomExtraction (Entity Extraction and Linking) ,Question Answering ,Language Proficiency Assessment Tutoring and Training, Summarization

TEXT BOOKS:

1. Daniel Jurafsky and James H Martin, “Speech and Language Processing – An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition”, Pearson Education.
2. Natural Language Processing with Python" by Steven Bird, Ewan Klein, and Edward Loper

REFERENCE BOOKS:

1. Speech and Audio Signal Processing: Processing and Perception of Speech and Music by Ben Gold and Nelson Morgan

JOURNALS/MAGAZINES:

1. IEEE/ACM Transactions on Audio, Speech, and Language Processing
2. Journal of Acoustical Society of America (JASA)
3. Journal of Speech, Language and Hearing Research (JSLHR)
4. IEEE Signal Processing Magazines

SWAYAM NPTEL/MOOCs

1. https://onlinecourses.nptel.ac.in/noc23_cs45/preview
2. https://archive.nptel.ac.in/noc/courses/noc19/SEM2/noc19-cs56/

Course Title	Generative Artificial Intelligence				Course Type		HC	
Course Code	B24EH0602	Credits	3		Class		VI SEMESTER	
	LTP	Credits	ontact Hours	Workl oad	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial		-	-				
	Practice		-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

Prerequisites: Deep Learning Fundamentals, Probability and Statistics

COURSE OVERVIEW:

This course introduces the fundamental concepts and techniques of generative AI models. Students will gain an understanding of how these models can create new data, explore various applications, and analyze the ethical considerations surrounding this powerful technology.

COURSE OBJECTIVE (S):

The objective of this course are to

1. Explain the core principles of generative AI models.
2. Illustrate different generative model architectures (e.g., VAEs, GANs).
3. Demonstrate generative models for tasks like text generation, image synthesis, and music creation.
4. Discuss the ethical considerations and potential societal impacts of generative AI.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Illustrate the architectural concepts in deep neural networks and generative modelling	1 to 5	1,2,3
CO2	Experiment with the components of Variational autoencoders for data reconstruction	1 to 5	1,2,3
CO3	Make use of the GAN concepts to learn image generation	1 to 5	1,2,3
CO4	Inspect the transformer architecture for its superior performance for generative tasks	1 to 5	1,2,3
CO5	Utilize large language models for text generation	1 to 5	1,2,3
CO6	Analyze the ethical challenges of generative AI and its applications.	1 to 8	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		✓				
CO2			✓			
CO3				✓		
CO4			✓			
CO5			✓			
CO6				✓		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2								3	3	2
CO2	2	3	3	3	2								3	3	2
CO3	3	3	3	3	3								3	3	2
CO4	3	3	3	3	3								3	3	2
CO5	3	3	3	3	3								3	3	2
CO6	3	2	2	3	3	3	3	3					3	3	2

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Generative AI:

Deep Learning Concepts: Data – Structured and unstructured, Deep learning – building blocks of deep neural networks (architecture, activation functions, optimization algorithms, parameter initialization), multilayer perceptron and back propagation, different deep neural architectures - CNN, RNN.

Generative modelling –The rules of probability, what is generative modelling, generative vs discriminative modelling. Discriminative and generative modeling and
Bayes' theorem, the generative modelling framework, probabilistic generative models, The challenges of Generative modelling.

Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs):

Variational Autoencoders (VAE): VAE architecture, loss function design in VAE – reconstruction loss, divergence loss, Techniques for training and evaluating VAEs, Applications of VAEs.

Generative Adversarial Networks (GAN): GAN architecture, training a GAN, GAN Challenges – Oscillating Loss, Mode collapse, Uninformative Loss, Large number of Hyperparameters, Tackling GAN Challenges – Wasserstein GAN, The Lischitz Constraint, Weight Clipping

Transformer Architecture and Large Language Models:

Converting text to numbers, Text Embeddings, Attention mechanism, Transformer architecture, Positional Encoding, BERT, GPT-2

Autoregressive Models, Language Modelling, Large Language models, Pretraining and fine tuning, few shot learning, Challenges with large language models, prompt engineering - the principles of writing effective prompts, Retrieval Augmented Generation.

Vision Models- Stable Diffusion for Image Generation

Ethical Consideration and Applications of Generative AI:

Challenges with generative AI – data privacy, safety, bias in output, hallucinations, jailbreaks. Ethical considerations for responsible development and use of generative AI models, societal impact of generative AI

Applications of generative ai for text generation, image synthesis, music generation, video creation, in drug discovery, material science, engineering design, entertainment.

Self-Learning Component: Variants of GAN, Generative modelling for scientific discovery, Generative AI and Cybersecurity.

TEXTBOOKS:

1. Generative Deep Learning, Teaching Machines to Paint, Write, Compose and Play, David Foster, Oreilly, 2019.

2. Generative AI with Python and TensorFlow, Joseph Babcock, Raghav Bali, Packt, 2021.

REFERENCE BOOKS:

1. Introduction to Generative AI, Numa Dhamani, Maggie Engler, Manning Publications Co., 2024,

JOURNALS/MAGAZINES:

1. Generative Artificial Intelligence: Trends and Prospects, IEEE
2. Generative AI for Software Practitioners, IEEE

SWAYAMNPTEL/MOOCs

1. Introduction to Language Models, NPTEL.
2. Generative AI with Large Language Models, Coursera, https://www.coursera.org/learn/generative-ai-with-llms

Course Title	Cloud Computing Tools				Course Type		HC	
Course Code	B24EH0603	Credits	3		Class		VI SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:
This course gives students an insight into the basics of cloud computing and tools. It also provides the concepts of virtualization along with it how one can migrate over it. It will provide the students basic understanding about cloud computing tools like Cloudsim, green cloud and Vmware Simulator etc.

COURSE OBJECTIVE (S):
The objective of this course are to

1. Explain the core concepts of the cloud computing paradigm and how it represents a significant shift from traditional computing models
2. Understand the fundamental concepts of datacenters and how load balancing approaches impact power, efficiency, and cost
3. Design and develop highly scalable cloud-based applications by creating and configuring virtual machines on the cloud and building private clouds
4. Identify the key architecture and infrastructure components of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, and hybrid cloud
5. Make informed recommendations on cloud computing solutions for enterprises based on their specific requirements and constraints
6. Critically analyze case studies to derive best practices for developing and deploying cloud-based application

COURSE OUTCOMES (COs)
After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about.	1 to 4, 12	1,2,3
CO2	To apply the fundamental concepts in datacenters to understand the tradeoffs in power, efficiency and cost by Load balancing approach.	1 to 5,12	1,2,3
CO3	To design and develop highly scalable cloud-based applications by creating and configuring virtual machines on the cloud and building private cloud.	1 to 5	1, 2, 3
CO4	To Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.	1 to 5, 8 to 10	1,2, 3
CO5	To Make recommendations on cloud computing solutions for an enterprise.	1-4,12	1,2,3
CO6	To Critically analyze case studies to derive the best practice tools to apply when developing and deploying cloud based applications	1-5, 8,9,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3		√				
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	3	2	1	2	1								1	2	2
CO2	3	2	1	2	1								2	3	2
CO3	3	3	2	2	1							2	1	2	3
CO4	3	3	3	2	2				2			2	3	2	3
CO5									2			2	3	3	3
CO6	3	2	2	1	1				3			3	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Cloud Computing Overview & Delivery Model:

Introduction to Cloud Computing, History and Evolution of Cloud Computing, Types of clouds, Private Public and hybrid clouds, Cloud Computing architecture, Cloud computing infrastructure, Merits of Cloud computing, , Cloud computing delivery models and services (IaaS, PaaS, SaaS), obstacles for cloud technology, Cloud vulnerabilities, Cloud challenges, Practical applications of cloud computing.

Cloud deployment model: Public clouds, Private clouds, Community clouds, Hybrid clouds and Advantages of Cloud computing. Hypervisors and Virtual machines Server Virtualization: Understanding Server Virtualization, types of server virtualization, Virtual machine basics, types of virtual machines, hypervisor concepts and types

Introduction to VMWare Simulator

Basics of VMWare, advantages of VMware virtualization, using Vmware workstation, creating virtual machines-understanding virtual machines, create a new virtual machine on local host, cloning virtual machines, virtualize a physical machine, starting and stopping a virtual machine.

Cloud Simulators- CloudSim and GreenCloud

Introduction to Simulator, understanding CloudSim simulator, CloudSim Architecture(User code, CloudSim, GridSim, SimJava) Understanding Working platform for CloudSim, Introduction to GreenCloud.

TEXT BOOKS:

1. Cloud computing for dummies- Judith Hurwitz , Robin Bloor , Marcia Kaufman ,Fern Halper, Wiley Publishing, Inc, 2010
Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011

2. Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter, TATA McGraw- Hill , New Delhi – 2010 Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008

3. David Marshall, Wade A. Reynolds, Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach

REFERENCE BOOKS:

1. Cloud Computing: Web-Based Applications That Change the Way You Work and CollaborateOnline - Michael Miller - Que 2008

2. John Rhoton, Cloud Computing Explained: Handbook for Enterprise Implementation

3. Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg,Andrzej Goscinski, John Wiley & Sons, Inc. 2011.

4. Building Applications in the Cloud: Concepts, Patterns and Projects by Christopher M. Moyer, Pearson Addison-Wesley Professional, ISBN-10: 0321720202, ISBN-13: 978-0321720207

5. Cloud Computing Design Patterns by Thomas Erl, Prentice Hall, ISBN-10: 0133858561, ISBN-13: 978-0133858563

JOURNALS/MAGAZINES:

1. https://www.sciencedirect.com/science/article/pii/S187705092031557X

2. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9152086

3. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9257359

4. https://www.researchgate.net/publication/24164535_CloudSim_A_Novel_Framework_for_Modeling_and_Simulation_of_Cloud_Computing_Infrastructures_and_Services

SWAYAMNPTEL/MOOCs

1. https://intellipaat.com/cloud-devops-architect-masters-program-training/

2. https://onlinecourses.nptel.ac.in/noc22_cs18/preview

Course Title	Human Computer Interface				Course Type		HC	
Course Code	B24EH0604	Credits	3		Class		VI SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course presents the foundations of Human Computer Interaction (HCI). The contents are structured into phases comprising: Basic definitions and motivations of HCI, interaction paradigms, design principles and models, User-centred design methods comprising user studies, design approaches for interfaces and interaction, evaluation methods and techniques for data analysis, Research frontiers of HCI, including accessibility, universal design, and pervasive computing (ubiquitous, mobile and wearable computing).

