B.E-COMPUTER SCIENCE AND ENGINEERING
Regulation 2024
CURRICULUM & SYLLABI

DEPARTMENT VISION

To provide quality education, to become a prominent centre of excellence for producing competent IT professionals, logical thinkers and empowering entrepreneurs to take up real world challenges in the emerging technological society.

DEPARTMENT MISSION

As a department, we are committed

- ❖ To provide high quality education to the students, and equip them with the state-of- the-art knowledge and skills in computer science that they need to take up real-world challenges.
- ❖ To make students to think beyond the boundaries using analytical and logical skills.
- ❖ To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.
- ❖ To gear up the graduates to grab the opportunities in the core field in pursuing higher education and to become empowering entrepreneurs.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Possess a strong foundation in computer science and apply to design and develop innovative computing solutions for a successful career (Technical Expertise).

PEO 2: Adapt to the rapidly evolving technologies, continuously learn and update their knowledge to pursue higher studies of research and solve problems effectively in a dynamic environment. (Adaptability and Lifelong Learning).

PEO 3: Exhibit strong communication and teamwork skills, the ability to work independently, an ethical understanding of the social and professional implications of computing, and the potential to pursue higher studies or research (Professional and Ethical Responsibility).

PROGRAMME OUTCOMES (POs)

Engineering Graduates will be able to:

PO 1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO 2: Problem analysis: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4: Conduct investigations of complex problems: Use research-based knowledge and research methods 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: The engineer 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 responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and 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)

Graduates will be able to:

PSO 1: Analyze, design and develop computing solutions by applying emerging technologies in areas like artificial intelligence, machine learning, and cyber security.

PSO 2: Apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems.

B.E-COMPUTER SCIENCE AND ENGINEERING
CURRICULUM AND SYLLABI (Regulation 2024)
(Applicable for students admitted from the Academic Year 2024-2025)
Minimum Credits to be Earned: 164

SEMESTER I										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA / ES**	**Course Type**	**Category**
1	24HS101	Technical English	2	0	2	3	4	50/50	TP	HSMC
2	24MA101	Matrices and Calculus	3	1	0	4	4	40/60	T	BSC
3	24PH101	Engineering Physics	3	0	2	4	5	50/50	TP	BSC
4	24CY101	Engineering Chemistry	3	0	2	4	5	50/50	TP	BSC
5	24GE101	Problem Solving using Python Programming	3	0	2	4	5	50/50	TP	ESC
6	24GE102	தமிழர்மரபு / Heritage of Tamils	1	0	0	1	1	40/60	T	HSMC
7	24GE111	Engineering Practices Laboratory	0	0	4	2	4	60/40	P	ESC
8	24MC111	Induction Program	3 Weeks						MC	MC
		TOTAL				22	28			

SEMESTER II										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24MA202	Probability and Statistics	3	1	0	4	4	40/60	T	BSC
2	24CS201	Programming in C	3	0	2	4	5	50/50	TP	PCC
3	24PH205	Physics for Information Science	3	0	0	3	3	40/60	T	BSC
4	24EE204	Basic Electrical and Electronics Engineering	3	0	0	3	3	40/60	T	ESC
5	24GE201	Computer Aided Engineering Graphics	3	0	2	4	5	50/50	TP	ESC
6	24GE202	தமிழரும் தொழில்நுட்பமும் / Tamils and Technology	1	0	0	1	1	40/60	T	HSMC
7	24MC201	Environmental Science	2	0	0	0	2	100/0	MC	MC
		Work Integrated Learning (Industrial Practice)	Regulation 2024, Clause 4.4							AP
		TOTAL				19	23			

SEMESTER III										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24HS301	Universal Human Values	2	0	0	2	2	100/0	T	HSMC
2	24MA302	Discrete Mathematics	3	1	0	4	4	40/60	T	BSC
3	24CS301	Data Structures	3	0	2	4	5	50/50	TP	PCC
4	24CS302	Digital Principles and Computer Organization	3	0	2	4	5	50/50	TP	ESC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
5	24CS303	Object Oriented Programming	3	0	2	4	5	50/50	TP	PCC
6	24CS304	Database Management Systems	3	0	2	4	5	50/50	TP	PCC
7	24SD311	Aptitude and Coding Skills -1	0	0	2	1	2	100/0	P	EEC
8	24EL311	Innovation and Product Development -1 (Idea Generation)	0	0	2	1	1	100/0	EL	EEC
		Work Integrated Learning (Industrial Training-I)	Regulation 2024, Clause 4.4							AP
TOTAL						24	29			

SEMESTER IV										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CS401	Theory of Computation	3	0	0	3	3	40/60	T	PCC
2	24CS402	Algorithms	3	0	2	4	5	50/50	TP	PCC
3	24CS403	Introduction to Operating Systems	3	0	2	4	5	50/50	TP	PCC
4	24CS404	Object Oriented Software Engineering	3	0	2	4	5	50/50	TP	PCC
5	24IT302	Foundations of Data Science	3	0	2	4	5	50/50	TP	PCC
6	24SD411	Aptitude and Coding Skills -2	0	0	2	1	2	100/0	P	EEC
7	24EL411	Innovation and Product Development -2 (Conceptualization)	0	0	2	1	2	100/0	EL	EEC
8		Mandatory Course- 3 (Choose from list)	2	0	0	0	2	100/0	MC	MC
		Work Integrated Learning (Industrial Training-II)	Regulation 2024, Clause 4.4							AP
TOTAL						21	29			

SEMESTER V										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CS501	Computer Networks	3	0	2	4	5	50/50	TP	PCC
2	24CS502	Compiler Design	3	0	2	4	5	50/50	TP	PCC
3	24CS503	Cryptography and Cyber Security	3	0	0	3	3	40/60	T	PCC
	24IT402	Artificial Intelligence and Machine Learning	3	0	2	4	5	50/50	TP	PCC
5		PEC-1	2	0	2	3	4	50/50	TP	PEC
6		OEC-1	3	0	0	3	3	40/60	T	OEC
7	24SD511	Advanced Aptitude and Coding Skills 1	0	0	2	1	2	100/0	P	EEC
8	24EL511	Innovation and Product Development-3 (Prototype Development and Testing)	0	0	2	1	2	100/0	EL	EEC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
9		Mandatory Course- 4 (Choose from list)	2	0	0	0	2	100/0	MC	MC
		Work Integrated Learning (Industrial Problem Solving-I)	Regulation 2024, Clause 4.4							AP
						TOTAL	23	31			

SEMESTER VI										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24MG601	Economics, Finance & Accounting	3	0	0	3	3	40/60	T	HSMC
2	24AD505	Cloud Computing	2	0	2	3	4	50/50	TP	PCC
3	24CS601	Embedded Systems and IoT	3	0	2	4	5	50/50	TP	PCC
4		PEC-2	2	0	2	3	4	50/50	TP	PEC
5		PEC-3	2	0	2	3	4	50/50	TP	PEC
6		OEC-2	3	0	0	3	3	40/60	T	OEC
	24SD611	Technical Proficiency -1	0	0	2	1	2	100/0	P	EEC
8	24SD612	Advanced Aptitude and Coding Skills 2	0	0	2	1	2	100/0	P	EEC
9	24SD613	Corporate Communication Skills	0	0	4	2	4	100/0	P	HSMC
10	24EL611	Innovation and Product Development–4 (Patent Filing / Startup Registration)	0	0	2	1	2	100/0	EL	EEC
		Work Integrated Learning (Industrial Problem Solving-II)	Regulation 2024, Clause 4.4							AP
						TOTAL	24	33			

SEMESTER VII										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CS701	Image and Video Analytics	2	0	2	3	4	50/50	TP	PCC
2		PEC-4	2	0	2	3	4	50/50	TP	PEC
3		PEC-5	2	0	2	3	4	50/50	TP	PEC
4		PEC-6	2	0	2	3	4	50/50	TP	PEC
5		OEC-3	3	0	0	3	3	40/60	T	OEC
6		OEC-4	3	0	0	3	3	40/60	T	OEC
7	24CS711	Mini Project	0	0	4	2	4	60/40	P	EEC
8	24SD711	Technical Proficiency -2	0	0	2	1	2	100/0	P	EEC
						TOTAL	21	28			

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
SEMESTER VIII										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CS811	Project Work / Semester Internship	0	0	20	10	20	60/40	P	EEC
2	Personality and Character Development Activity Points (Refer Regulation 2024, Clause 4.2)									AP
3	Work Integrated Learning Activity Points (Refer Regulation 2024, Clause 4.4)									AP
		TOTAL				10	20			

