R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

REGULATIONS–2022
CHOICE BASED CREDIT SYSTEM
B.E. COMPUTER SCIENCE AND ENGINEERING

(For the students admitted in the Academic year 2023-2024)

PROGRAM EDUCATIONAL OBJECTIVES

The Computer Science and Engineering Graduates of R.M.D. Engineering College will:

PEO 1

To enable graduates to pursue higher education and research, or have a successful career in industries associated with Computer Science and Engineering, or as entrepreneurs.

PEO 2

To ensure that graduates will have the ability and attitude to adapt to emerging technological changes.

PEO 3

To enable graduates to value social responsibility and possess strong leadership and interpersonal abilities**.**

PROGRAMME OUTCOMES (POs)

On successful completion of the programme, Engineering Graduates will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needswith appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation ofdata, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

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.
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.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
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.
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.
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 oftechnological change.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

After the successful completion of the program, the graduates will be able to:

PSO 1

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems.

Mapping of POs/PSOs to PEOs

Contribution 1: Reasonable 2: Significant 3: Strong

PEOs & POs

The B.E. Computer Science and Engineering program outcomes leading to the achievement of theobjectives are summarized in the following table.

PROGRAM EDUCATIONAL OBJECTIVES	PROGRAM OUTCOMES											
		PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10		PO 12
I		3	3	3	2	2	2	3	3	3		1
II		3	3	3	2	2	2	1	1	1		1
III	2	2	2	2	2	3	2	3	3	1	1	1

PROGRAM EDUCATIONAL OBJECTIVES	PROGRAM SPECIFIC OUTCOMES		
	PSO 1	PSO 2	PSO 3
I	3	3	3
II	3	3	3
III	1	1	1

MAPPING OF COURSE OUTCOMES WITH PROGRAMME OUTCOMES

A broad relation between the Course Outcomes and Programme Outcomes is given in the following table.

Contribution 1: Reasonable 2: Significant 3: Strong

YEAR	SEMESTER	COURSE NAME	Programme Outcome (PO)											
			1	2	3	4	5	6	7	8	9	10	11	12
YEAR I	SEMESTER I	Heritage of Tamils												
		Matrices and Calculus	✓	✓	✓	✓	✓	✓	✓					✓
		Engineering Chemistry	✓	✓				✓	✓					✓
		Problem Solving using C++	✓	✓	✓					✓	✓	✓		✓
		Software Development Practices	✓	✓	✓		✓	✓		✓	✓	✓		✓
		Digital Principles and System Design	✓	✓	✓					✓	✓	✓		✓
		Computer aided Engineering Graphics	✓		✓		✓					✓		
		Product Development Lab-1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Induction Program						✓	✓	✓	✓	✓	✓	✓
	SEMESTER II	Tamils and Technology												
		Transforms and Numerical Methods	✓	✓	✓	✓	✓	✓						✓
		Data Structures	✓	✓	✓					✓	✓	✓		✓
		Physics for Computer Science and Information Technology	✓	✓	✓	✓								
		Professional Communication									✓	✓		✓
		Java Programming	✓	✓	✓		✓			✓	✓	✓		✓
		Database Management System	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Product Development Lab - 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Environmental Sciences and Sustainability (Non Credit)	✓	✓				✓	✓			✓		✓

		Yoga for Stress Management												
YEAR II	SEMESTER III	Discrete Mathematics	✓	✓	✓	✓								
		Universal Human Values 2: Understanding Harmony						✓	✓	✓	✓	✓	✓	✓
		Computer Organization and Architecture	✓	✓	✓	✓								
		Advanced Java Programming	✓	✓	✓	✓				✓	✓	✓		✓
		Design and Analysis of Algorithms	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Operating Systems	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Product Development Lab - 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Aptitude and Coding Skills I	✓	✓	✓						✓	✓		
		Internship and Seminar	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Value Education (Non-Credit)												
	SEMESTER IV	Microcontrollers and Embedded Systems	✓	✓	✓	✓	✓							
		Probability and Statistics	✓	✓	✓	✓				✓	✓	✓		✓
		Distributed and Cloud Computing	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Web Development Frameworks	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Artificial Intelligence	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Professional Elective I												
		Aptitude and Coding Skills II	✓	✓	✓						✓	✓		
		Product Development Lab - 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Yoga/Personality Development (Non-Credit)												