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Describe and analyze the human and computer processing models, identifying their key characteristics and differences
2. Compare and contrast the various interaction models, critically discussing the significance of ergonomics in HCI
3. Analyze the main approaches to user support and their effectiveness in enhancing the user experience
4. Evaluate HCI design rules and usability principles, and apply them effectively in software development projects
5. Implement universal design principles and multi-modal interaction by designing user interfaces for interactive systems
6. Analyze and apply different evaluation techniques to measure the quality and effectiveness of user interfaces

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Describe and analyze the human and the computer processing models.	1 to 4,12	1,2,3
CO2	Identify and compare the different interaction models and critically discuss the significance of ergonomics.	1 to 4,12	1,2,3
CO3	Analyze the main approaches to user support.	1 to 5,12	1,2,3
CO4	Evaluate HCI design rules and usability principles and their application on software development	1 to 5,12	1,2,3
CO5	Implement universal design principles and multi-modal interaction by designing a user interface for an interactive system.	1 to 5, 8 to 10	1,2,3
CO6	Analyze the different evaluation techniques used to measure the quality of User Interface.	1 to 5,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		✓				
CO2		✓				
CO3				✓		

CO4					✓	
CO5			✓			
CO6				✓		

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	2								2	3	1	1
CO2	2	3	2	1		2						2	3	2	2
CO3	2	3	3	2									3	3	3
CO4	3	3	3	2				2	2	2			3	3	3
CO5	2	3	2	2									3	3	3
CO6	3	3	2	2								2	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Human and the Computer: Human: Input–output channels, Human memory, Thinking: reasoning and problem solving, Emotion, Individual differences, Psychology and the design of interactive systems. The computer: Positioning, pointing and drawing, Display devices, Devices for virtual reality and 3D interaction, Physical controls, sensors and special devices, Paper: printing and scanning, Memory, Processing and networks

The interaction and Paradigms: Models of interaction, Frameworks and HCI, Ergonomics, Interaction styles, Elements of the WIMP interface, Interactivity, The context of the interaction. Paradigms: Paradigms for interaction. Interaction design basics: The process of design, User focus, Scenarios, Navigation design, Screen design and layout, Iteration and prototyping.

HCI in the software process and Design rules: The software life cycle, Usability engineering, Iterative design and prototyping. Principles to support usability, Standards, Guidelines, Golden rules and heuristics, HCI patterns. Universal designs.

Evaluation techniques: Goals of evaluation, Evaluation through expert analysis, Evaluation through user participation, choosing an evaluation method.

Self-Learning Component:

1. Designing user support systems
2. Ubiquitous computing applications research

TEXT BOOKS:

1. Alan Dix, Janet Finlay, Gregory Abowd & Russell Beale, Human-Computer Interaction. 3rd Edition. Prentice Hall, 2004.
2. Julie A. Jacko, Human-Computer Interaction Handbook, 3rd Edition, CRC Press, 2012.
3. Ben Shneiderman, Catherine Plaisant, Designing the User Interface, 6th Edition, Addison Wesley, 2017.

REFERENCE BOOKS:

1. Jonathan Lazar, Jinjuan Heidi Feng, & Harry Hochheiser Research Methods in Human-Computer Interaction, Wiley, Second edition, 2010.
2. ACM, International Journal of Human-Computer Studies.
3. IEEE, Transactions on Human-Machine Systems.
4. Elsevier, International Journal of Human-Computer Studies.

JOURNALS/MAGAZINES:

1. ACM Transactions on Computer-Human Interaction
2. ACM Transactions on Accessible Computing
3. Advances in Cognitive Psychology
4. Advances in Human-Computer Interaction
5. Foundations and Trends in Human-Computer Interaction
6. International Journal of Human-Computer Interaction
7. International Journal of Human-Computer Studies
8. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans

SWAYAMNPTEL/MOOCs

1. https://nptel.ac.in/courses/106103115
2. https://archive.nptel.ac.in/courses/106/106/106106177/
3. https://nptel.ac.in/courses/106103237
4. https://www.mooc-list.com/tags/human-computer-interaction
5. https://www.edx.org/certificates/professional-certificate/gtx-human-computer-interaction

Professional Elective 5

Course Title	Machine Learning Operations				Course Type		SC	
Course Code	B24EHS611	Credits	3		Class		VI SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course provides an in-depth understanding of the practices, tools, and frameworks involved in deploying, monitoring, and maintaining machine learning models in production environments. Students will learn how to bridge the gap between data science and operations, ensuring robust, scalable, and maintainable machine learning systems.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Understanding the components and structure of MLOps Frameworks.
2. Demonstrate the concepts of Machine Learning/ Technologies in MLOps Tools .
3. Learn to Deploy Machine Learning models in productions.
4. Demonstrate Building and Automating Machine Learning models in AutoML and KaizenML.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Understand the concepts of Machine Learning operations and its frameworks.	1,2,3,5,11,12	1,2,3
CO2	Learn to configure cloud shell environment and building MLops	1,2,3,4,5,9,1 1,12	1,2,3
CO3	Understand the advantage of MLOps for Containers and Edge Devices	1,2,3,4,5,6,9, 11,12	1,2,3
CO4	Demonstrate the strategies of ML Model deployment	1,2,3,5,12	1,2,3
CO5	Outline the practical aspects of deploying machine learning systems.	1,2,3,4,5,9,1 1,12	1,2,3
CO6	Learn to deploy ML models in Azure, AWS and other tools	1,2,3,4,5,6,9, 11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember	Understand	Apply	Analyze	Evaluate	Create
CO1		✓				
CO2			✓			
CO3			✓			
CO4			✓			
CO5				✓		
CO6				✓	✓	

COURSE ARTICULATIONMATRIX:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	3	2	2	3						3	3	1	2	2
CO 2	3	2	3	3	3				3		3	2	2	3	2
CO 3	3	2	3	3	3	2			3		3	3	1	2	3
CO	2	3	2	2	3							3	3	2	3
CO 5	3	2	3	2	3				3		3	2	3	3	3
CO 6	3	2	3	3	3	2			3		3	3	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Machine Learning Operations
Why Now and Challenges Defining MLOps and Its Challenges, MLOps to Mitigate Risk, MLOps for Responsible AI
Key MLOps Features: Model Development, Productionalization and Deployment, Monitoring
Introduction to MLOps : Rise of the Machine Learning Engineer and MLOps, DevOps and MLOps, An MLOps Hierarchy of Needs,

MLOps Foundations: Cloud Shell Development Environments, Cloud Computing Foundations and Building Blocks, Machine Learning Key Concepts, Build an MLOps Pipeline from Zero
MLOps for Containers and Edge Devices:Containers,edge devices – coral, Containers for Managed ML Systems

Model Deployment: Static and dynamic deployment, Deployment strategies, automated Deployment, Versioning, and Metadata, Model Deployment Best Practices

AutoML and KaizenML: AutoML, Apple's Ecosystem, Google's AutoML and Edge Computer Vision, AWS AutoML, Open Source AutoML Solutions

Self Learning Component:

1. Azure tool
2. AWS cloud
3. Docker hub
4. Putty with Github
5. TensorFlow

TEXT BOOKS:

1. Mark Treveil, and the Dataiku Team, "Introducing MLOps: How to Scale Machine Learning in the Enterprise", O'Reilly Media, Inc. 2021
2. Noah Gift and Alfredo Deza,"Practical MLOps", O'Reilly Media, Inc. 2021
3. Andriy Burkov,"Machine Learning Engineering", True Positive Inc.2020

REFERENCE BOOKS:

1. Khaled El Emam, Lucy Mosquera, and Richard Horne, "Machine Learning Operations (MLOps): A Complete Guide to Managing Data Science Projects" .

2. Emmanuel Ameisen,"Building Machine Learning Powered Applications: Going from Idea to Product"

JOURNALS/MAGAZINES:

1. Journal of Machine Learning Research

2. Machine Learning Journal (Springer)

SWAYAMNPTEL/MOOCs:

1. Coursera MLOps Specialization

2. Udacity MLOps Nanodegree

Course Title	SIGNALS AND SYSTEMS				Course Type		SC	
Course Code	B24EHS612	Credits	3		Class		VI SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	

COURSE OVERVIEW:
Signals and Systems is an introductory course that covers analog and digital signal processing, which is a fundamental topic in many areas of engineering such as communications, speech processing, image processing, and control systems. The course presents concepts for both continuous-time and discrete-time signals and systems, developing representations in both time and frequency domains.

COURSE OBJECTIVE(S):
The objectives of this course are :

1. Provide insight into fundamentals of Discrete-time signals and systems, their properties and representations.
2. Introduce time domain representation of Linear Time invariant Systems such as Convolution Sum.
3. Provide understanding of signal representation in Fourier domain such as Fourier Series, Fourier transform, discrete time Fourier transform.
4. Provide insights into applications of Fourier transform and brief understanding of signal representation in Z-domain.

COURSE OUTCOMES(COs)
After the completion of the course, the students will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Analyze the basics properties of discrete time signals and systems.	1,2,3,4,5,9,10	1 and 3
CO2	Apply Convolution operation on an LTI System to calculate the output.	1,2,3,4,5,9,10	1 and 3
CO3	Identify Fourier representations and analyze the signals in time and frequency domain.	1,2,3,4,5,9,10	1 and 3
CO4	Apply Fourier representations to mixed signals and sampling theorem to discretize the analog signals.	1,2,3,4,5,9,10	1 and 3
CO5	Analyze discrete time signals & systems using Z transforms.	1,2,3,4,5,9,10	1 and 3
CO6	Represent the discrete time system in Z-domain and determine the behavior of Causal LTI system using properties of Z-Transform.	1,2,3,4,5,9,10	1 and 3

BLOOM'SLEVELOFTHECOURSEOUTCOMES:

	Bloom's Level					
CO#	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1	✓	✓	✓	✓		
CO2	✓	✓	✓	✓		
CO3	✓	✓	✓	✓		

CO4	✓	✓	✓	✓		
CO5	✓	✓	✓	✓		
CO6	✓	✓	✓	✓		

URSEARTICULATIONMATRIX:

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	1	1	2				2	2					
CO2	3	3	2	1	2				2	2					
CO3	3	2	1	1	2				2	2					
CO4	3	3	1	1	2				2	2					
CO5	3	3	2	1	2				2	2					
CO6	3	2	1	1	2				2	2					

Note:1-Low,2-Medium,3-High

COURSE CONTENT

Introduction and classification of signals: Definition of signal and systems with examples, Elementary signals /Functions: Exponential, sinusoidal, step, impulse and ramp functions.
Basic operations on discrete time signals: Amplitude scaling, addition, multiplication, time scaling, time shift and time reversal. Expression of triangular, rectangular, and other waveforms in terms of elementary signals.
Discrete time system classification and properties: Linear-nonlinear, time variant-invariant, causal-noncausal, static-dynamic, stable-unstable, invertible-noninvertible.

Time domain representation of discrete time LTI Systems: Impulseresponse, convolution sum, computation of convolution sum using graphicalmethod for two finite/infinite sequences (unit step and unit step, unit stepand exponential, exponential and exponential, unit step and rectangular,rectangular and rectangular).
Discrete time LTI system Properties in terms of impulse response:
Memory less, Causal, Stable, Invertible and Deconvolution, step response ofan LTI system, and LTI system interconnection.
Difference equation representation of LTI systems: Difference equationrepresentation and its solution by recursive method.
Fourier representations for signals: Introduction to Fourier series and Fourier transform (Qualitative analysis only).
Fourier representations for discrete time signals: Discrete time periodic signals: DFTS representation and properties of DTFS, Discrete time non- periodic signals: DTFT representation and properties of DTFT.
Discrete Fourier Transform: Sampling of DTFT, DFT and inverse DFT.
The Z-Transforms: Z-transform, properties of the region of convergence, properties of the Z-transform, Inverse Z-transform by partial fraction expansion and long division method, Transform analysis of LTI system, Causality and stability, Unilateral Z-transform and its application.

TEXTBOOKS:

1. Simon Haykin, “Signals and Systems”, John Wiley, India Pvt Ltd, Second Edition, 2018.
2. S Palani, “Signals and Systems”, Springer International Publishing AG, Second Edition, 2021.
3. I J Nagrath, “Signals and Systems”, Tata McGraw Hill, 3rd edition, 2010

REFERENCEBOOKS:

1. Michael Roberts, “Fundamentals of signals and systems”, TATA McGraw Hill, Second Edition ,2010
2. Allan V. Oppenheim, S. Wilsky and S. H. Nawab, “Signals and Systems”, Pearson Education, Second Edition, 2019.
3. Benoit Boulet, “Fundamentals of Signals and Systems”, Da Vinci Engineering Press, 2nd edition, 2006.