Humanities, Social Science and Management Courses (12 Credits)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24HS101	Technical English	2	0	2	3	4	50/50	TP	HSMC
2	24GE102	தமிழர்மரபு / Heritage of Tamils	1	0	0	1	1	40/60	T	HSMC
3	24GE202	தமிழரும் தொழில்நுட்பமும் / Tamils and Technology	1	0	0	1	1	40/60	T	HSMC
4	24HS301	Universal Human Values	2	0	0	2	2	100/0	T	HSMC
5	24MG601	Economics, Finance & Accounting	3	0	0	3	3	40/60	T	HSMC
6	24SD613	Corporate Communication Skills	0	0	4	2	4	100/0	P	HSMC

Basic Science Courses (23 Credits)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24MA101	Matrices and Calculus	3	1	0	4	4	40/60	T	BSC
2	24PH101	Engineering Physics	3	0	2	4	5	50/50	TP	BSC
3	24CY101	Engineering Chemistry	3	0	2	4	5	50/50	TP	BSC
4	24MA202	Probability and Statistics	3	1	0	4	4	40/60	T	BSC
5	24PH205	Physics for Information Science	3	0	0	3	3	40/60	T	BSC
6	24MA302	Discrete Mathematics	3	1	0	4	4	40/60	T	BSC

Engineering Science Courses (17 Credits)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24GE101	Problem Solving using Python Programming	3	0	2	4	5	50/50	TP	ESC
2	24GE111	Engineering Practices Laboratory	0	0	4	2	4	60/40	P	ESC
3	24EE204	Basic Electrical and Electronics Engineering	3	0	0	3	3	40/60	T	ESC
4	24GE201	Computer Aided Engineering Graphics	3	0	2	4	5	50/50	TP	ESC
5	24CS302	Digital Principles and Computer Organization	3	0	2	4	5	50/50	TP	ESC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
Professional Core Courses (60 Credits)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CS201	Programming in C	3	0	2	4	5	50/50	TP	PCC
2	24CS301	Data Structures	3	0	2	4	5	50/50	TP	PCC
3	24CS303	Object Oriented Programming	3	0	2	4	5	50/50	TP	PCC
4	24CS304	Database Management Systems	3	0	2	4	5	50/50	TP	PCC
5	24CS401	Theory of Computation	3	0	0	3	3	40/60	T	PCC
6	24CS402	Algorithms	3	0	2	4	5	50/50	TP	PCC
7	24CS403	Introduction to Operating Systems	3	0	2	4	5	50/50	TP	PCC
8	24CS404	Object Oriented Software Engineering	3	0	2	4	5	50/50	TP	PCC
9	24IT302	Foundations of Data Science	3	0	2	4	5	50/50	TP	PCC
10	24CS501	Computer Networks	3	0	2	4	5	50/50	TP	PCC
11	24CS502	Compiler Design	3	0	2	4	5	50/50	TP	PCC
12	24CS503	Cryptography and Cyber Security	3	0	0	3	3	40/60	T	PCC
13	24IT402	Artificial Intelligence and Machine Learning	3	0	2	4	5	50/50	TP	PCC
14	24AD505	Cloud Computing	2	0	2	3	4	50/50	TP	PCC
15	24CS601	Embedded Systems and IoT	3	0	2	4	5	50/50	TP	PCC
16	24CS701	Image and Video Analytics	2	0	2	3	4	50/50	TP	PCC

Professional Elective Courses - Vertical-1 (Data Science)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**
1	24CSE01	Data Visualization	2	0	2	3	4	50/50	TP	PEC
2	24ITE15	Deep Learning	2	0	2	3	4	50/50	TP	PEC
3	24ITE16	Text and Speech Analytics	2	0	2	3	4	50/50	TP	PEC
4	24ITE21	Exploratory Data Analysis	2	0	2	3	4	50/50	TP	PEC
5	24ITE22	Recommender Systems	2	0	2	3	4	50/50	TP	PEC
6	24CSE20	Business Analytics	2	0	2	3	4	50/50	TP	PEC
7	24CSE02	Big Data Analytics	2	0	2	3	4	50/50	TP	PEC

Professional Elective Courses - Vertical-2 (Full Stack Web Development)										
S.No.	**Course Code**	**Course**	**L**	**T**	**P**	**C**	**Prds. /wk.**	**CA/ ES**	**Course Type**	**Category**

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
1	24CSE03	Software Testing and Automation	2	0	2	3	4	50/50	TP	PEC
2	24CSE04	DevOps	2	0	2	3	4	50/50	TP	PEC
3	24CSE05	Principles of Programming Languages	2	0	2	3	4	50/50	TP	PEC
4	24ITE02	App Development	2	0	2	3	4	50/50	TP	PEC
5	24ITE03	Cloud Services Management	2	0	2	3	4	50/50	TP	PEC
6	24ITE04	UI and UX Design	2	0	2	3	4	50/50	TP	PEC
7	24ITE05	Web Application Security	2	0	2	3	4	50/50	TP	PEC
8	24ADE05	Web Technologies	2	0	2	3	4	50/50	TP	PEC

Professional Elective Courses - Vertical-3 (Cloud Computing And Data Center										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24CSE06	Virtualization	2	0	2	3	4	50/50	TP	PEC
2	24ITE03	Cloud Services Management	2	0	2	3	4	50/50	TP	PEC
3	24CSE07	Data Warehousing	2	0	2	3	4	50/50	TP	PEC
4	24CSE08	Storage Technologies	2	0	2	3	4	50/50	TP	PEC
5	24CSE09	Software Defined Networks and Applications	2	0	2	3	4	50/50	TP	PEC
6	24CSE10	Stream Processing	2	0	2	3	4	50/50	TP	PEC
7	24ITE13	Security and Privacy in Cloud	2	0	2	3	4	50/50	TP	PEC

Professional Elective Courses - Vertical-4 (Cyber Security And Data Privacy)										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24ITE06	Ethical Hacking	2	0	2	3	4	50/50	TP	PEC
2	24ITE07	Digital and Mobile Forensics	2	0	2	3	4	50/50	TP	PEC
3	24ITE08	Social Network Security	2	0	2	3	4	50/50	TP	PEC
4	24ITE09	Modern Cryptography	2	0	2	3	4	50/50	TP	PEC
5	24ITE10	Engineering Secure Software Systems	2	0	2	3	4	50/50	TP	PEC
6	24ITE11	Cryptocurrency and Blockchain Technologies	2	0	2	3	4	50/50	TP	PEC
7	24ITE12	Network Security	2	0	2	3	4	50/50	TP	PEC
8	24ITE13	Security and Privacy in Cloud	2	0	2	3	4	50/50	TP	PEC

Professional Elective Courses - Vertical-5 (Artificial Intelligence And Machine

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24CSE18	Knowledge Engineering	2	0	2	3	4	50/50	TP	PEC
2	24ADE01	Soft Computing	2	0	2	3	4	50/50	TP	PEC
3	24ITE15	Deep Learning	2	0	2	3	4	50/50	TP	PEC
4	24ITE16	Text and Speech Analytics	2	0	2	3	4	50/50	TP	PEC
5	24ITE17	Optimization Techniques	2	0	2	3	4	50/50	TP	PEC
6	24CSE19	Game Theory	2	0	2	3	4	50/50	TP	PEC
7	24ITE19	Cognitive Science	2	0	2	3	4	50/50	TP	PEC
8	24ITE20	Ethics and AI	2	0	2	3	4	50/50	TP	PEC

Professional Elective Courses - Vertical-6 (Emerging Technologies)										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24CSE11	Game Development	2	0	2	3	4	50/50	TP	PEC
2	24CSE12	FOG Computing	2	0	2	3	4	50/50	TP	PEC
3	24CSE13	Robotic Process Automation	2	0	2	3	4	50/50	TP	PEC
4	24CSE14	EDGE Computing	2	0	2	3	4	50/50	TP	PEC
5	24CSE15	Natural Language Processing and Conversational AI	2	0	2	3	4	50/50	TP	PEC
6	24CSE16	Quantum Computing	2	0	2	3	4	50/50	TP	PEC
7	24CSE17	3D Printing and Design	2	0	2	3	4	50/50	TP	PEC
8	24ITE11	Cryptocurrency and Blockchain Technologies	2	0	2	3	4	50/50	TP	PEC