YEAR III	SEMESTER V	Computer Networks	✓	✓	✓		✓			✓	✓	✓		✓
		Theory of Computation	✓	✓	✓		✓			✓	✓	✓		✓
		Machine Learning	✓	✓	✓		✓			✓	✓	✓		✓
		Professional Elective II												
		Professional Elective III												
		Open Elective I												
		Advanced Aptitude and Coding Skills I	✓	✓	✓						✓	✓		
		Internship/Seminar	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Indian Constitution (Non Credit)	✓	✓							✓	✓		
	SEMESTER VI	Professional Ethics						✓	✓	✓	✓	✓	✓	✓
		Compiler Design	✓	✓	✓		✓			✓	✓	✓		✓
		Object Oriented Software Engineering	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Professional Elective IV												
		Professional Elective V												
		Open Elective II												
		Advanced Aptitude and Coding Skills II	✓	✓							✓	✓		
	SEMESTER VII	Cryptography and Cyber Security	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Data Analytics	✓	✓	✓	✓	✓			✓	✓	✓		✓
		Professional Elective VI												
		Open Elective III												
		Open Elective IV												
		Essence of Indian Knowledge Tradition (Non Credit)	✓	✓							✓	✓		

	SEMESTER VIII	Project Work	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

PROFESSIONAL ELECTIVES

VERTICALS	**COURSE NAME**	**PROGRAM OUTCOMES**											
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
CYBER SECURITY	Ethical Hacking	✓	✓	✓		✓			✓	✓	✓		✓
	Social Network Security	✓	✓	✓		✓			✓	✓	✓		✓
	Blockchain Technology	✓	✓	✓		✓			✓	✓	✓		✓
	Cloud and Data Security	✓	✓	✓		✓			✓	✓	✓		✓
	Digital and Mobile Forensics	✓	✓	✓		✓			✓	✓	✓		✓
	Vulnerability Analysis and Penetration Testing	✓	✓	✓		✓			✓	✓	✓		✓
	Engineering Secure Software Systems	✓	✓	✓		✓			✓	✓	✓		✓
	Network Design and Programming	✓	✓	✓		✓			✓	✓	✓		✓
	Fault Tolerant Computing	✓	✓	✓		✓			✓	✓	✓		✓
	Enterprise Cyber Security	✓	✓	✓		✓			✓	✓	✓		✓
	Rest Application Development Using Spring Boot and JPA	✓	✓	✓		✓			✓	✓	✓		✓
	Capstone Project	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
CLOUD COMPUTING	Cloud Foundations	✓	✓	✓		✓			✓	✓	✓		✓
	Cloud Architecting	✓	✓	✓		✓			✓	✓	✓		✓
	Virtualization	✓	✓	✓		✓			✓	✓	✓		✓
	DevOps	✓	✓	✓		✓			✓	✓	✓		✓
	Data Engineering in Cloud	✓	✓	✓		✓			✓	✓	✓		✓