JOURNALS/MAGAZINES:

1. https://ieeexplore.ieee.org/abstract/document/9244176
2. https://ocw.mit.edu/resources/res-6-007-signals-and-systems-spring-2011/lecture-notes/
3. https://www.khanacademy.org/science/electrical-engineering/ee-signals
4. http://bonnie.ece.gatech.edu/book/TUTORIAL/tutorial.html
5. https://stanford.edu/~boyd/ee102
6. https://www.springer.com/journal/34
7. https://www.inderscience.com/jhome.php?jcode=ijsise
8. https://ieeexplore.ieee.org/document/1143815
9. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=79
10. https://www.ieee.org/membershipcatalog/productdetail/showProductDetailPage.html?product=PER310-PRT

SWAYAMNPTEL/MOOCs

1. https://nptel.ac.in/courses/108/104/108104100/
2. https://nptel.ac.in/courses/117/101/117101055/
3. https://nptel.ac.in/courses/108/106/108106163/
4. https://www.coursera.org/courses?query=signals%20and%20systems
5. https://nptel.ac.in/courses/117/104/117104074/

Course Title	Cyber Physical Systems				Course Type		
Course Code	B22EIO512	Credits			Class		
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester	Assessment Weightage	
	Lecture	3	3	3			
	Tutorial	-	-	-			
	Practice	-	-	-	Theory	CIE	SEE
	Total	**3**	**3**	**3**	**42**	**50%**	**50%**

COURSE OVERVIEW:

Cyber-physical systems (CPS) is an amalgamation of physical processes, wireless networking and extensive computing. The physical processes are monitored with varied sensors and then communicated through networking to create an automated feedback-based embedded control system. The automated/ smart systems are modelled from the data points through technological abstractions of IoT, intelligent systems, machine learning and pattern recognition.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Infer the mathematical models behind cyber-physical systems.
2. Study different techniques and security attacks.
3. Understandtheinteractionsandissuesrelatedtocyber-physicalsystems.
4. Developanexpositionofthechallengesinimplementingacyber-physicalsystem
5. Expose the student to real world problems in this domain and provide a walkthrough the design and validation problems for such systems.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Construct the symbolic models for a given cyber physical system	1,2,3,4	1
CO2	Summarize the design aspects of controller system in cyber physical system perceptive.	1,2	1
CO3	Inspect the security aspects of various cyber physical systems.	1,2,11	1
CO4	Compare different synchronous and asynchronous architectures to improve the interactions of globally connected (distributed) CPS.	1,2	1
CO5	Analyze the existing challenges of security in cyber physical systems.	1,2,4,5,9.11	1
CO6	Interpret the different cyber physical systems applications.	2,4,5	1

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1			√			
CO2		√				
CO3				√		
CO4		√				
CO5				√		
CO6			√			

COURSE ARTICULATIONMATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	3									1		
CO2	2	2											3		
CO3	2	2									3		1		
CO4	2	2											3		
CO5	2	2		3	2				2		3		3		
CO6		2		2	3								3		

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

SYMBOLIC SYNTHESIS FOR CYBER-PHYSICAL SYSTEMS : Introduction and Motivation, Basic Techniques, Layered Models in CPS, CPS design steps, Problem Definition, Solving the Synthesis Problem, Construction of Symbolic Models, Advanced Techniques.

SOFTWARE AND PLATFORM ISSUES IN FEEDBACK: CONTROL SYSTEMS: Introduction, Basic Techniques, Controller Timing, Controller Design for resource efficiency, Advanced Techniques. Logical Correctness of Hybrid Systems: Introduction, Basic Techniques, Discrete Verification, Advanced Techniques.

SECURITY OF CYBER-PHYSICAL SYSTEMS: Cyber Security Requirements, Attack Model, Counter measures, CPS attacks, Advanced Techniques, Internet of Things security, SDN, Network security and challenges.

SYNCHRONIZATIONINDISTRIBUTEDCYBER-PHYSICALSYSTEMS&APPLICATIONS: Challenges in Cyber-Physical Systems, A Complexity Reduction Technique for Synchronization, Basic Techniques, CPS tools ,Medical Cyber-Physical Systems, Energy Cyber-Physical Systems, Cyber Physical Systems platform components, implementation issues and secure Deployment.

Self Learning Component:

- Northrop Grumman Corporation's work at the intersection of cyber-physical systems and critical infrastructure
- Raytheon Company's work at the intersection of cyber-physical systems and autonomy
- Telos Corporation's work at the intersection of cyber-physical systems and the Internet of Things

TEXT BOOKS:

1. Raj Kumar R., de Niz, D., and Klein, M., "Cyber-Physical Systems", 1st Edition, Addison-Wesley Professional, 2017.

2. AlurR.,"PrinciplesofCyber-PhysicalSystems",MITPress,2015.

3. Roy,S.,andDas,S.K.,"PrinciplesofCyber-PhysicalSystems:AnInterdisciplinaryApproach", CambridgeUniversityPress,2020.

4. Alasdiar, GillChryst," IotSecurityIssues",De-Gruyter.

REFERENCE BOOKS:

1."EmbeddedSystemsFoundationsofCyber-PhysicalSystems", PeterMarwedel,Springer.
2."PrinciplesofCyber-PhysicalSystems", RajeevAlur,MITPress.

JOURNALS/MAGAZINES:

1. https://digital-library.theiet.org/content/journals/iet-cps
2. https://www.sciencedirect.com/topics/computer-science/cyber-physical-systems

SWAYAMNPTEL/MOOCs

https://onlinecourses.nptel.ac.in/noc23_cs62/preview

Open Elective -2

Course Title	Augmented reality and Virtual reality				Course Type		OE	
Course Code	B24EFO611	Credits	3		Class		VI Semester	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

This course covers basic concepts of augmented reality and virtual reality. The course also introduces the student to the working of multiple models of input and output interface in VR. The course also helps the student to understand development tools and frameworks in VR. Further, this course helps the student to work on the application of VR in digital entertainment

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the principles and multidisciplinary features of Virtual Reality.
2. Illustrate the multimodal user interaction and perception in Virtual Reality.
3. Demonstrate the use of objects for managing large scale Virtual Reality environment in real time.
4. Discuss the various solutions using Virtual Reality system framework and development tools for industry and social relevant applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify the components of Augmented and Virtual Reality	1 ,10 1	1
CO2	Apply multimodal user interaction and perception techniques involved in Virtual Reality.	1 to 3,5,10	3
CO3	Develop real world applications using Simulation and Interactive techniques.	1 to 3, 10	2,3
CO4	Choose the innovative Virtual Reality solutions for industrial and social relevant applications.	1,10	1
CO5	Make use of unity3D to develop innovative project.	1 to 3, 10	2,3
CO6	Apply VR modeling techniques to solve real world applications.	1,2,10	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			

CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3									3			3		
CO2	3	3	3		3					3					3
CO3	3	3	3							3				3	3
CO4	3									3			3		
CO5	3	2	2							3				3	3
CO6	3	2								3			3	3	

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Augmented Reality (AR): Definition and Scope, A Brief History of Augmented Reality, Examples, Related Fields, System Structure of Augmented Reality, Key Technology in AR.
Introduction to Virtual Reality (VR): Fundamental Concept and Components of VR, Primary Features and Present Development on VR.

Core Technologies of AR: Computer Vision: Image recognition, feature tracking, and environmental understanding, 3D Modeling, and Registration: Creating and integrating 3D objects in real-time, Display Technologies: Head-mounted displays (HMDs), AR glasses, and other interfaces, Tracking and Registration: Marker-based and markerless tracking techniques

Environment Modelling in VR: Geometric Modelling, Behaviour Simulation, Physically Based Simulation. Interactive Techniques in VR: Body Track, Hand Gesture, 3D Manus, Object Grasp.

Development Tools and Frameworks in AR/VR: Frameworks of Software Development Tools in AR/VR, X3D Standard, Vega, MultiGen, Virtools, and Unity.
Application of AR/VR in Digital Entertainment: AR/VR Technology in Film and TV Production, Healthcare, VR Technology in Physical Exercises and Games, Demonstration of Digital Entertainment by VR.

Self Learning Component: Unity 3D, Manus VR, Blender

TEXT BOOKS:

1. Dieter Schmaltzier and Tobias Hollerer,"Augmented Reality: Principles and Practice", Addison-Wesley, 2006.

2. Burdea, G. C. and P. Coffet. ,"Virtual Reality Technology", Second Edition. Wiley-IEEE Press, 2003/2006.

REFERENCE BOOKS

1. Sherman, William R. and Alan B. Craig, "Understanding Virtual Reality – Interface, Application, and Design", Morgan Kaufmann, 2002.

2. Fei GAO, "Design and Development of Virtual Reality Application System", Tsinghua Press, March 2012.

3. Guangran LIU, "Virtual Reality Technology", Tsinghua Press, Jan. 2011.

JOURNALS/MAGAZINES:

1. https://www.mdpi.com/journal/electronics/special_issues/VR_AR
2. https://www.sciencedaily.com/releases/2020/04/200420145025.htm
3. https://www.springer.com/journal/10055

SWAYAMNPTEL/MOOCs

1. https://nptel.ac.in/courses/106/106/106106182/
2. https://www.classcentral.com/course/augmented-reality-virtual-reality-mixed--10508
3. https://www.edx.org/learn/augmented-reality

Course Title	Data Communication and Systems				Course Type		OE	
Course Code	B24EF0612	Credits	3		Class		VI SEMSETER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course introduces the fundamental concepts of data communication and computer networks. Students will explore the underlying principles of data transmission, network protocols, network architectures, and various network devices. The course emphasizes understanding the technologies that enable communication between computers and devices across a network.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the fundamental concepts of data communication and networking.
2. Analyze the different network models (OSI and TCP/IP) and identify various network devices and their functionalities.
3. Understand the operation of data link layer protocols and Explain network layer routing algorithms and protocols.
4. Describe transport layer protocols and services and Analyze application layer protocols and their applications.

COURSE OUTCOMES (COs):

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Explain the fundamental concepts of data communication, network models, and network protocols.	1,2 and 9	1,2,3
CO2	Analyze the functionalities of various network devices and their role in data communication.	1,2 and 10	1,2,3
CO3	Describe the operation of data link layer protocols like error detection and correction techniques.	1,2 and 9	1,2,3
CO4	Explain network layer routing algorithms and protocols (e.g., static routing, dynamic routing).	1,2 and 9	1,2,3
CO5	Describe transport layer protocols (TCP, UDP) and their services and Analyze application layer protocols and their applications.	1,2 and 9	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		??				
CO2				??		
CO3		??				
CO4		??				
CO5		??		??		

COURSE ARTICULATIONMATRIX:

CO#/ Pos	PO	PO	PO 3	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3	3							3				3	1	1
CO2	2	2								2			3	1	1
CO3	2	2							2				1	3	1
CO4	2	2							2				3	1	2
CO5	2	2							2				2	1	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Data Communication and Networks: Basic Concepts: Define Data Communication, Networks, and Network Components (nodes, links, devices), Advantages of using Computer Networks (resource sharing, communication, cost savings), Different Network Types (LAN, WAN, MAN, PAN). Data and Signals, Digital Signals, Transmission Impairment.