Open Elective Courses (12 Credits)										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24CEO01	Environmental Science and Sustainability	3	0	0	3	3	40/60	T	OEC
2	24CEO02	Green Building Design	3	0	0	3	3	40/60	T	OEC
3	24CEO03	Municipal Solid Waste Management	3	0	0	3	3	40/60	T	OEC
4	24CEO04	Sustainable Infrastructure	3	0	0	3	3	40/60	T	OEC
5	24CEO05	Disaster Management	3	0	0	3	3	40/60	T	OEC
6	24CEO06	Geographical Information System	3	0	0	3	3	40/60	T	OEC
7	24CEO07	Environment and Agriculture	3	0	0	3	3	40/60	T	OEC
8	24CEO08	Earthquake Engineering	3	0	0	3	3	40/60	T	OEC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
9	24EEO01	Electrical Safety	3	0	0	3	3	40/60	T	OEC
10	24EEO02	Introduction to E-Vehicle	3	0	0	3	3	40/60	T	OEC
11	24EEO03	Hybrid Energy Technology	3	0	0	3	3	40/60	T	OEC
12	24EEO04	Solar Photovoltaic System	3	0	0	3	3	40/60	T	OEC
13	24EEO05	Energy Forecasting, Modeling, and Project Management	3	0	0	3	3	40/60	T	OEC
14	24EEO06	Energy Efficient Buildings Design	3	0	0	3	3	40/60	T	OEC
15	24EEO07	Government Rules, Opportunities, Testing, and Certification of EV	3	0	0	3	3	40/60	T	OEC
16	24EEO08	PLC and Automation	3	0	0	3	3	40/60	T	OEC
17	24ECO01	Micro Sensors and Nano Sensors	3	0	0	3	3	40/60	T	OEC
18	24ECO02	Fundamentals of VLSI	3	0	0	3	3	40/60	T	OEC
19	24ECO03	Mass Communication	3	0	0	3	3	40/60	T	OEC
20	24ECO04	Sensor Technology	3	0	0	3	3	40/60	T	OEC
21	24ECO05	Nanomaterials and Devices	3	0	0	3	3	40/60	T	OEC
22	24ECO06	ML for Electronic Systems	3	0	0	3	3	40/60	T	OEC
23	24ECO07	Wireless and Mobile Communication	3	0	0	3	3	40/60	T	OEC
24	24MEO01	Introduction to MEMS and NEMS	3	0	0	3	3	40/60	T	OEC
25	24MEO02	Energy Conservation and Management	3	0	0	3	3	40/60	T	OEC
26	24MEO03	Fundamentals of Additive Manufacturing	3	0	0	3	3	40/60	T	OEC
27	24MEO04	Lean Six Sigma	3	0	0	3	3	40/60	T	OEC
28	24MEO05	Agriculture Technology	3	0	0	3	3	40/60	T	OEC
29	24MHO01	Robots and systems in smart Manufacturing	3	0	0	3	3	40/60	T	OEC
30	24MHO02	Robotics and Automation	3	0	0	3	3	40/60	T	OEC
31	24MHO03	Autonomous Mobile Robots	3	0	0	3	3	40/60	T	OEC
32	24MHO04	Introduction to Drone Technology	3	0	0	3	3	40/60	T	OEC
33	24MHO05	Medical Mechatronics	3	0	0	3	3	40/60	T	OEC
34	24MHO06	Sensors and Actuators	3	0	0	3	3	40/60	T	OEC
35	24MHO07	Micro Electro Mechanical System	3	0	0	3	3	40/60	T	OEC
36	24MHO08	Industry 4.0	3	0	0	3	3	40/60	T	OEC
37	24BMO01	Fundamentals of Radiological Equipments	3	0	0	3	3	40/60	T	OEC
38	24BMO02	Biomedical Instrumentation Basics	3	0	0	3	3	40/60	T	OEC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
39	24BMO03	Medical Robotics	3	0	0	3	3	40/60	T	OEC
40	24BMO04	Principles of Telemedicine	3	0	0	3	3	40/60	T	OEC
41	24BMO05	Hospital Management	3	0	0	3	3	40/60	T	OEC
42	24BMO06	Bio MEMS and Applications	3	0	0	3	3	40/60	T	OEC
43	24BMO07	Fundamentals of Brain Computer Interface	3	0	0	3	3	40/60	T	OEC
44	24BMO08	Principles of Rehabilitation Engineering	3	0	0	3	3	40/60	T	OEC
45	24CHO01	Industrial pollution prevention and control	3	0	0	3	3	40/60	T	OEC
46	24CHO02	Petroleum Technology	3	0	0	3	3	40/60	T	OEC
47	24CHO03	Green Chemistry and Engineering	3	0	0	3	3	40/60	T	OEC
48	24CHO04	Bio-Energy Technology	3	0	0	3	3	40/60	T	OEC
49	24CHO05	Carbon capture utilization and storage	3	0	0	3	3	40/60	T	OEC
50	24CHO06	Battery Technology	3	0	0	3	3	40/60	T	OEC
51	24CHO07	Energy Management	3	0	0	3	3	40/60	T	OEC
52	24CHO08	Environmental Audit	3	0	0	3	3	40/60	T	OEC
53	24ITO01	Introductions to Full Stack Web Development	3	0	0	3	3	40/60	T	OEC
54	24ITO02	Introduction to Digital Marketing	3	0	0	3	3	40/60	T	OEC
55	24ITO03	Social Media Marketing	3	0	0	3	3	40/60	T	OEC
56	24ITO04	Ethical Hacking	3	0	0	3	3	40/60	T	OEC
57	24ITO05	Information Storage Management	3	0	0	3	3	40/60	T	OEC
58	24ITO07	Software Testing Essentials	3	0	0	3	3	40/60	T	OEC
59	24ADO05	R Programming	3	0	0	3	3	40/60	T	OEC
60	24ADO07	AI in Healthcare Applications	3	0	0	3	3	40/60	T	OEC

Employability Enhancement Courses (22 Credits)										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24SD311	Aptitude and Coding Skills -1	0	0	2	1	2	100/0	P	EEC
2	24EL311	Innovation and Product Development -1 (Idea Generation)	0	0	2	1	1	100/0	EL	EEC
3	24SD411	Aptitude and Coding Skills -2	0	0	2	1	2	100/0	P	EEC
4	24EL411	Innovation and Product Development -2 (Conceptualization)	0	0	2	1	2	100/0	EL	EEC
5	24SD511	Advanced Aptitude and Coding Skills 1	0	0	2	1	2	100/0	P	EEC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
6	24EL511	Innovation and Product Development-3 (Prototype Development and Testing)	0	0	2	1	2	100/0	EL	EEC
7	24SD611	Technical Proficiency -1	0	0	2	1	2	100/0	P	EEC
8	24SD612	Advanced Aptitude and Coding Skills 2	0	0	2	1	2	100/0	P	EEC
9	24EL611	Innovation and Product Development–4 (Patent Filing / Startup Registration)	0	0	2	1	2	100/0	EL	EEC
10	24CS711	Mini Project	0	0	4	2	4	60/40	P	EEC
11	24SD711	Technical Proficiency -2	0	0	2	1	2	100/0	P	EEC
12	24CS811	Project Work / Semester Internship	0	0	20	10	20	60/40	P	EEC

MANDATORY COURSES –1 & 2										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24MC111	Induction Program	3 Weeks						MC	MC
2	24MC201	Environmental Science	2	0	0	0	2	100/0	MC	MC

MANDATORY COURSES – 3										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24MC301	Introduction to Women and Gender Studies	2	0	0	0	2	100/0	MC	MC
2	24MC302	Elements of Literature	2	0	0	0	2	100/0	MC	MC
3	24MC303	Film Appreciation	2	0	0	0	2	100/0	MC	MC
4	24MC304	Well, Being with Traditional Practices Yoga, Ayurveda and Siddha	2	0	0	0	2	100/0	MC	MC
5	24MC305	History of Science and Technology in India	2	0	0	0	2	100/0	MC	MC
6	24MC306	Political and Economic Thought for a Humane Society	2	0	0	0	2	100/0	MC	MC
7	24MC307	Indian Constitution	2	0	0	0	2	100/0	MC	MC
8	24MC308	Bureau of Indian Standards (BIS)	2	0	0	0	2	100/0	MC	MC