Network Models (OSI and TCP/IP): TCP/IP Protocol Suite: Layered Architecture, Layers in TCP/IP suite, Description of layers, Encapsulation and Decapsulation, Addressing, Multiplexing and Demultiplexing, The OSI Model: OSI Versus TCP/IP.

Physical Layer and Data Link Layer: Transmission Media (twisted pair, coaxial cable, fiber optics), Different Transmission Techniques (analog vs. digital, baseband vs. broadband), Multiplexing and Spread Spectrum Nodes and Links, Data Link Control (DLC) services: Framing, Flow and Error Control, Data Link Layer Protocols: Simple Protocol, Stop and Wait protocol, Piggybacking.

Network Layer: Introduction, Network Layer services: Packetizing, Routing and Forwarding, Packet Switching, IPV4 Addresses: Classful Addressing, Classless Addressing, Internet Protocol (IP): Datagram Format, Fragmentation, Options, Security of IPv4 Datagrams. Routing Algorithms: Distance Vector Routing, Link State Routing.

Transport Layer and Application Layer: User Datagram Protocol: User Datagram, UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, TCP congestion Control. Standard Client -Server Protocols: World wide web, Hyper Text Transfer Protocol, FTP: Two connections, Control Connection, Data Connection.

Self-Learning Component:

1. Website Link "https://www.tutorialspoint.com/computer_fundamentals/computer_networking.htm"
2. Network Management Tools like GNS3, Cisco Packet Tracer

TEXT BOOKS:

1. "Computer Networks" by Andrew S. Tanenbaum
2. "Data Communications and Networking" by Behrouz A. Forouzan
3. "Data and Computer Communication" by William Stallings

REFERENCE BOOKS:

1. "Computer Networking: A Top-Down Approach" by James F. Kurose & Keith W. Ross

2. James J Kurose, Keith W Ross, " Computer Networks", Pearson Education.

3. Wayne Tomasi, "Introduction to Data Communication and Networking'', Pearson Education.

JOURNALS/MAGAZINES:

1. https://www.comsoc.org/publications/journals/ieee-tcom
2. https://link.springer.com/journal/10922
3. https://www.officialmediaguide.com/ie31/commmag/

SWAYAM NPTEL/MOOCs

1. **Fundamentals of Network Communication** by University of Colorado System (https://www.coursera.org/learn/fundamentals-network-communications)
2. Introduction to TCP/IP (https://www.coursera.org/learn/tcpip)
3. Communication Networks (https://onlinecourses.nptel.ac.in/noc24_ee71/preview)

Course Title	Generative Artificial Intelligence Lab				Course Type		HC	
Course Code	B24EH0605	Credits	1		Class		VI Semester	
Course Structure	LTP	Credits	Contact Hours	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2	0	28	25	25

COURSE OVERVIEW:

This lab course offers a hands-on exploration of Generative AI models, where machines learn to create entirely new data. You'll delve into core techniques like generative adversarial networks and variational auto encoders and use open-source generative models for text generation, image generation, text embeddings using Python programming.

COURSE OBJECTIVE (S):

The objective of this course are to

1. Evaluate the generative AI model capabilities and configurations.
2. Implement generative models like GANs and VAE from scratch.
3. Use opensource generative AI models for different use cases.
4. Choose the most appropriate models for a specific task.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Prompt Engineering: Construct and refine prompts for generative models.	1 to 5	1,2,3
CO2	Compare different LLMs for same prompting conditions.	1 to 5	1,2,3
CO3	Build and evaluate generative models like GANs and VAEs from scratch.	1 to 5	1,2,3
C04	Make use of opensource text generation models for a specific use case like sentiment analysis.	1 to 5	1,2,3
CO5	Make use of opensource vision models like stable diffusion.	1 to 5	1,2,3
CO6	Experiment with finetuned models like Llama2-Chat.	1 to 5	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply(L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1			✓			
CO2			✓	✓	✓	
CO3			✓	✓	✓	

C04			✓			
CO5			✓			
CO6			✓			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	3		3	3					3	3	2
CO2	1	2	3	2	3		3	3					3	3	2
CO3	3	3	3	3	3		3	3					3	3	2
CO4	3	3	3	3	3		3	3					3	3	2
CO5	3	3	3	3	3		3	3					3	3	2
CO6	3	3	3	2	3		3	3					3	3	2

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Exploring Google AI Studio and Open AI Playground - models, model configuration options, prompt types. (https://aistudio.google.com/, https://platform.openai.com/playground)	Google AI Studio & OpenAI Playground	Understanding the model types, their capabilities, configurations.
2	Construct and refine prompts to generate different creative text formats like code, scripts, or stories using ChatGPT/Google Gemini. Compare the performance and capabilities of both LLMs under the same prompting conditions.	ChatGPT/ Google Gemini/	Prompt Engineering
3	Implement few-shot prompting examples with ChatGPT or Gemini web interface. Compare the performance and capabilities of both LLMs under the same prompting conditions.	ChatGPT/ Google Gemini/	Prompt Engineering
4	Train a GAN to generate numbers with a specific pattern in Python	Python, Jupyter Notebook,	Data preprocessing and implementing GANs
5	Implement a GAN in Python for generating images of handwritten digits like MNIST data.	Python, Jupyter Notebook	Data preprocessing and implementing GANs
6	Implement a VAE to generate handwritten digits in Python	Python, Jupyter Notebook	Data preprocessing and implementing VAEs
7	Implement a VAE to generate human faces in Python	Python, Jupyter Notebook	Data preprocessing and implementing VAEs
8	Use an open-source text embedding model like Sentence Transformers to generate embeddings for a given sentence in Python.	Python, Jupyter Notebook, Hugging Face	Grasp the concept of text embeddings and use the model for generating them.
9	Use Bert Transformer for sentiment analysis in Python	Python, Jupyter Notebook, Hugging Face	Using Transformers from Python Library
10	Use an open-source text generation model GPT-2 from Hugging Face for text generation.	Python, Jupyter Notebook, Hugging Face	Use of opensource models in Python for text generation.

11	Use open-source stable diffusion for image generation in Python	Python, Jupyter Notebook, Hugging Face	Use of opensource models in Python for image generation
12	Implement text summarization using a finetuned open-source model in Python	Python, Jupyter Notebook, Hugging Face	Using a fine-tuned model.
13	Use Llama2 Chat (https://huggingface.co/meta-llama/Llama-2-7b-chat-hf) model for creating a chatbot in Python.	Python, Jupyter Notebook, Hugging Face	Using a task specific fine-tuned model
14	Use an open-source text to speech model to convert given text into speech in Python.	Python, Jupyter Notebook, Hugging Face	Using a text to speech model.

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Implement stable diffusion from scratch for image generation.	Python, Jupyter Notebook	Build models from scratch for a given use case
2	Build a Generative Pretrained Transformer from Scratch.	Python, Jupyter Notebook	Build and pretrain a model from scratch.

TEXTBOOKS:

1. Generative Deep Learning, Teaching Machines to Paint, Write, Compose and Play, David Foster, Oreilly, 2019.

2. Generative AI with Python and TensorFlow, Joseph Babcock, Raghav Bali, Packt, 2021.

REFERENCE BOOKS:

1. Introduction to Generative AI, Numa Dhamani, Maggie Engler, Manning Publications Co., 2024,

JOURNALS/MAGAZINES:

1. Generative Artificial Intelligence: Trends and Prospects, IEEE

2. Generative AI for Software Practitioners, IEEE

Course Title	Cloud Computing Tools Lab				Course Type		HC	
Course Code	B24EH0606	Credits	1		Class		VI SEMESTER	
Course Structure	LTP	Credits	Contact	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture			-				
	Theory	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2		28	25	25

COURSE OVERVIEW:

This course demonstrates the creation of a virtual machine and performs several operations. It also demonstrates the use of CloudSim. CloudSim is a simulation toolkit that supports the modeling and simulation of the core functionality of the cloud, like job/task queue, processing of events, creation of cloud entities(datacenter, datacenter brokers, etc.), communication between different entities, implementation of broker policies, etc.

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Demonstrate the creation of a Virtual Environment.
2. Illustrate the mechanism of cloning in a Virtual Environment.
3. Outline the different entities in a CloudSim.
4. Demonstrate the creation of a datacenter with one (or) two hosts with other entities in a CloudSim.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Recall the creation of a Virtual Environment on the hypervisor.	1,2,3,4,5,9,10 ,11,12	1,2,3
CO2	Explain the mechanism of creation of cloning and switch with multiple networks.	1,2,3,4,5,9,10 ,11,12	1,2,3
CO3	Apply the fundamental concepts of CloudSim.	1,2,3,4,5,9,10 ,11,12	1,2,3
C04	Summarize the different entities in a CloudSim.	1,2,3,4,5,9,10 ,11,12	1,2,3
CO5	Interpret the creation of a datacenter using the entities available in CloudSim.	1,2,3,4,5,9,10 ,11,12	1,2,3
CO6	Build the scalable and pause simulation in a CloudSim.	1,2,3,4,5,9,10 ,11,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	**Understand (L2)**	**Apply(L3)**	**Analyze(L4)**	**Evaluate(L5)**	**Create(L6)**
CO1	✓					
CO2		✓				
CO3			✓			
C04		✓				

CO5		✓				
CO6			✓			

COURSE ARTICULATION MATRIX

CO# /	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	3	3	2	2				1	1	2	3	3	3	2
CO2	3	3	3	2	2				1	1	2	3	3	3	2
CO3	3	3	3	3	3				1	1	2	3	3	3	3
CO4	3	3	3	3	3				1	1	2	3	3	3	3
CO5	3	3	3	3	3				1	1	2	3	3	3	3
CO6	3	3	3	2	3				1	1	2	3	3	3	3

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Demonstrate on the Virtual Environment on hypervisor. a) Communication between the VM's. b) The backup and restore mechanism.	Virtualization Technique	Understand virtualization technique
2	Demonstrate the following a) Mechanism of cloning b) Create a switch with multiple networks having the different VM's.	Virtualization Technique	Understand the use of cloning
3	Demonstrates how to simulate a Data Center with one host and run one Cloudlet on it using cloudsim	CloudSim	Understand the working of CloudSim with different entities
4	Demonstrates how to create a datacenter with one host, two virtual machines and run two cloudlets on it. Both virtual machine (vm1,vm2) has same machine configuration.	CloudSim	Understand the working of CloudSim with different entities
5.	Demonstrates how to create a datacenter with two hosts, two virtual machines and run two cloudlets on it. The cloudlets run in VMs with different MIPS requirements. The second VM will have twice the priority of virtual machine one(VM1) and so cloudlet will receive twice CPU time to complete the execution	CloudSim	Understand the working of CloudSim with different entities

6	Demonstrate how to create two datacenters with one host each and run two cloudlets on them.	CloudSim	Understand the working of CloudSim with different entities

7	Demonstrate how to create two datacenters with one host each and run cloudlets of two users on them.	CloudSim	Understand the working of CloudSim with different entities
8	Demonstrate how to create scalable simulations.	CloudSim	Understand the working of CloudSim with different entities
9	Demonstrate how to pause simulations.	CloudSim	Understand the working of CloudSim with different entities
10	Demonstrate how to add entities in run time	CloudSim	Understand the working of CloudSim with different entities

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Implement the service model of "Infrastructure as a Service" or private cloud datacenter. Use DevStack (Openstack) or Open Nebula or any other open-source framework for IaaS.	Openstack	Understand the open-source cloud platform for building and managing cloud computing platforms for public and private clouds.

TEXT BOOKS:

1. Dan C. Marinescu, Cloud Computing: Theory and Practice, MK, 2017.
2. Thomas Erl, Ricardo Puttini, Zaigham Mahmood Cloud Computing: Concepts, Technology & Architecture PHI, 2013.

REFERENCE BOOKS:

1. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing Principles and Paradigms, 1st Edition, Wiley, 2013.

JOURNALS/MAGAZINES:

1. https://www.researchgate.net/publication/258652616_OpenStack_Toward_an_Open-Source_Solution_for_Cloud_Computing
2. https://ijettjournal.org/archive/ijett-v8p216
3. https://www.researchgate.net/publication/349835588_CloudSim_303_Simulator_Step_by_Step
4. https://www.researchgate.net/publication/339935689_Cloud_Computing_Modeling_and_Simulation_using_CloudSim_Environment

5. https://pdfcoffee.com/cloudsim-303-manual-pdf-free.html

Course Title	Speech and Language Processing Lab				Course Type		HC	
Course Code	B24EH0607	Credits	3		Class		VI Semester	
Course Structure	LTP	Credits	Contact	Workload	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practice	1	2	2				
	Total	1	2	2	0	28	25	25

COURSE OVERVIEW:

The Speech and Language Processing Lab offers students hands-on experience with implementing and experimenting with algorithms and techniques in the field of speech and language processing. Through a series of structured experiments and projects, students will gain practical skills in analyzing, processing, and synthesizing both speech and text data. The lab activities are designed to complement the theoretical concepts covered in the associated lecture course, providing students with opportunities for deeper exploration and application of the material.

COURSE OBJECTIVE (S):

The objective this course are to

1. Implement algorithms for various speech and language processing tasks, such as text processing, speech synthesis, and natural language understanding.
2. Utilize programming languages, tools, and libraries commonly employed in speech and language processing, such as Python, NLTK, and HTK.
3. Apply critical thinking and problem-solving skills to tackle real-world speech and language processing challenges.
4. Categorize and compare different techniques of text processing and speech synthesis used in various applications.
5. Engage in hands-on projects to explore and discover the challenges and applications of speech and language processing in real-world scenarios.
6. Stay updated with current research trends and applications in the field of speech and language processing.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	To discuss implementation of algorithms for speech and language processing tasks.	1,2,3,5,12	1,2,3
CO2	To prepare skills in programming and using tools and libraries commonly employed in speech and language processing.	1,2,3,5,12	1,2,3
CO3	To construct critical thinking and problem-solving abilities in the context of speech and language processing tasks.	1,2,3,5,12	1,2,3
C04	To categorize techniques of text processing and speech synthesis in real world applications.	1,2,3,5,12	1,2,3
CO5	To discover real-world applications and challenges in speech and language processing through hands-on projects.	1,2,3,5,12	1,2,3
CO6	To explore current research trends and applications in the field.	1,2,3,5,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply(L3)	Analyze(L4)	Evaluate(L5)	Create(L6)
CO1		√				
CO2			√			
CO3			√			
C04				√		
CO5				√		
CO6				√		

COURSE ARTICULATION MATRIX

CO# /	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	2	2		3							2	3	3	3
CO2	3	2	2		3							2	2	2	2
CO3	3	2	2		3							2	2	2	2
CO4	2	3	2		3							2	2	2	2
CO5	3	3	3		3							3	3	3	3
CO6	3	3	3		3							3	3	3	3

Practice:

No	Title of the Experiment	Tools and Techniques	Expected Skill/Ability
1	Demonstrate noise removal for any textual data and remove regular expression pattern such as hash tags from textual data.	NLTK	Python
2	Perform lemmatization and stemming using python library NLTK.	NLTK	Python
3	Demonstrate object standardization such as replace social media slangs from a text.	NLTK	Python
4	Perform part of speech (POS) tagging on any textual data.	NLTK	Python
5	Demonstrate Term Frequency- Inverse Document Frequency using python.	TF-IDF	Python
6	Demonstrate word embedding's using word2vec.	Word2vec	Python
7	Illustrate automating the task of speech signal-to-symbol transformation, by manually segmenting a given speech signal into its constituent symbols.	Python	Python

8	Express the characteristics of speech waveform in terms of the production characteristics by identifying/locating voiced/unvoiced/plosive/silence regions, and providing acoustic phonetic description of the regions.	Python	Python
9	Study the effects of sampling (aliasing) and quantization on speech signals by playing them at different sampling rates and bits per sample (up to 16 bps).	Python, Kaldi, Julius, openSMILE	Python
10	Study the effect of size of the analysis window (less than one pitch period, one pitch period, two to four pitch periods) and study the effect of shape of the analysis window (rectangular, Hamming and Hanning window functions).	Python, Kaldi, Julius, openSMILE	Python
11	Study the characteristics of speech using linear prediction (LP) analysis.	Python, Kaldi, Julius, openSMILE	Python
12	Break down the speech signal into different components such as the excitation source information (pitch and gain) and vocal tract system information (formants) (Using analysis by synthesis).	Python, Kaldi, Julius, openSMILE	Python

Project			
No	**Title of the Experiment**	**Tools and Technics**	**Expected Skill/Ability**
1	Explore techniques for natural language understanding and response generation.	NLTK	Python
2	Integrate speech recognition and synthesis modules to create a spoken dialogue system.	Python, Kaldi, Julius, openSMILE	Python

TEXT BOOKS:

1. Speech and Language Processing - An Introduction to Natural Language Processing, Competitional Linguistics and Speech Recognition by Daniel Jurafsky and James H Martin, Pearson Education.
2. Speech and Audio Signal processing: Processing and Perception of Speech and Music by Ben Gold and Nelson Morgan, Wiley.

REFERENCE BOOKS:

1. Natural Language Processing with Python by Steven Bird, Ewan Klein and Edward Loper, First Edition, OReilly Media, 2009.

JOURNALS/MAGAZINES:

1. IEEE/ACM Transactions on Audio, Speech, and Language Processing.

Course Title	Mini Project				Course Type		HC	
Course Code	B24EH0608	Credits	2		Class		VI Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	2	4	4				
	Total	2	4	4	-	56	50	50

COURSE OVERVIEW

The course overview is designed to guide participants through the comprehensive landscape of project management, focusing on developing the necessary skills and knowledge to effectively manage complex projects. The curriculum emphasizes practical application, sustainable development, and ethical responsibilities in project execution.

COURSE OBJECTIVES

The objectives of this course are to

1. Demonstrate comprehensive understanding and expertise in the selected project topic, ensuring a solid foundation for further exploration and application.
2. Identify, analyze, and articulate complex problems relevant to the project, formulating substantiated conclusions that guide the development of effective solutions.
3. Design innovative and practical solutions tailored to the identified project problem, incorporating theoretical and empirical insights.
4. Undertake thorough investigations into the project problem, employing appropriate methodologies to derive valid conclusions and insights.
5. Utilize modern engineering techniques, resources, and tools effectively throughout the project lifecycle to enhance project outcomes.
6. Apply project results in a manner that promotes sustainable development, considering social, economic, and environmental impacts.
7. Understand and evaluate the implications of project outcomes in the context of environmental sustainability, ensuring responsible project execution.
8. Recognize and uphold professional and ethical responsibilities during project execution, fostering integrity and accountability.
9. Function effectively both as an individual contributor and as a collaborative team member, enhancing project dynamics and outcomes.
10. Develop and refine oral and written communication skills for effective presentation and reporting of project findings.
11. Demonstrate knowledge and understanding of cost estimation and time management techniques necessary for successful project execution.
12. Engage in continuous learning and professional development to enhance knowledge and competencies in the chosen project area, adapting to evolving industry standards and practices.

COURSE OUTCOMES (CO'S):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project work to attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing the project work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysis required for carrying out the project.	11	1,2,3
CO12	Engage in lifelong learning to improve knowledge and competence in the chosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3
CO12												3	3	3	3

Note: 1-Low, 2-Medium, 3-High

The students are informed to follow the following instructions to complete the Skill development-2: Students will be offered training and certifications on the trending technologies from the industry experts.The students are evaluated and certified after the training programs.

<table>
<tr><td>Course Title</td><td colspan="4">AEC-4(Ability Enhancement Course Placement Training)</td><td colspan="2">Course Type</td><td colspan="2">AEC</td></tr>
<tr><td>Course Code</td><td>B24EF0609</td><td>Credit</td><td colspan="2">1</td><td colspan="2">Class</td><td colspan="2">VI SEMESTER</td></tr>
<tr><td rowspan="5">Course Structure</td><td>LTP</td><td>Credit</td><td>Contact Hours</td><td>Work Load</td><td colspan="2" rowspan="2">Total Number of Classes Per Semester</td><td colspan="2" rowspan="2">Assessment Weightage</td></tr>
<tr><td>Lecture</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Tutorial</td><td>-</td><td>-</td><td>-</td><td rowspan="2">Theory</td><td rowspan="2">Practical</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td></tr>
<tr><td>Practical</td><td>1</td><td>2</td><td>2</td></tr>
<tr><td>Total</td><td>1</td><td>2</td><td>2</td><td>-</td><td>28</td><td>25</td><td>25</td></tr>
</table>

COURSE OVERVIEW

Placement training classes can effectively be conducted as a part of an Ability Enhancement Course (AEC). These courses are designed to enhance students' skills, making them more prepared and competitive for the job market.

VII SEMESTER

VII Semester												
Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0701	Multi Large Language Models	HC	3	0	0	3	3	50	50	100	PCC
2	B24EHS71X	Professional Elective-6	SC	3	0	0	3	3	50	50	100	PEC
3	B24EH0702	Capstone Project Phase-1	HC	0	0	2	2	4	50	50	100	Proj
4	B24EH0703	Internship / Global Certification	HC	0	0	2	2	4	50	50	100	Intern
TOTAL				6	0	4	10	14	200	200	400	
TOTAL SEMESTER CREDITS				10								
TOTAL CUMULATIVE CREDITS				145								
TOTAL CONTACT HOURS				14								
TOTAL MARKS				400								

Course Title	Multi Large Language Models				Course Type		HC	
Course Code	B24EH0701	Credits	3		Class		VII SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:

This course provides a comprehensive introduction to Multi-Large Language Models (MLLMs), a rapidly advancing field in artificial intelligence that has revolutionized various domains, including natural language processing, computer vision, and multimodal learning. Through a combination of theoretical concepts and practical applications, students will gain a deep understanding of MLLM principles, architectures, and their real- world implications

COURSE OBJECTIVE (S):

The objectives of this course are to

1. Gain a comprehensive understanding of Multi Large Language Models (MLLMs), including their principles, architectures, and applications.
2. Explore various MLLM architectures, particularly transformer-based models, and understand their components.
3. Study practical applications of MLLMs across different domains such as natural language processing (NLP), computer vision (CV), and multimodal learning involving image and video data.
4. Develop skills to analyze, implement, and evaluate MLLMs using state-of-the-art techniques and tools.
5. Prepare for careers in research, industry, or further study in fields related to artificial intelligence, machine learning, and data science.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	describe the evolution of language models from rule-based systems to modern Multi-Large Language Models (LLMs) like GPT, BERT, and T5	**1,2,12**	1,2
CO2	Apply skills to implement and train LLMs, including data preprocessing, training techniques, and efficient handling of large-scale data	**1,5,12**	1,2
CO3	Integrate various techniques multimodal information into LLMs, exploring vision-language and video-language models	**1,2,3,4**	1,2
CO4	Explore practical applications of LLMs in natural language understanding, machine translation, text generation, and sentiment analysis	**1,2,5**	1,2
CO5	Evaluate the ethical implications of LLMs, considering issues of bias, fairness, privacy, and security	**2,6,7,8**	1,2
CO6	Analyze future trends and challenges in LLM development, including scalability, efficiency, and addressing ethical concerns	**2,4,7,12**	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3			√	√		
CO4				√		
CO5					√	
CO6				√		

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	3	2										2	3	3	2
CO2	3				3							2	3	2	3
CO3	3	3	2		3								3	3	3
CO4	3	3			3								3	3	2
CO5		2				3	3	2					3	2	3
CO6		3		4			2					2	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Multi-Large Language Models: History and Evolution of Language Models, Defining LLMs, Brief history of LLMs, modern LLMs, impact of LLMs From Rule-based Systems to Deep Learning, **Multi-Large Language Models:** Definition and Importance, Scope and Potential Applications, Strengths and limitations of LLMs.