MANDATORY COURSES - 4										
S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA/ ES	Course Type	Category
1	24MC401	Japanese Language	2	0	0	0	2	100/0	MC	MC
2	24MC402	German Language	2	0	0	0	2	100/0	MC	MC
3	24MC403	Korean Language	2	0	0	0	2	100/0	MC	MC
4	24MC404	French Language	2	0	0	0	2	100/0	MC	MC
5	24MC405	Hindi Language	2	0	0	0	2	100/0	MC	MC

S.No.	Course Code	Course	L	T	P	C	Prds. /wk.	CA / ES	Course Type	Category
6	24MC406	Industrial Safety	2	0	0	0	2	100/0	MC	MC
7	24MC407	Disaster Risk Reduction and Management	2	0	0	0	2	100/0	MC	MC
8	24MC408	State, Nation Building and Politics in India	2	0	0	0	2	100/0	MC	MC

B.E-COMPUTER SCIENCE AND ENGINEERING

CURRICULUM AND SYLLABI (Regulation 2024)

SEMESTERWISE CREDITS DISTRIBUTION									
Category	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**Credits**
HSMC	4	1	2			5			12
BSC	12	7	4						23
ESC	6	7	4						17
PCC		4	12	19	15	7	3		60
PEC					3	6	9		18
OEC					3	3	6		12
EEC			2	2	2	3	3	10	22
MC									
Total	**22**	**19**	**24**	**21**	**23**	**24**	**21**	**10**	**164**

B.E-COMPUTER SCIENCE AND ENGINEERING

Regulations 2024

Syllabi Sem III and IV

Course Code	24MA302	Course Name	DISCRETE MATHEMATICS	Course Category	T	Basic Science Courses	L	T	P	C
							3	1	0	4

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Science and Humanities	Data Book / Codes / Standards	Nil		

Course Objective: ***The purpose of learning this course is to:***

- To extend students logical and mathematical ability to deal with abstraction.
- To introduce most of the basic terminologies used in computer science courses and application ideas to solve practical problems.
- To understand the basic concepts of combinatorics and graph theory.
- To familiarize the applications of algebraic structures .
- To understand the concepts and significance of lattices and Boolean algebra which are widely used in computer science and engineering.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Understand the validity of the logical arguments, mathematical proofs and correctness of the algorithm	U	3	3	2	3	2	-	-	-	-	-	-	3	-		
CO-2:	Apply Combinatorial counting techniques in solving combinatorial related problems.	AP	3	3	2	3	2	-	-	-	-	-	-	3	-		
CO-3:	Use graph models and their connectivity, travers ability in solving real world problems.	AP	3	3	2	3	2	-	-	-	-	-	-	3	-		
CO-4:	Understand the significance of algebraic structural ideas used in coding theory and cryptography	U	3	3	2	3	2	-	-	-	-	-	-	3	-		
CO-5:	Apply Boolean laws and Boolean functions in combinatorial circuit designs	AP	3	3	2	3	2	-	-	-	-	-	-	3	-		

Unit-1 **LOGIC AND PROOFS**	***9+3 Periods***
Propositional Logic–Propositional Equivalences–Normal Forms–Predicates and Quantifiers–Nested Quantifiers–Rules of Inference–Introduction to Proofs–Proof Methods and Strategy	
Unit-2 **COMBINATORICS**	***9+3 Periods***
Mathematical Induction –Strong Induction and Well Ordering–The Basics of Counting–The Pigeonhole Principle–Permutations and Combinations–Recurrence Relations–Solving Linear Recurrence Relations Using Generating Functions –Inclusion-Exclusion Principle and its Applications	
Unit-3 **GRAPHS**	***9+3 Periods***
Graphs and Graph Models –Graph Terminology and Special types of Graphs –Matrix Representation of Graphs and Graph Isomorphism –Connectivity–Euler and Hamiltonian Paths.	
Unit-4 **ALGEBRAIC STRUCTURES**	***9+3 Periods***
Groups–Subgroups – Homomorphisms – Normal Subgroups and Cosets – Lagrange's Theorem –Rings and Fields (Definitions and Examples	
Unit-5 **LATTICES AND BOOLEAN ALGEBRA**	***9+3 Periods***
Partial Ordering – Posets – Lattices as Posets– Properties of Lattices –Lattices as Algebraic Systems– Sub lattices –Direct Product and Homomorphism – Some Special Lattices– Boolean Algebra.	

	Total 60 Periods

	Text Book	**Reference**
Learning Resources	1.Kenneth H.Rosen, “Discrete Mathematics and its Applications”, Tata McGraw Hill Pub.Co.Ltd., Seventh Edition, Special Indian Edition,NewDelhi,2011. 2.Tremblay J.P. and Manohar R, “Discrete Mathematical Structures with Applications to Computer Science”, Tata McGraw Hill Pub. Co. Ltd. ,Third Edition,NewDelhi,2013.	1.Thomas Koshy, ”Discrete Mathematics with Applications”, ElsevierPublications,Boston,2004. 2.GrimaldiR.P.,“Discrete and Combinatorial Mathematics”, Pearson Education Pvt.Ltd.,5th Edition,Singapore,2004

Course Code	24CS301	Course Name	DATA STRUCTURES	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	C Programming	Co- requisite Courses	Nil	Progressive Courses	Algorithms
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	Nil		

Course Objective: ***The purpose of learning this course is to:***

- Understand the concepts of ADTs.
- Learn linear data structures - lists, stacks, and queues.
- Understand non-linear data structures - trees and graphs.
- Understand sorting, searching and hashing algorithms.
- Apply Tree and Graph structures.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Implement list ADT using arrays and linked lists (singly, circular, and doubly linked lists).	AP	2	2	2	2	2	-	-	-	-	-	-	-	2	2	2
CO-2:	Explain the structure and operations of Stack and Queue ADTs	U	1	1	1	1	-	-	-	-	-	-	-	-	1	1	1
CO-3:	Describe Tree ADT and different tree traversal techniques	U	1	1	1	1	-	-	-	-	-	-	-	-	1	1	1
CO-4:	Apply appropriate graph algorithms for graph applications	AP	2	2	2	2	2	-	-	-	-	-	-	-	2	2	2
CO-5:	Analyze the various searching and sorting algorithms	AN	3	2	2	2	3	-	-	-	-	-	-	-	3	3	3

Unit-1 LISTS **9 Periods**

Abstract Data Types (ADTs) - List ADT - Array-based implementation - Linked list implementation- Singly linked lists - Circularly linked lists - Doubly-linked lists - Applications of lists – Polynomial ADT .

Unit-2 STACKS AND QUEUES **9 Periods**

Stack ADT - Operations - Applications - Balancing Symbols - Evaluating arithmetic expressions-Infix to Postfix conversion - Function Calls - Queue ADT - Operations - Circular Queue - DeQueue- Applications of Queues.

Unit-3 TREES **9 Periods**

Tree ADT - Tree Traversals - Binary Tree ADT - Expression trees - Binary Search Tree ADT - AVL Trees - Priority Queue (Heaps) - Binary Heap.

Unit-4 MULTIWAY SEARCH TREES AND GRAPHS **9 Periods**

B-Tree - B+ Tree - Graphs - Types-Representation - Breadth-first traversal - Depth-first traversal – Bi-connectivity - Euler circuits - Dijkstra's algorithm - Minimum Spanning Tree – Prim's algorithm – Kruskal's algorithm.

Unit-5 SEARCHING, SORTING AND HASHING TECHNIQUES **9 Periods**

Searching - Linear Search - Binary Search. Sorting - Bubble sort - Selection sort - Insertion sort - Quick sort -Merge Sort - Hashing - Hash Functions - Separate Chaining - Open Addressing -Rehashing - Extendible Hashing.

Total Theory: 45 Periods

List of Indicative Experiments	Total Practical: 30 Periods

1. Array implementation of Stack, Queue and Circular Queue ADTs
2. Implementation of Singly Linked List
3. Linked list implementation of Stack and Linear Queue ADTs
4. Implementation of Polynomial Manipulation using Linked list
5. Implementation of Evaluating Postfix Expressions, Infix to Postfix conversion
6. Implementation of Binary Search Trees
7. Implementation of AVL Trees
8. Implementation of Heaps using Priority Queues
9. Implementation of Dijkstra's and Kruskal's Algorithm
10. Implementation of Linear Search and Binary Search
11. Implementation of Insertion Sort and Merge Sort
12. Implementation of Open Addressing ((Linear Probing and Quadratic Probing)

	Text Books	References
Learning Resources	1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson Education, 2018 2. Kamthane, Introduction to Data Structures in C, 1st Edition, Pearson Education, 2020	1. Langsam, Augenstein and Tanenbaum, Data Structures Using C and C++, 2nd Edition, Pearson Education, 2015. 2. Thomas H. Cormen, Charles E. Leiserson, Ronald L.Rivest, Clifford Stein, Introduction to Algorithms", Fourth Edition, Mcgraw Hill/ MIT Press, 2022. 3. Alfred V. Aho, Jeffrey D. Ullman,John E. Hopcroft ,Data Structures and Algorithms, 1st edition, Pearson, 2002. 4. Kruse, Data Structures and Program Design in C, 2nd Edition, Pearson Education, 2006.