Rise of Large Language Models: Development of Transformer Models, Key Models: GPT, BERT, T5,

Mathematical and Statistical Principles: Probability Theory and Statistics in NLP, Neural Networks and Deep Learning Basics

Algorithms and Techniques: Attention Mechanism, Transformers and Variants, Transfer Learning, Zero-shot and Few-shot Learning

Architectures of Multi-Large Language Models:
Model Architectures: Transformer Architecture, Variants and Innovations (e.g., GPT-3, BERT, T5), **Parallelismand Distribution:** Data Parallelism and Model Parallelism, Distributed Training Techniques,

Inter-model Communication and Coordination: Ensemble Methods, Model Integration Techniques

Training Multi-Large Language Models:
Data Preparation and Preprocessing: Data Collection and Cleaning, Handling Imbalanced Data
Training Techniques and Optimization: Gradient Descent Variants, Hyperparameter Tuning and Optimization

Handling Large-scale Data: Efficient Data Management, Memory and Computational Efficiency

Multimodal Large Language Models: Techniques: **Concatenation-based Methods**: **Attention Mechanisms**, **Transformers**

Vision-Language Models: CLIP (Contrastive Language-Image Pre-training), ViLT (Vision-and-Language Transformer), DALL-E

Video-Language Models: Techniques: **Spatiotemporal Attention, Video Captioning, Video Question Answering.**

Applications and Use Cases:

Natural Language Understanding: Sentiment Analysis, Named Entity Recognition,

Machine Translation: Advances in Neural Machine Translation, Cross-lingual Models,

Text Generation and Summarization: Generative Pre-trained Models, Techniques for Summarization

Challenges and Future Directions: Ethical Considerations, Addressing Bias and Fairness ,Privacy and Security Concerns, **Scalability and Efficiency:** Computational and Energy Efficiency, Scalability Solutions, **Future Trends in Multi-Large Language Models**

Self Learning Component:

1. Identify and discuss a real-world application of multi-large language models
2. personal learning plan based on identified resources for continued study
3. Collect and preprocess a dataset for training a language model
4. Implement a simple transformer model using a deep learning framework
5. **Software Frameworks and Libraries:** Frameworks (e.g., TensorFlow, PyTorch), Specialized NLP Libraries,

TEXT BOOKS:

1. Multimodal Large Language Models: Principles, Architectures, and Case Studies, Tech Innovators Publishing India Pvt. Ltd, **ISBN**: 978-1-23456-789-0

2. Designing Large Language Model Applications, Publisher(s): O'Reilly Media, Inc. ISBN: 9781098150501

REFERENCE BOOKS:

1. Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play" by David Foster
2. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
3. "Natural Language Processing with Python Cookbook" by Krishna Bhavsar and Naresh Kumar

JOURNALS/MAGAZINES:

1. International Journal of Computer Vision (IJCV)
2. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)
3. Computer Vision and Image Understanding (CVIU)
4. **Online Courses, Tutorials, and Workshops: Coursera,** Google AI, TensorFlow, Op

Professional Elective-6

Course Title	Augmented reality and Virtual reality				Course Type		SC	
Course Code	B24EHS711	Credits	3		Class		VII Semester	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

This course covers basic concepts of augmented reality and virtual reality. The course also introduces the student to the working of multiple models of input and output interface in VR. The course also helps the student to understand development tools and frameworks in VR. Further, this course helps the student to work on the application of VR in digital entertainment

COURSE OBJECTIVE (S):

The objectives of this course are to:

1. Explain the principles and multidisciplinary features of Virtual Reality.
2. Illustrate the multimodal user interaction and perception in Virtual Reality.
3. Demonstrate the use of objects for managing large scale Virtual Reality environment in real time.
4. Discuss the various solutions using Virtual Reality system framework and development tools for industry and social relevant applications.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Identify the components of Augmented and Virtual Reality	1 ,10 1	1
CO2	Apply multimodal user interaction and perception techniques involved in Virtual Reality.	1 to 3,5,10	3
CO3	Develop real world applications using Simulation and Interactive techniques.	1 to 3, 10	2,3
CO4	Choose the innovative Virtual Reality solutions for industrial and social relevant applications.	1,10	1
CO5	Make use of unity3D to develop innovative project.	1 to 3, 10	2,3
CO6	Apply VR modeling techniques to solve real world applications.	1,2,10	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1			√			
CO2			√			
CO3			√			
CO4			√			
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO	PSO	PSO
CO1	3									3			3		
CO2	3	3	3		3					3					3
CO3	3	3	3							3				3	3
CO4	3									3			3		
CO5	3	2	2							3				3	3
CO6	3	2								3			3	3	

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Augmented Reality (AR): Definition and Scope, A Brief History of Augmented Reality, Examples, Related Fields, System Structure of Augmented Reality, Key Technology in AR.
Introduction to Virtual Reality (VR): Fundamental Concept and Components of VR, Primary Features and Present Development on VR.

Multiple Models of Input and Output Interface in VR: Input – Tracker, Sensor, Digital Glove, Movement Capture, Video-based Input, 3D Menus and 3D Scanner. Output – Visual, Auditory, Haptic Devices.

Environment Modelling in VR: Geometric Modelling, Behaviour Simulation, Physically Based Simulation. Interactive Techniques in VR: Body Track, Hand Gesture, 3D Manus, Object Grasp.
Development Tools and Frameworks in VR: Frameworks of Software Development Tools in VR, X3D Standard, Vega, MultiGen, Virtools, and Unity. Application of VR in Digital Entertainment: VR Technology in Film and TV Production, VR Technology in Physical Exercises and Games, Demonstration of Digital Entertainment by VR.

Introduction to unity: Installation of unity, setting up Unity for VR development, working on setting up the environment on unity using objects

Self-Learning Component:

1. Unity 3D, Manus VR, Blender

2. Do mini project by group of 3 or 4.

TEXT BOOKS:

1. Dieter Schmaltzier and Tobias Hollerer,"Augmented Reality: Principles and Practice", Addison-Wesley, 2006.

2. Burdea, G. C. and P. Coffet. ,"Virtual Reality Technology", Second Edition. Wiley-IEEE Press, 2003/2006.

REFERENCE BOOKS

1. Sherman, William R. and Alan B. Craig, "Understanding Virtual Reality – Interface, Application, and Design", Morgan Kaufmann, 2002.

2. Fei GAO, "Design and Development of Virtual Reality Application System", Tsinghua Press, March 2012.

3. Guangran LIU, “Virtual Reality Technology”, Tsinghua Press, Jan. 2011.

E-RESOURCES:

1. VIRTUAL REALITY (https://lavalle.pl/vr/)

JOURNALS/MAGAZINES:

1. **https://www.mdpi.com/journal/electronics/special_issues/VR_AR**
2. **https://www.sciencedaily.com/releases/2020/04/200420145025.htm**
3. **https://www.springer.com/journal/10055**

SWAYAMNPTEL/MOOCs

1. **https://nptel.ac.in/courses/106/106/106106182/**
2. **https://www.classcentral.com/course/augmented-reality-virtual-reality-mixed--10508**
3. **https://www.edx.org/learn/augmented-reality**

Course Title	Industrial Computing				Course Type		SC	
Course Code	B24EHS712	Credits	3		Class		VII SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42	-	50	50

COURSE OVERVIEW:

This course offers a comprehensive exploration of industrial computing with artificial intelligence (AI) within the context of industrial systems. Students will delve into the foundational principles of industrial computing, production economics, and AI, while also gaining practical insights into knowledge representation, reasoning, expert systems, and the application of AI in industrial settings. Through lectures, hands-on exercises, and case studies, students will develop the knowledge and skills necessary to leverage computing technologies and AI to optimize industrial processes and drive innovation.

COURSE OBJECTIVE (S):

The objective of this course are to

1. To identify potential areas for computing and justify need for industrial computing
2. Study the evolution of Robotics in Industrial computing.
3. Learn the methods of solving problems using Artificial Intelligence.
4. Apply the concepts of AI to attain industrial computing

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Proficiency in AI Integration**:** Students will integrate AI technologies into industrial computing systems	1 to 5,9 10,12	1,2,3
CO2	Optimization of Industrial Processes**:** Students will analyze industrial processes and use AI techniques to optimize efficiency, and enhance productivity in industrial robotics.	1 to 5,9 10,12	1,2,3
CO3	Real-time Monitoring and Control**:** Students will learn to design and implement AI-driven systems for monitoring and controlling industrial operations in real-time.	1 to 5,9 10,12	1,2,3
CO4	Predictive Maintenance: Students will learn to use AI techniques like predictive analytics and machine learning for predictive maintenance in industries.	1 to 5,9 10,12	1,2,3
CO5	Safety and Compliance: Students will learn how AI improves safety and compliance in industries.	1 to 5,9 10,12	1,2,3
CO6	Data-driven Decision Making: Students will learn to use data analytics and AI tools for making informed decisions in industries.	1 to 5,9 10,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	halyze (L4)	aluate (L5)	reate (L6)
CO1		√				
CO2			√			
CO3			√			
CO4		√				
CO5			√			
CO6			√			

COURSE ARTICULATIONMATRIX:

CO #/ POs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	3	3	2	2	2				3	3		1	3	3	
CO 2	3	3	2	3	1				3	3		1	3	3	
CO 3	3	3	2	2	2				3	3		1	3	3	3
CO 4	3	3	2	3	1				3	3		1	3	3	
CO 5	3	3	2	2	2				3	3		1	3	3	3
CO 6	3	3	2	3	1				3	3		1	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Industrial Computing - Computing in Production System, Principles and Strategies of Computing, Basic Elements of an Computing System, Advanced Computing Functions, Levels of Computing.

Evolution of robotics. Robot anatomy- Co-ordinate Systems, Work envelope, types and classification – Specifications – Pitch, Yaw, Roll, Joint Notations, Speed of Motion, Pay Load – Basic robot motions – Point to point control, Continuous path control. Robot Parts and Their Functions – Need for Robots Different Applications. Robot drive systems: Pneumatic Drives – Hydraulic Drives – Mechanical Drives – Electrical Drives – D.C. Servo Motors, Stepper Motor, A.C. Servo Motors – Salient Features, Applications End Effectors – Grippers – Mechanical Grippers, Pneumatic and Hydraulic Grippers, Magnetic Grippers, Vacuum Grippers; Two Fingered and Three Fingered Grippers; Internal Grippers and External Grippers; Selection and Design Considerations

Knowledge Representation and Reasoning -Ontologies-foundations of knowledge representation and reasoning-representing and reasoning about objects-relations- event sactions- time- and space- predicate logic- situation calculus-description logics-reasoning with defaults,-reasoning about knowledge-sample applications- Representing Knowledge and reasoning in an Uncertain Domain- Bayes rule-bayesian networks-probabilistic inferences ample applications- Planning: planning as search- partial order planning- construction and use of planning graphs.

Expert systems -Expert systems – Architecture of expert systems, Roles of expert systems – Knowledge Acquisition – Meta knowledge, Heuristics. Typical expert systems – MYCIN, DART, XOON

Industrial AI applications and Case studies - Applications of Industrial AI in Monitoring, optimization and control. AI applications in Industry Automation using -natural language processing-computer vision-speech recognition-computer vision

Self Learning Component:
AI Basic concepts, Real-time computing systems used in industrial applications, including real-time operating systems (RTOS), scheduling algorithms, and techniques for ensuring timely and predictable response times.

TEXT BOOKS:

1. Rich and Knight, "Artificial Intelligence", 3rd Edition, Tata McGraw Hill, 2014.

2. M.P.Groover,"Automation, Production Systems and Computer Integrated Manufacturing", 5th edition, Pearson Education, 2009.
3. Fu, K.S., Gonzalez, R.C. and Lee C.S.G. – 'Robotics: Control, Sensing, Vision, and Intelligence' – McGraw Hill, New York, NY – 1987

REFERENCE BOOKS:

1. Anuradha Srinivasaraghavan, Vincy Joseph "Machine Learning", Wiley, 2019
2. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", 2nd Edition, Prentice Hall, 2003.
3. Rajiv Chopra, "Deep Learning", 1st edition, Khanna Publishing House,2018.
4. R K Mittal and I J Nagrath, 'Robotics and Control', Tata McGraw-Hill Publishing Company Limited, New Delhi, 2003.
5. J Craig, 'Introduction to Robotics: Mechanics and Control', Addison-Wesley, Reading, MA, 1989 (second edition).

JOURNALS/MAGAZINES:

1. ACM Computers in Industry
2. ACM Transactions on Design Automation of Electronic Systems

SWAYAMNPTEL/MOOCs

1. NPTEL course on "Applied Accelerated Artificial Intelligence"

   //onlinecourses.nptel.ac.in/noc23_cs104/preview
2. Udemy course on "Industrial cloud and Distributed cloud services"

   //www.udemy.com/course/industrial-cloud-distributed-cloud-services/?couponCode=OF52424

Course Title	Autonomous Drone				Course Type		SC	
Course Code	B24EHS713	Credits	3		Class		VII SEMESTER	
	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	0	0	0				
	Practice	-	-	-	Theory	Practical	CIE	SEE
	Total	3	3	3	42	0	50	50

COURSE OVERVIEW:

Autonomous Drone Systems is an advanced course designed to provide students with a comprehensive understanding of the technologies, algorithms, and systems involved in the development and operation of autonomous drones. The course will cover both theoretical foundations and practical applications, including drone design, sensor integration, flight control, navigation, and autonomous decision-making.

COURSE OBJECTIVE (S):

This objectives of the course are to

1. Integrate various sensors and instrumentation used in autonomous drones.
2. Develop algorithms for flight control and stability.
3. Implement navigation systems using GPS and computer vision.
4. Understand the principles of drone mechanics and aerodynamics.
5. Design and test autonomous decision-making algorithms for real-world scenarios.
6. Analyze the ethical, legal, and safety considerations in the deployment of autonomous drones.

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Compare the different types of drones and their specific applications.	1,2,3,12	1,2,3
CO2	Identify the essential components required for designing a drone airframe and propulsion system.	1,2,3,12	1,2,3
CO3	Design a motor and ESC configuration for a specific drone application, ensuring optimal performance.	1,2,3,12	1,2,3
CO4	Investigate the factors that affect drone flight performance and efficiency.	1,2,3,12	1,2,3
CO5	• Execute accurate and reliable navigation plans for autonomous drone operation in various environments.	1,2,3,12	1,2,3
CO6	• Implement obstacle avoidance and autonomous decision-making algorithms using machine learning techniques.	1,2,3,12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1			√	√		
CO2		√		√		
CO3			√			
CO4		√	√			

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	PO 3	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	2	3	3									2	3	3	3
CO2	3	3	3									2	3	3	3
CO3	3	3	3												
CO4	3	3	3										3	3	3
CO5	3	3	3										3	3	3
CO6	3	3	3									2	3	3	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to Drones

Definition and history of drones, Types of drones and their applications, Drone components and terminology, Regulations and Guidelines for drone usage

Drone Design and Assembly

Design considerations for drone airframe and propulsion systems, Selecting and assembling drone components such as motors, batteries, flight controllers, and cameras, Basic wiring and soldering techniques.

Drone Motors and ESC

Working, Types: Brush and Brushless Motors, motor sizing and identification, mounting patterns andthread size, Thrust to Weight ratio, KV ratings, advanced motor selection, Electronic Speed Controller(ESC).

Flight Mechanics and Dynamics

Basic principles of flight mechanics, flight controller board, Selection of drone controller with example, Factors affecting drone flight performance and efficiency.

Self Learning Component:

1. **X4 and ArduPilot:** Open-source autopilot software for drones, providing platforms to experiment with flight control and navigation algorithms.
2. **Gazebo and ROS (Robot Operating System):** Simulation tools that allow for testing and development in a virtual environment.
3. **QGroundControl:** Ground control station software for configuring and operating drones.

TEXT BOOKS:

1. M. LaFay, Building Drones for Dummies, John Wiley & Sons, Inc.,

2. E. Tooley, Practical Drones: Building, Programming, and Applications, Apress, 2021.

3. D. Levy, Drone Programming: AGuide to Code Your Own Drones, Packt Publishing, n.d.

4. S. K. Kopparthy, Drone Technology: Theory and Practice, Springer, 2020.

5. P. Horowitz and W. Hill, The Art of Electronics, Cambridge University Press, 2015.

6. K. Sundar and R. V. Rajakumar, Multicopters: Principles and Applications, Springer, 2021.

References:

1. D. Saxby, Drone Aerial Photography and Video: Techniques and Stories from the Field, CengageLearning, 2018.

2. D. McLeod, Getting Started with Drone: How to Build, Fly, and Program Your Own Drone,Apress, 2019.

3. M. A. Banks, Building and Flying Electric Model Aircraft, O'Reilly Media, Inc., 2014.

4. G. C. Camara Leal, Flying Robots: An Introduction to Autonomous Aerospace Systems, Springer,2017.

JOURNALS/MAGAZINES:

1. IEEE Transactions on Internet of Things
2. Springer Journal of Internet of Things
3. Elsevier Journal of Internet of Things

SWAYAMNPTEL/MOOCs

1. https://www.udemy.com/ Internet of Things
2. https://www.coursera.org/learn/ Internet of Things

Course Title	Capstone Project Phase-1				Course Type		HC	
Course Code	B24EH0702	Credits	2		Class		VII Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes Per Semester		Assessment in Weightage	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	2	4	4				
	Total	2	4	4	-	56	50	50

COURSE OVERVIEW:

The major project is a two semester-long practical project with the main objective that students show their ability to apply theoretical concepts learned in lectures to solve (complex) practical problems. The results are to be presented in a project report and as an oral presentation. The major project must be completed as a teamproject. Team projects are limited to a minimum of two students to a maximum number of four students.

COURSE OBJECTIVE (S):

The objectives of this course are to

1. To allow students to demonstrate a wide range of the skills learned during their course of study byasking them to deliver a product that has passed through the design, analysis, testing and evaluation.
2. To encourage multidisciplinary research through the integration learned in a number of courses.
3. To allow students to develop problem solving, analysis, synthesis and evaluation skills.
4. To encourage teamwork.
5. To improve students' communication skills by asking them to produce both a professional report andto give an oral presentation.

COURSE OUTCOMES (CO'S):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project work toattain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering toolsnecessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmentalsustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing theproject work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing andpresenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysisrequired for carrying out the project.	11	1,2,3

O12	Engage in lifelong learning to improve knowledge and competence in thechosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

	Blooms Level					
	Remember	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Create**
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3

CO11											**3**		**3**	**3**	**3**
CO12												**3**	**3**	**3**	**3**

Note: 1-Low, 2-Medium, 3-High

The students are informed to follow the following instructions to complete the Project Phase-1:

- Student should carry out project work in a group which is formed in the VII semester. Student mustselect group members from the same section as they belong to and select a faculty member from department of CSE as aninternal project guide based on research domain and expertise. Student may optionally also select external guide bearing domain expertise from different departments within University and Industry to carry out multidisciplinary project.
- Student group must propose a project title, after consultation with guides and after carrying out a literature survey. The proposed title must be submitted in form a document (synopsis) that contains the proposed title of the project, an abstract, Introduction, Survey, Feasibility, and cost estimation tocarry out the project.
- Further with the help of respective guide, each student group have to the literature review based onthe literature survey, identify the research gaps in the selected research/project domain, and then finalize the problem statement and objectives for the project.
- Each student groups shall be reviewed and evaluated in two reviews through the semester.
- Review 1 shall be on the presentation of the synopsis and justification of the title and feasibility of the project
- Review 2 shall be on the presentation on the literature survey carried out.

Finally, the Project Phase-1 shall conclude with each project group apply for idea patent or copyright andpublish a survey paper in SCOPUS indexed journals, write research proposals for fundings from various governmental organizations or industries.

Course Title	Internship/Global Certification				Course Type		HC	
Course Code	B24EH0703	Credits	2		Class		VII Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes		Assessment in	
	Lecture							
	Tutorial				Theory	Practical	CIE	SEE
	Practical	2	4	4				
	Total	2	4	4		56	50	50

COURSE OVERVIEW:

An internship can present students with new skills and opportunities. Interns not only gain technical knowledge within the industry of their choice, but they also learn how to interact with professionals in a workplace setting, and develop essential soft skills like time management, organization, adaptability, problem-solving and teamwork.

COURSE OBJECTIVES:

The objectives of this course are to

1. To allow students to develop problem solving, analysis, synthesis and evaluation skills.
2. To encourage team work.
3. To help students to gain exposure into industries.
4. To improve students' communication skills by asking them to produce both a professional report andto give an oral presentation.