Learning Assessment

Continuous Learning Assessment (CLA) (50% weightage)				End Semester Examination (50% weightage)
Theory (25% weightage)		Practical (25% weightage)		
Average of Internal Test (15%)	Critical Thinking Assessment (10%)	Evaluation of Laboratory Observation, Record (15%)	Model Lab Exam (10%)	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%

Course Code	24CS302	Course Name	DIGITAL PRINCIPLES AND COMPUTER ORGANIZATION	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Nil	Co- requisite Courses	Nil	Progressive Courses	Computer Networks
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	Nil		

Course Objective: ***The purpose of learning this course is to:***

- Analyze and design combinational circuits.
- Analyze and design sequential circuits
- Understand the basic structure and operation of a digital computer.
- Study the design of data path unit, control unit for processor and to familiarize with the hazards.
- Understand the concept of various memories and I/O interfacing.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Apply Karnaugh Maps to simplify Boolean expressions and design efficient combinational circuits	AP	3	3	3	3	3	-	-	-	-	-	-	-	2	3	3
CO-2:	Design sequential circuits and analyze the design procedures	R	3	3	3	3	2	-	-	-	-	-	-	-	1	2	2
CO-3:	State the fundamentals of computer systems and analyze the execution of an instruction	U	3	3	3	3	2	-	-	-	-	-	-	-	2	3	1
CO-4:	Analyze execution, stack organization, micro-operations, data path design, control unit design (hardwired and microprogrammed), and pipelining with data and control hazards.	AN	3	3	3	3	1	-	-	-	-	-	-	-	1	3	1
CO-5:	Identify the characteristics memory hierarchy, management, cache memory techniques, interprocess communication, virtual memory, DMA, I/O interfaces, interrupt handling, and interconnection standards.	R	3	3	3	3	1	-	-	-	-	-	-	-	1	2	1

Unit-1 COMBINATIONAL LOGIC **9 Periods**

Combinational Circuits – Karnaugh Map - Analysis and Design Procedures – Binary Adder – Subtractor – Decimal Adder - Magnitude Comparator – Decoder – Encoder – Multiplexers – Demultiplexers

Unit-2 SYNCHRONOUS SEQUENTIAL LOGIC **9 Periods**

Introduction to Sequential Circuits - Flip-Flops – Operation and Excitation tables -NAND- RS Latch - NOR LS Latch, RS flipflop, JK flipflop, T Flipflop, D flipflop-Half Adder- Full Adder-Triggering of FF, Analysis and design of clocked sequential circuits – Design – Moore/Mealy models, State Minimization,State Assignment, Circuit Implementation - Registers – Counters.

Unit-3 COMPUTER FUNDAMENTALS **9 Periods**

Functional Units of a Digital Computer: Von Neumann Architecture – Operation and Operands of Computer Hardware Instruction ,Shift Register , Bidirectional Shift Register- Instruction Set Architecture (ISA): Memory Location, Address and Operation – Instruction and Instruction Sequencing – Addressing Modes, Encoding of Machine Instruction – Interaction between Assembly and High Level Language.

Unit-4 PROCESSOR	**9 Periods**
Instruction Execution -Stack Organization , Instruction Formats, Arithmetic Micro Operations , Logic Micro Operations , Shift Micro operations, Building a Data Path - Designing a Control Unit - Hardwired Control, Microprogrammed Control - Pipelining - Data Hazard - Control Hazards	
Unit-5 MEMORY AND I/O	**9 Periods**
Memory Concepts and Hierarchy - Memory Management - Cache Memories: Mapping and Replacement Techniques , Cache Coherence , Inter Process Communication and synchronization -Virtual Memory - DMA - I/O - Accessing I/O: Parallel and Serial Interface - Interrupt I/O - Interconnection Standards: USB, SATA	
	Total Theory: 45 Periods
List of Indicative Experiments	**Total Practical: 30 Periods**

1. Verification of Boolean theorems using logic gates
2. Design and implementation of combinational circuits using gates for arbitrary functions.
3. Implementation of 4-bit binary adder/subtractor circuits.
4. Implementation of code converters.
5. Implementation of BCD adder, encoder and decoder circuits
6. Implementation of functions using Multiplexers.
7. Implementation of the synchronous counters
8. Implementation of a Universal Shift register
9. Simulator based study of Computer Architecture

	Text Books	References
Learning Resources	1. M. Morris Mano, Michael D. Ciletti, "Digital Design : With an Introduction to the Verilog HDL, VHDL, and System Verilog", Sixth Edition, Pearson Education, 2018. 2. David A. Patterson, John L. Hennessy, "Computer Organization and Design, The Hardware/Software Interface", Sixth Edition, Morgan Kaufmann/Elsevier, 2020.	1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian, "Computer Organization and Embedded Systems", Sixth Edition, Tata McGraw-Hill, 2012. 2. William Stallings, "Computer Organization and Architecture - Designing for Performance", Tenth Edition, Pearson Education, 2016. 3. M. Morris Mano, "Digital Logic and Computer Design", Pearson Education, 2016. 4. Rajaraman, V., & Radhakrishnan, T. (2018). Computer Organization and Architecture. Prentice Hall India. (Author: Former Professor, IIT Kanpur) 5. Bhaskaran, R. (2021). Fundamentals of Digital Logic and Microprocessors. NIT Trichy Publications. 6. Ghosh, K. (2019). Advanced Computer Architecture and Parallel Processing. IIT Kharagpur Press.

Learning Assessment

Continuous Learning Assessment (CLA) (50% weightage)				End Semester Examination (50% weightage)
Theory (25% weightage)		Practical (25% weightage)		
Average of Internal Test (15%)	Critical Thinking Assessment (10%)	Evaluation of Laboratory Observation, Record (15%)	Model Lab Exam (10%)	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%

Course Code	24CS303	Course Name	OBJECT ORIENTED PROGRAMMING	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses		Co- requisite Courses	Data Structures	Progressive Courses	OBJECT ORIENTED SOFTWARE ENGINEERING
Course Offering Department	Computer Science and Engineering		Data Book / Codes Standards	Nil	

Course Objective: ***The purpose of learning this course is to:***

- Understand the fundamental concepts of Object-Oriented Programming and the basics of Java programming.
- Learn the core principles of packages, inheritance, and interfaces in Java.
- Define and implement exception handling and I/O stream operations in Java applications.
- Develop Java applications utilizing threads and generic classes for enhanced functionality.
- Design and create Graphical User Interface (GUI) applications using JavaFX.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Apply the principles of classes and objects to develop solutions for fundamental problems.	U	3	2	3	1	3	2	1	-	3	1	1	1	3	1	1
CO-2:	Develop programs that use inheritance, packages, and interfaces to improve modularity and reusability of code.	U	3	3	3	1	3	2	1	-	2	1	1	1	3	2	1
CO-3:	Create Java applications that effectively use string manipulation, exception handling, and I/O stream operations.	AP	3	3	3	1	2	2	1	-	3	1	1	1	3	3	1
CO-4:	Build multithreaded applications and use generic classes in Java to address real-world problems.	AP	3	3	3	2	3	2	2	-	1	2	1	1	3	2	2
CO-5:	Apply event handling and JavaFX components to develop interactive graphical user interface (GUI) applications.	AP	3	3	3	3	2	2	2	-	2	2	1	1	3	3	2

Unit-1	**INTRODUCTION TO OOP AND JAVA**	**9 Periods**
Programming Paradigms - Features of Object-Oriented Programming - Data Types, Variables and Arrays - Operators - Control Statements -Classes - Methods -Constructors- -Access specifiers - Static members- Overloading Methods - Objects -Passing Object Parameters - Returning Objects -Classes-Static- Nested - Inner Classes-Final.		
Unit-2	**INHERITANCE, PACKAGES AND INTERFACES**	**9 Periods**
Inheritance-Super -Abstract Classes- Method Overriding - Dynamic Method Dispatch - Packages and Interfaces: Packages -Importing Packages - Interfaces.		
Unit-3	**STRING HANDLING, I/O AND EXCEPTION HANDLING**	**9 Periods**
String class- Methods - String Buffer Class-Streams API-File Streams - Reading and Writing Files-Object Serialization-Exception Handling -Try Clause - Catch Clause -Final Clause – Nested try Statements – Built-in Exceptions – User defined Exception-JDBC		
Unit-4	**MULTITHREADING AND GENERIC PROGRAMMING**	**9 Periods**
Multithreaded Programming- Threads – Priorities – Synchronization – Interthread Communication-Generic Programming – Generic classes – Generic Methods – Generic Collections-Bounded Types – Streams API		

Unit-5 JAVAFX EVENT HANDLING, CONTROLS AND COMPONENTS	9 Periods
JAVAFX Events and Controls- Key and Mouse Events-Text Controls- Layouts -Menus -Spring Boot Framework	
	Total Theory: 45 Periods

List of Experiments	**Total Practical: 30 Periods**

1. Implement basic programming using Java
2. Create programs using classes,methods,constructors.
3. Implement inheritance using abstract classes.
4. Implement interface .
5. Implement Exception handling.
6. Develop programs to implement Multithreading.
7. Develop programs to perform File operations and JDBC.
8. Develop programs using Generic classes.
9. Develop Event-driven programs for GUI applications
10. Mini project-Spring Boot

	Text Books	**References**
Learning Resources	1. Y. Daniel Liang, "Introduction to Java Programming and Data Structures, Comprehensive Version", 12th Edition, Pearson Education, 2021. 2. Paul Dietel and Harvey Deitel, "Java - How to Program Early Objects", 11th Edition, Pearson Education, 2018	1. Sachin Malhotra, Sourabh Choudhary, "Programming in Java", Revised 2nd Edition, Oxford University Press, 2018.

Learning Assessment				
Continuous Learning Assessment (CLA) (50% weightage)				**End Semester Examination (50% weightage)**
Theory (25% weightage)		**Practical (25% weightage)**		
Average of Internal Test (15%)	**Critical Thinking Assessment (10%)**	**Evaluation of Laboratory Observation, Record (15%)**	**Model Lab Exam (10%)**	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%

Course Code	24CS304	Course Name	DATABASE MANAGEMENT SYSTEMS	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Nil	Co- requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	Nil		

Course Objective: ***The purpose of learning this course is to:***

- Learn the fundamentals of data models, relational algebra and SQL
- Represent a database system using ER diagrams and to learn normalization techniques
- Understand the fundamental concepts of transaction, concurrency and recovery processing
- Understand the internal storage structures using different file and indexing techniques which will help in physical DB design
- Have an introductory knowledge about the Distributed databases, NOSQL and database security

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Construct SQL Queries using relational algebra	U	2	2	3	2	1	-	-	-	-	-	-	-	2	1	3
CO-2:	Design database using ER model and normalize the database	AP	3	1	1	1	1	1	1	1	1	1	1	1	3	1	2
CO-3:	Construct queries to handle transaction processing and maintain consistency of the database	AP	3	2	3	2	1	1	1	1	1	1	1	1	2	3	3
CO-4:	Compare and contrast various indexing strategies and apply the knowledge to tune the performance of the database	AN	1	2	3	2	2	2	2	2	2	2	2	2	1	2	3
CO-5:	Appraise how advanced databases differ from Relational Databases and find a suitable database for the given requirement.	AN	1	1	3	3	2	2	2	2	2	2	2	2	2	2	2

Unit-1 RELATIONAL DATABASES **7 Periods**

Database system- Data Models - System Architecture- Relational Model - Keys - Relational Algebra - SQL - Embedded SQL

Unit-2 DATABASE DESIGN **9 Periods**

Entity-Relationship model - ER Diagrams - Enhanced ER Model - ER to Relational Mapping - Functional Dependencies - Non-loss Decomposition - First, Second, Third Normal Forms, Dependency Preservation - Boyce/Codd Normal Form - Multi-valued Dependencies and Fourth Normal Form - Join Dependencies and Fifth Normal Form

Unit-3 TRANSACTIONS **9 Periods**

Transaction Concepts - ACID Properties - Schedules - Serializability - Need for Concurrency - Concurrency control -Two Phase Locking- Timestamp - Multiversion - Validation and Snapshot isolation- Multiple Granularity locking - Deadlock Handling - Recovery Concepts - Recovery based on deferred and immediate update - Shadow paging – ARIES Algorithm

Unit-4 STORAGE AND QUERY PROCESSING **9 Periods**

RAID - File Organization - Organization of Record in Files - Data dictionary Storage - Column Oriented Storage- Indexing and Hashing -Ordered Indices - B+ tree Index Files - B tree Index Files - Static Hashing - Dynamic Hashing - Query Processing Overview - Algorithms for Selection, Sorting and join operations - Query optimization using Heuristics - Cost Estimation.

Unit-5	**ADVANCED TOPICS**	**11 Periods**
Distributed Databases: Architecture, Data Storage, Transaction Processing, Query processing and optimization - NOSQL Databases: Introduction - CAP Theorem - Document Based systems - Key value Stores - Database Security:Security issues - Access control based on privileges - Role Based access control - SQL Injection - Statistical Database security - Flow control - Encryption and Public Key infrastructures - Challenges. Hadoop-Mongo DB		
		Total Theory: 45 Periods

List of Indicative Experiments **Total Practical: 30 Periods**

1. Create a database table, add constraints (primary key, unique, check, Not null), insert rows, update and delete rows using SQL DDL and DML commands.
2. Create a set of tables, add foreign key constraints and incorporate referential integrity.
3. Query the database tables using different 'where' clause conditions and also implement aggregate functions.
4. Query the database tables and explore sub queries and simple join operations,natural, equi and outer joins.
5. Write user defined functions and stored procedures in SQL.
6. Execute complex transactions and realize DCL and TCL commands.
7. Write SQL Triggers for insert, delete, and update operations in a database table.
8. Create View and index for database tables with Kaggle dataset.
9. Create Document using NOSQL database tools.
10. Case Study using **any one** of the real life database applications from the following list a) Inventory Management for a EMart Grocery Shop b) Society Financial Management c) Cop Friendly App – Eseva d) Property Management – eMall e) Star Small and Medium Banking and Finance

    Repository Dataset to be used

	Text Books	**References**
Learning Resources	1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Seventh Edition, McGraw Hill, 2020. 2. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education, 2021	1. C.J.Date, A.Kannan, S.Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.

Learning Assessment

Continuous Learning Assessment (CLA) (50% weightage)				**End Semester Examination (50% weightage)**
Theory (25% weightage)		**Practical (25% weightage)**		
Average of Internal Test (15%)	**Critical Thinking Assessment (10%)**	**Evaluation of Laboratory Observation, Record (15%)**	**Model Lab Exam (10%)**	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%

Course Code	24CS401	Course Name	THEORY OF COMPUTATION	Course Category	T	Professional Core Courses	L	T	P	C
							3	0	0	3

Pre-requisite Courses	Discrete Mathematics	Co- requisite Courses	Nil	Progressive Courses	Compiler Design
Course Offering Department	Computer Science and Engineering	Data Book / Codes / Standards			

Course Objective: ***The purpose of learning this course is to:***

- Understand foundations of computation including automata theory
- Construct models of regular expressions and languages.
- Design context free grammar and push down automata
- Understand Turing machines and their capability
- Understand Undecidability and NP class problems

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Construct automata theory using Finite Automata	C	1	3	2	3	-	-	-	-	-	-	-	-	1	3	2
CO-2:	Write regular expressions for any pattern	U	2	2	3	2	-	-	-	-	-	-	-	-	3	1	2
CO-3:	Design context free grammar and Pushdown Automata	U	2	2	3	2	-	-	-	-	-	-	-	-	1	2	2
CO-4:	Design Turing machine for computational functions	AP	2	2	2	1	-	-	-	-	-	-	-	-	3	1	3
CO-5:	Differentiate between decidable and undecidable problems	R	2	2	2	2	1	-	-	-	-	-	-	-	2	2	2

Unit-1	**AUTOMATA AND REGULAR EXPRESSIONS**	***9 Periods***

Automata theory - Finite Automata (FA) - Deterministic Finite Automata (DFA) - Non-deterministic Finite Automata (NFA) - Finite Automata with Epsilon transitions - Equivalence of NFA and DFA- Conversion of NFA into DFA – Minimization of DFAs.