COURSE OUTCOMES:

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project workto attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing theproject work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysisrequired for carrying out the project.	11	1,2,3
O12	Engage in lifelong learning to improve knowledge and competence in thechosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3
CO12												3	3	3	3

Note: 1-Low,2-Medium,3-High

The students are informed to follow the following instructions to complete the Internship:

1. The internship should be paid internship in IT industry.
2. The internship should be for minimum of three months.
3. The project title must be submitted in form a document (synopsis) that contains the proposed titleof the project, an abstract, Introduction and their roles and responsibilities in company.
4. Each student shall be reviewed and evaluated in two reviews through the semester.
5. Review 1 shall be on the presentation of the synopsis.
6. Review 2 shall be on the presentation on the roles and responsibilities carried out with modulecompletion results (as applicable)

VIII SEMESTER

VIII Semester												
Sl. No	Course Code	Title of the Course	HC/FC/ SC/OE /MC	Credit Pattern				Contact Hours/ Week	Examination			AICTE Course category
				L	T	P	Total		CIE Marks	SEE Marks	Total Marks	
1	B24EH0801	AI in Cybersecurity	HC	3	0	0	3	3	50	50	100	PCC
2	B24EH0802	Capstone project phase-II	Proj	0	0	10	10	20	50	50	100	Proj
3	B24EH0803	Internship /Global Certification	HC	0	0	2	2	4	50	50	100	Intern
TOTAL				3	0	12	15	27	150	150	300	
TOTAL SEMESTER CREDITS				15								
TOTAL CUMULATIVE CREDITS				160								
TOTAL CONTACT HOURS				27								
TOTAL MARKS				300								

Course Title	AI in Cyber Security				Course Type		HC	
Course Code	B24EH0801	Credits	3		Class		VIII SEMESTER	
	LTP	Credits	ontact Hours	Work Load	tal Number of Classes Per Semester		Assessment Weightage	
	Lecture	3	3	3				
	Tutorial	-	-	-				
	Practice	-	-	-	Theory	Practica l	CIE	SEE
	Total	3	3	3	42		50	50

COURSE OVERVIEW:
This course provides a comprehensive introduction to the application of Artificial Intelligence (AI) in cybersecurity, equipping students with the knowledge and skills necessary to tackle the evolving challenges in this critical domain. Through a combination of theoretical concepts and practical applications, students will explore the fundamental principles of AI and its role in enhancing cybersecurity measures.

COURSE OBJECTIVE (S):
The objective of this course are to

1. To learn the basic concepts of Artificial Intelligence in cybersecurity.
2. Introduction to basics of Malware Threat Detection and Network Anomaly Detection.
3. To have an overview of the cyber security for Mobile Devices, Digital Payments, Email, Web and Wireless networks
4. To Securing User Authentication abuse prevention, Fraud Prevention with Cloud AI Solutions
5. Introducing fraud detection algorithms
6. To study the defensive Attacks and Defenses GANs in a nutshell, Evaluating Algorithms Best practices of feature engineering
7. Assessing the Artificial Arsenal Evading Machine Learning detectors

COURSE OUTCOMES (COs)
After the completion of the course, the student will be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Show the ways in which AI may be used in a variety of cybersecurity contexts, such as threat identification and prevention.	1, 2, 3, 4, 5, 6	1, 2, 3
CO2	Use AI approaches to efficiently discriminate between spam and ham when d etecting and classifying email-based cybersecurity concerns.	1, 2, 3, 4, 5, 6	1, 2, 3
CO3	Recognise the underlying approaches and techniques and analyse and use AI-based solutions for identifying malware threats and network anomalies.	1, 2, 3, 4, 5, 6, 8	1, 2, 3
CO4	By utilizing a variety of algorithms and cloud based solutions, AI may be used to improve user authentication procedures a nd prevent fraud.	1, 2, 3, 4, 5, 6, 8	1, 2, 3
CO5	Examine and put into practice cutting-edge AI methods like Generative Adversarial Networks (GANs), and judge how well AI models-such as evasion tactics and best practices for feature engineering.	1, 2, 3, 4, 5, 6, 8	1, 2, 3
CO6	Gain expertise in using Python for AI applications in cybersecurity, including e nvironment and tool setup.	1, 2, 3, 4, 5, 11, 12	1, 2, 3

BLOOM'S LEVEL OF THE COURSE OUTCOMES:

CO#	Bloom's Level					
	Remember(L1)	Understand(L2)	pply (L3)	nalyze (L4)	aluate (L5)	reate (L6)
CO1	✓					
CO2		✓				
CO3				✓		
CO4				✓		
CO5					✓	
CO6			✓			

COURSE ARTICULATIONMATRIX:

CO#/ POs	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	P O	PS O	PS O	PS O
CO1	3	3	3	3	3	3	-	-	1	1	1	3	2	2	3
CO2	3	3	3	3	3	3	-	-	1	1	1	3	2	2	3
CO3	3	3	3	3	3	3	-	-	1	1	1	3	2	2	3
CO4	3	3	3	3	3	3	-	-	1	1	1	3	2	2	3
CO5	3	3	3	3	3	3	-	-	1	1	1	3	2	2	3
CO6	3	3	3	3	3	2	-	1	3	2	3	3	2	2	3

Note: 1-Low, 2-Medium, 3-High

COURSE CONTENT

Introduction to AI for Cybersecurity Professionals Applying Cloud Computing, Setting Up Your AI for Cybersecurity Arsenal Getting to know Python for AI and cybersecurity, Ham or Spam? Detecting Email Cybersecurity Threats with AI

Malware Threat Detection Malware analysis at a glance, Network Anomaly Detection with AI Network anomaly detection techniques

Securing User Authentication Abuse Prevention, Fraud Prevention with Cloud AI Solutions Introducing fraud detection algorithms

GANs - Attacks and Defenses GANs in a nutshell, Evaluating Algorithms Best practices of feature engineering, Assessing your AI Arsenal Evading ML detectors

Self Learning Component: Exploring the tools on Artificial Intelligence tools on Cyber security and

TEXT BOOKS:

1. Alessandro Parisi, Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies, Packt Publication, August 2019, ISBN: 9781789804027

REFERENCE BOOKS:

1. L. F. Sikos, Ed., AI in Cybersecurity. Springer International Publishing, 2019. doi: 10.1007/978-3-319- 98842-9.
2. Shengjie Xu, Cybersecurity In Intelligent Networking Systems by, 2022, John Wiley
3. James Graham, Richard Howard and Ryan Otson, Cyber Security Essentials, CRC Press, 2011.

JOURNALS/MAGAZINES:

1. Journal of Information Security and Applications (JISA) - Elsevier
2. Security and Privacy-Wiley
3. ACM Transactions on Intelligent Systems and Technology
4. IET Cyber-Physical Systems: Theory & Applications

SWAYAM NPTEL/MOOCs

1. https://www.udemy.com/course/the-complete-artificial-intelligence-for-cyber-security-2021/
2. https://www.coursera.org/learn/gen-ai-in-cybersecurity
3. https://onlinecourses.nptel.ac.in/noc23_cs127/preview

Course Title	Capstone Project Phase-II				Course Type		HC	
Course Code	B24EH0802	Credits	9		Class		VIII Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes		Assessment in	
	Lecture	-	-	-				
	Tutorial	-	-	-	Theory	Practical	CIE	SEE
	Practical	10	20	20				
	Total	**10**	**20**	**20**	**-**	**280**	**50**	**50**

COURSE OVERVIEW:

Project Phase-2 is continuation of Project Phase-1 from semester VII.

COURSE OBJECTIVE (S):

The objective of this course are to

1. To allow students to demonstrate a wide range of the skills learned during their course of study by asking them to deliver a product that has passed through the design, analysis, testing and evaluation.
2. To encourage multidisciplinary research through the integration learned in a number of courses.
3. To allow students to develop problem solving, analysis, synthesis and evaluation skills.
4. To encourage teamwork.
5. To improve students' communication skills by asking them to produce both a professional report and to give an oral presentation

COURSE OUTCOMES (CO'S):

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project work to attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing the project work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysis required for carrying out the project.	11	1,2,3

Note: 1-Low, 2-Medium, 3-High

CO12	Engage in lifelong learning to improve knowledge and competence in the chosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

	Bloom's Level					
	Remember	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Create**
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3
CO12												3	3	3	3

The students are informed to follow the following instructions to complete the Project Phase-2:

1. Each student group shall, conduct the required experiment to implement the proposed project with the consultation of respective guides.
2. Each student groups shall be reviewed and evaluated in two reviews through the semester and finally each group shall demonstrate the completed project to a team of examiners.
3. Review 1 shall be on the presentation of the methodology employed and model created.
4. Review 2 shall be on the presentation on the functional project.
5. Finally, the Capstone-Project Phase-2 shall conclude with each project group apply for patent or copyright and publish a paper in SCOPUS indexed journals. In Semester end examination, each student in groups shall be evaluated, based on the course outcomes.

Course Title	Internship/Global Certification				Course Type		HC	
Course Code	B24EH0803	Credits	2		Class		VIII Semester	
Course Structure	LTP	Credits	Contact Hours	Work Load	Total Number of Classes		Assessment in	
	Lecture							
	Tutorial				Theory	Practical	CIE	SEE
	Practical	2	4	4				
	Total	2	4	4		56	50	50

COURSE OVERVIEW:

An internship can present students with new skills and opportunities. Interns not only gain technical knowledge within the industry of their choice, but they also learn how to interact with professionals in a workplace setting, and develop essential soft skills like time management, organization, adaptability, problem-solving and teamwork.

COURSE OBJECTIVES:

The objective of this course are to

1. To allow students to develop problem solving, analysis, synthesis and evaluation skills.
2. To encourage team work.
3. To help students to gain exposure into industries.
4. To improve students' communication skills by asking them to produce both a professional report andto give an oral presentation.

COURSE OUTCOMES:

On successful completion of this course, the student shall be able to:

CO#	Course Outcomes	POs	PSOs
CO1	Demonstrate in-depth knowledge on the project topic	1	1,2,3
CO2	Identify, analyze and formulate complex problem chosen for project workto attain substantiated conclusions.	2	1,2,3
CO3	Design solutions to the chosen project problem.	3	1,2,3
CO4	Undertake investigation of project problem to provide valid conclusions.	4	1,2,3
CO5	Use the appropriate techniques, resources, and modern engineering tools necessary for project work.	5	1,2,3
CO6	Apply project results for sustainable development of the society.	6	1,2,3
CO7	Understand the impact of project results in the context of environmental sustainability.	7	1,2,3
CO8	Understand professional and ethical responsibilities while executing theproject work.	8	1,2,3
CO9	Function effectively as individual and a member in the project team.	9	1,2,3
CO10	Develop communication skills, both oral and written for preparing and presenting project report.	10	1,2,3
CO11	Demonstrate knowledge and understanding of cost and time analysisrequired for carrying out the project.	11	1,2,3
O12	Engage in lifelong learning to improve knowledge and competence in thechosen area of the project.	12	1,2,3

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO#	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
CO1		√				
CO2			√			
CO3						√
CO4				√		
CO5			√			
C06			√			
CO7		√				
CO8		√				
CO9	√			√		
CO10			√			√
CO11		√		√		
CO12			√			

COURSE ARTICULATION MATRIX

CO#/ Pos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3												3	3	3
CO2		3											3	3	3
CO3			3										3	3	3
CO4				3									3	3	3
CO5					3								3	3	3
C06						3							3	3	3
CO7							3						3	3	3
CO8								3					3	3	3
CO9									3				3	3	3
CO10										3			3	3	3
CO11											3		3	3	3
CO12												3	3	3	3

Note: 1-Low,2-Medium,3-High

The students are informed to follow the following instructions to complete the Internship:

1. The internship should be paid internship in IT industry.
2. The internship should be for minimum of three months.
3. The project title must be submitted in form a document (synopsis) that contains the proposed titleof the project, an abstract, Introduction and their roles and responsibilities in company.
4. Each student shall be reviewed and evaluated in two reviews through the semester.
5. Review 1 shall be on the presentation of the synopsis.
6. Review 2 shall be on the presentation on the roles and responsibilities carried out with modulecompletion results (as applicable).

=

REVA UNIVERSITY

Bengaluru, India

Rukmini Knowledge Park, Kattigenahalli
Yelahanka, Bengaluru - 560 064
Karnataka, India.

Ph: +91- 90211 90211, +91 80 4696 6966
E-mail: admissions@reva.edu.in

www.reva.edu.in

Follow us on

/revauniversity_official /REVAUniversity

/revauniversity_official /@revauniversity_official /REVA University

reva.edu.in +91 90211 90211