Unit-2	**REGULAR EXPRESSIONS AND LANGUAGES**	***9 Periods***

Regular expression – Regular Languages- Equivalence of Finite Automata and regular expressions – Pumping Lemma – Closure properties .

Unit-3	**CONTEXT FREE GRAMMAR AND PUSH DOWN AUTOMATA**	***9 Periods***

Types of Grammar - Chomsky's hierarchy of languages -Context-Free Grammar (CFG) and Languages - Ambiguity- Push Down Automata (PDA)- Instantaneous descriptions- Languages of pushdown automata - Equivalence of pushdown automata and CFG-CFG to PDA-PDA to CFG - Deterministic Pushdown Automata.

Unit-4	**NORMAL FORMS AND TURING MACHINES**	***9 Periods***

Normal forms for CFG - Simplification of CFG- Chomsky Normal Form (CNF) and Greibach Normal Form (GNF) - Pumping lemma for CFL - Closure properties of Context Free Languages -Turing Machine- Language acceptance by TM – TM as Computer of Integer functions – Programming techniques for Turing machines. Case study-Turing Machine.

Unit-5	**UNDECIDABILITY**	***9 Periods***

Unsolvable Problems and Computable Functions -PCP-MPCP- Recursive and recursively enumerable languages - Properties - Universal Turing machine -Tractable and Intractable problems - P and NP completeness – Kruskal's algorithm – Travelling Salesman Problem- 3-CNF SAT problems.Applications of NLP in Robotics.

Total: 45 Periods

	Text Books	References
Learning Resources	1. Hopcroft J.E., Motwani R. & Ullman J.D., "Introduction to Automata Theory, Languages and Computations", 3rd Edition, Pearson Education, 2020. 2. John C Martin , "Introduction to Languages and the Theory of Computation", 4th Edition, Tata McGraw Hill, 2011.	1. Harry R Lewis and Christos H Papadimitriou , "Elements of the Theory of Computation", 2nd Edition, Prentice Hall of India, 2015. 2. Peter Linz, "An Introduction to Formal Language and Automata", 6th Edition, Jones & Bartlett, 2016. 3.K.L.P.Mishra and N.Chandrasekaran, “Theory of Computer Science: Automata Languages and Computation”, 3rd Edition, Prentice Hall of India, 2006.

Learning Assessment		
Continuous Learning Assessment (CLA) (40% weightage)		**End Semester Examination (60% weightage)**
Average of Internal Test (20%)	***Critical Thinking Assessment (20%)***	
***** The expected levels for Bloom’s Taxonomy should be: - Lower-order thinking skills: not more than 40% - Higher-order thinking skills: not less than 60%		

Course Code	24CS402	Course Name	ALGORITHMS	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Data Structures	Co- requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	Nil		

Course Objective: ***The purpose of learning this course is to:***

- Understand and apply the algorithm analysis techniques on searching and sorting algorithms for real world problems.
- Critically analyze the efficiency of graph algorithms
- Understand different algorithm design techniques
- Solve programming problems using state space tree
- Understand the concepts behind NP Completeness, Approximation algorithms and randomized algorithms.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Analyze the efficiency of algorithms using various frameworks	AP	2	3	1	2	2	1	1	1	1	1	1	1	2	1	3
CO-2:	Solve problems using algorithm design techniques like divide and conquer, dynamic programming and greedy techniques.	AP	2	2	1	2	2	1	1	1	1	1	1	1	2	2	2
CO-3:	Apply dynamic programming and graph algorithms to solve problems and analyze their efficiency.	AP	2	3	1	2	3	1	1	1	1	1	1	1	2	1	2
CO-4:	Implement the state space tree method and string matching algorithm for solving problems.	AN	3	2	1	1	3	2	2	2	2	2	2	2	2	3	1
CO-5:	Analyse approximation algorithms for real world applications.	AN	3	2	1	2	3	2	2	2	2	2	2	2	2	2	3

Unit-1 INTRODUCTION **9 Periods**

Time and space complexity - Asymptotic Notations and its properties- Recurrence relation- substitution method Recurrence tree method-Master's method- Best case, Worst case and average case analysis for linear search, binary search and Interpolation Search, Best case, Worst case and average case analysis for Insertion sort.

Unit-2 DIVIDE AND CONQUER AND GREEDY ALGORITHMS **9 Periods**

Divide and Conquer methodology–Analysis of Merge sort – Analysis of Quick sort -Finding maximum and minimum - Greedy Technique-Elements of the greedy strategy – Activity selection problem – Optimal Merge pattern – Huffman Trees. Randomized Algorithms- Randomized quick sort - Finding kth smallest number.

Unit-3 DYNAMIC PROGRAMMING AND GRAPH ALGORITHMS **9 Periods**

Dynamic programming- Elements of dynamic programming — Matrix-chain multiplication -Longest common search. - Shortest path: Bellman-Ford algorithm - Floyd-Warshall algorithm Network flow: Flow networks - Ford-Fulkerson method

Unit-4 STATE SPACE AND STRING MATCHING ALGORITHMS **9 Periods**

Backtracking: n-Queens problem - Hamiltonian Circuit Problem - Branch and Bound: Knapsack Problem - Travelling Salesman Problem -String Matching- The naïve string matching algorithm - Knuth-Morris-Pratt algorithm.

Unit-5	**NP-COMPLETE AND APPROXIMATION ALGORITHMS** **9 Periods**
Tractable and intractable problems: Polynomial time algorithms –- NP-algorithms - NP-hardness and NP-completeness -Problem reduction-CNF-3-CNF problem- TSP. Approximation Algorithms: TSP-The Vertex Cover problem	
	Total Theory: 45 Periods

List of Indicative Experiments	**Total Practical: 30 Periods**
1. Develop a program to analyse Linear and Binary search algorithms. 2. Develop a program to analyse insertion sort 3. Develop a program to analyse merge sort 4. Develop a program to optimize the cost in merging multiple sorted list using Greedy algorithm 5. Implement randomized algorithms for finding the kth smallest number. 6. Implement Floyd's algorithm for the All-Pairs- Shortest-Paths problem. 7. Compute the transitive closure of a given directed graph using Warshall's algorithm. 8. Develop a program to find out the maximum and minimum numbers in a given list of n numbers using the divide and conquer technique. 9. Implement N Queens problem using Backtracking. 10. 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 Books	**References**
Learning Resources	1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 4th Edition, Prentice Hall of India, 2022. 2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran "Computer Algorithms/C++" Orient Blackswan, 2nd Edition, 2019.	1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson Education, 2012. 2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Reprint Edition, Pearson Education, 2006. 3. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press, 2014

Learning Assessment				
Continuous Learning Assessment (CLA) (50% weightage)				**End Semester Examination (50% weightage)**
Theory (25% weightage)		**Practical (25% weightage)**		
Average of Internal Test (15%)	**Critical Thinking Assessment (10%)**	**Evaluation of Laboratory Observation, Record (15%)**	**Model Lab Exam (10%)**	
* The expected levels for Bloom's Taxonomy should be: - Lower-order thinking skills: not more than 40% - Higher-order thinking skills: not less than 60%				

Course Code	24CS403	Course Name	INTRODUCTION TO OPERATING SYSTEMS	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Programming in C	Co- requisite Courses	NIL	Progressive Courses	NIL
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	NIL		

Course Objective: ***The purpose of learning this course is to:***

- Understand the basics and functions of operating systems.
- Understand processes and threads.
- Analyze scheduling algorithms and process synchronization.
- Understand the concept of deadlocks.
- Analyze various memory management schemes.

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Illustrate the concepts of operating systems.	U	3	1	2	2	-	-	-	-	-	-	-	-	1	2	2
CO-2:	Analyze various scheduling algorithms and process synchronization and explain deadlock prevention and avoidance algorithms.	AP	2	2	3	1	1	-	-	-	-	-	-	-	2	1	2
CO-3:	Compare and contrast various memory management schemes.	U	1	3	2	2	1	-	-	-	1	1	1	1	1	2	2
CO-4:	Explain the storage management and functionality of file systems, I/O systems.	U	1	3	3	3	2	2	2	2	2	2	2	2	1	3	2
CO-5:	Apply virtualization to OS, implement VMs, and analyze mobile OS .	AP	3	1	2	1	2	2	2	2	2	2	2	2	1	2	2

Unit-1 INTRODUCTION **7 Periods**

Operating System concepts - Evolution - Structures -Services - User Operating System Interface - System Calls - System Programs - Design and Implementation - Structuring methods.

Unit-2 PROCESS MANAGEMENT **11 Periods**

Process Concept - Process Scheduling - Operations on Processes - Inter-process Communication; CPU Scheduling - Scheduling criteria - Scheduling algorithms: Threads - Multi Thread Models - Threading issues; Process Synchronization - The Critical-Section problem - Synchronization hardware - Semaphores - Mutex - Classical problems of synchronization - Monitors; Deadlock - Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock.

Unit-3 MEMORY MANAGEMENT **10 Periods**

Main Memory - Swapping - Contiguous Memory Allocation - Paging - Structure of the Page Table - Segmentation, Segmentation with paging; Virtual Memory - Demand Paging – Copy on Write - Page Replacement - Allocation of Frames -Thrashing – working set model.

Unit-4 STORAGE MANAGEMENT **10 Periods**

Mass Storage system – Disk Structure - Disk Scheduling and Management; File-System Interface - File concept - Access methods - Directory Structure - Directory organization - File system mounting - File Sharing and Protection- File System Implementation - File System Structure - Directory implementation - Allocation Methods - Free Space Management; I/O Systems - I/O Hardware- Application I/O interface- Kernel I/O subsystem -Storage structure-Linux File system

Unit-5 VIRTUAL MACHINES AND MOBILE OS	7 Periods
Virtual Machines – History, Benefits and Features, Building Blocks, Types of Virtual Machines and their Implementations, Virtualization and Operating-System Components; Mobile OS - iOS and Android.	
	Total Theory: 45 Periods

List of Indicative Experiments	**Total Practical: 30 Periods**

1. Implementation of Unix System Calls
2. Simulation and Analysis of Non Preemptive and Preemptive CPU Scheduling Algorithms
3. i. Simulation of Producer – Consumer Problem using Semaphores
   ii. Implementation of Dining Philosophers Problem to demonstrate Process Synchronization
4. Simulation of Banker's Algorithm for Deadlock Avoidance
5. Analysis and Simulation of Memory Allocation and Management Techniques
6. Implementation of Page Replacement Techniques
7. Simulation of Disk Scheduling Algorithms
8. Implementation of File organization Techniques
9. Case study -analysis of any OS

	Text Books	**References**
Learning Resources	1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts" 10th Edition, John Wiley and Sons Inc., 2018. 2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.	1. Ramaz Elmasri, A. Gil Carrick, David Levine, " Operating Systems - A Spiral Approach", Tata McGraw Hill Edition, 2010. 2. William Stallings, "Operating Systems: Internals and Design Principles",7th Edition, Prentice Hall, 2018. 3. Achyut S.Godbole, Atul Kahate, "Operating Systems", McGraw Hill Education, 2016.

Learning Assessment

Continuous Learning Assessment (CLA) (50% weightage)				**End Semester Examination (50% weightage)**
Theory (25% weightage)		**Practical (25% weightage)**		
Average of Internal Test (15%)	**Critical Thinking Assessment (10%)**	**Evaluation of Laboratory Observation, Record (15%)**	**Model Lab Exam (10%)**	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%

Course Code	24CS404	Course Name	OBJECT ORIENTED SOFTWARE ENGINEERING	Course Category	TP	Professional Core Courses	L	T	P	C
							3	0	2	4

Pre-requisite Courses	OBJECT ORIENTED PROGRAMMING LANGUAGE	Co- requisite Courses		Progressive Courses	SOFTWARE TESTING AND AUTOMATION
Course Offering Department	Computer Science and Engineering	Data Book / Codes Standards	NIL		

Course Objective: ***The purpose of learning this course is to:***

- Understand Software Engineering Lifecycle Models
- Perform software requirements analysis
- Gain knowledge of the System Analysis and Design concepts using UML.
- Understand software testing and maintenance approaches
- Work on project management scheduling using DevOps

			Program Outcomes (PO)												Program Specific Outcome (PSO)		
Course Outcomes (CO): ***At the end of this course, learners will be able to:***		**BL**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**	**3**
CO-1:	Compare various Software Development Life cycle Models	U	2	2	1	2	2	-	-	-	-	1	1	2	2	2	1
CO-2:	Evaluate project management approaches as well as cost and schedule estimation strategies.	U	2	3	2	3	2	-	-	-	-	2	1	2	3	2	1
CO-3:	Perform formal analysis on specifications.	U	2	3	2	1	1	-	-	-	-	2	1	2	2	3	1
CO-4:	Use UML diagrams for analysis and design.	AP	2	3	2	2	3	-	-	-	-	2	1	2	2	3	1
CO-5:	Architect and design using architectural styles and design patterns, and test the system	AN	2	1	1	2	2	2	2	2	2	2	2	2	3	2	2

Unit-1 SOFTWARE PROCESS AND AGILE DEVELOPMENT **9 Periods**

Introduction to Software Process-Software Life cycle model-Process Models-Agile Process-Extreme programming-XP Process.

Unit-2 REQUIREMENTS ANALYSIS AND SPECIFICATION **9 Periods**

Requirement analysis specification – Software Requirement Specification – Formal system specification – Finite State Machines – Petri Nets – Object modelling using UML- Use case Model – Class diagrams – Interaction diagrams – Activity Diagrams-State chart Diagrams-Functional Modelling-Data Flow Diagram.

Unit-3 SOFTWARE DESIGN **9 Periods**

Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade -Architectural styles – Layered - Client Server - Tiered - Pipe and filter- User interface design.

Unit-4 SOFTWARE TESTING AND MAINTENANCE **9 Periods**

Testing-Unit Testing-Black box testing-White box testing-Integration and System testing- Regression Testing-Debugging-Program Analysis-Symbolic Execution- Model Checking.

Unit-5 PROJECT MANAGEMENT **9 Periods**

Software Project Management- Software Configuration Management - Project Scheduling- DevOps: Motivation-Cloud as a platform-Operations- Deployment Pipeline: Overall Architecture Building and Testing-Deployment- Tools.

Total Theory: 45 Periods

List of Indicative Experiments	Total Practical: 30 Periods

1. Identify the software system requirements.
2. Document the Software Requirements Specification(SRS).
3. Identify use cases and develop the Use Case model.
4. Identify the conceptual classes and develop a Domain Model and also derive a Class Diagram from that.
5. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams
6. Draw relevant Statechart and Activity Diagrams .
7. Implement the system.
8. Test the software system for all the scenarios identified as per the use case diagram.
9. Improve the reusability and maintainability of the software system by applying appropriate design patterns.
10. Implement the Modified System and Test it for various scenarios.

Choice of Applications

1. Passport automation system. 2. Book bank 3. Exam registration 4. Stock maintenance system. 5. Online course reservation system 6. Airline/Railway reservation system 7. Software personnel management system 8. Credit card processing 9. e-book management system 10. Recruitment system 11. Foreign trading system 12. Conference management system 13. BPO management system 14. Library management system 15. Student information system

	Text Books	References
Learning Resources	1. Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineering: Using UML, Patterns and Java", Third Edition, Pearson Education, 2009. 2. RogerS.Pressman,Object-Oriented Software Engineering: An Agile Unified Methodology, First Edition, Mc Graw-Hill International Edition, 2014.	1. Carlo Ghezzi,Mehdi Jazayeri,Dino Mandrioli, Fundamentals Software Engineering,2nd edition, PHI Learning Pvt. Ltd., 2010. 2. Craig Larman, Applying UML and Patterns,3rd ed,PearsonEducation,2005. 3.LenBass,Ingo Weber andLiming Zhu,"DevOps: A Software Architect's Perspective", Pearson Education, 2016 4. Rajib Mall, Fundamentals of Software Engineering, 3rd edition, PHI Learning Pvt. Ltd.,2009. 5. Stephen Schach,Object-Oriented and Classical Software Engineering,8th, McGraw-Hill, 2010.

Learning Assessment				
Continuous Learning Assessment (CLA) (50% weightage)				End Semester Examination (50% weightage)
Theory (25% weightage)		Practical (25% weightage)		
Average of Internal Test (15%)	Critical Thinking Assessment (10%)	Evaluation of Laboratory Observation, Record (15%)	Model Lab Exam (10%)	

* The expected levels for Bloom's Taxonomy should be:
- Lower-order thinking skills: not more than 40%
- Higher-order thinking skills: not less than 60%