	Cloud Security Foundations	✓	✓	✓		✓			✓	✓	✓		✓
	Software Defined Networks	✓	✓	✓		✓			✓	✓	✓		✓
	Storage Technologies	✓	✓	✓		✓			✓	✓	✓		✓
	Machine Learning for NLP in Cloud	✓	✓	✓		✓			✓	✓	✓		✓
	Cloud Services Management	✓	✓	✓		✓			✓	✓	✓		✓
	Rest Application Development Using Spring Boot and JPA	✓	✓	✓		✓			✓	✓	✓		✓
	Capstone Project	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
FULL STACK TECHNOLOGY	UI/UX Design	✓	✓	✓	✓	✓			✓	✓	✓		✓
	MERN Stack Development	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Mobile Architecture and Application Development	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Micro Service Architecture	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Web Application Security	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Blockchain Technology	✓	✓	✓		✓			✓	✓	✓		✓
	Software Testing and Automation	✓	✓	✓	✓	✓			✓	✓	✓		✓
	DevOps	✓	✓	✓		✓			✓	✓	✓		✓
	Usability Design of Software Application	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Generative AI Fundamentals	✓	✓	✓		✓			✓	✓	✓		✓
	Rest Application Development Using Spring Boot and JPA	✓	✓	✓		✓			✓	✓	✓		✓
	Capstone Project	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
DATA SCIENCE	Data Science using Python	✓	✓	✓	✓	✓			✓	✓	✓		✓
	Data Exploration and Visualization	✓	✓	✓		✓			✓	✓	✓		✓
	Text and Speech Analytics	✓	✓	✓					✓	✓	✓		✓
	Image and Video Analytics	✓	✓	✓					✓	✓	✓		✓
	Stream Processing and Analytics	✓	✓	✓		✓			✓	✓	✓		✓

	Cognitive Science and Analytics	✓	✓	✓		✓			✓	✓	✓		✓
	Capstone Project	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ARTIFICIAL INTELLIGENCE	Soft Computing	✓	✓	✓		✓			✓	✓	✓		✓
	Neural Networks and Deep Learning	✓	✓	✓					✓	✓	✓		✓
	Reinforcement and Ensemble Learning	✓	✓	✓					✓	✓	✓		✓
	Applied AI and ML	✓	✓	✓					✓	✓	✓		✓
	Recommender Systems	✓	✓	✓					✓	✓	✓		✓
	Generative AI Fundamentals	✓	✓	✓		✓			✓	✓	✓		✓
	Capstone Project	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EMERGING TECHNOLOGIES	Industrial IoT	✓	✓	✓		✓			✓	✓	✓		✓
	GPU Computing	✓	✓	✓		✓			✓	✓	✓		✓
	Introduction to Augmented and Virtual Reality	✓	✓	✓		✓			✓	✓	✓		✓
	Digital Marketing	✓	✓	✓		✓			✓	✓	✓		✓
	Quantum Computing	✓	✓	✓		✓			✓	✓	✓		✓
	Intelligent Robots	✓	✓	✓		✓			✓	✓	✓		✓
	Game Development	✓	✓	✓		✓			✓	✓	✓		✓
	Principles of 3D Printing and Design	✓	✓	✓		✓			✓	✓	✓		✓
	Computer Vision	✓	✓	✓		✓			✓	✓	✓		✓

R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

B.E. - COMPUTER SCIENCE AND ENGINEERING

REGULATIONS – 2022

CHOICE BASED CREDIT SYSTEM

I - VIII SEMESTER CURRICULUM AND SYLLABI

(For the Students admitted in the Academic Year 2023-24)

SEMESTER – I								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	22GE101	Heritage of Tamils	HSMC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
2	22MA101	Matrices and Calculus	BSC	5	3	0	2	4
3	22CH101	Engineering Chemistry	BSC	5	3	0	2	4
3	22CS101	Problem Solving using C++	ESC	5	3	0	2	4
4	22CS102	Software Development Practices	ESC	5	3	0	2	4
5	22EC101	Digital Principles and System Design	ESC	5	3	0	2	4
LABORATORY COURSES WITH THEORY COMPONENT								
6	2GE112	Computer Aided Engineering Graphics	ESC	3	1	0	2	2
LABORATORY COURSES								
7	22GE111	Product Development Lab-1	EEC	2	0	0	2	1
MANDATORY COURSES								
8		Induction Program (Non Credit)	MC	3 Weeks				
			TOTAL	**31**	**17**	**0**	**14**	**24**

SEMESTER – II								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	22GE201	Tamils and Technology	HSMC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
2	22MA201	Transforms and Numerical Methods	BSC	5	3	0	2	4
3	22CS201	Data Structures	ESC	5	3	0	2	4
4	22PH201	Physics for Computer Science and Information Technology	BSC	5	3	0	2	4
5	22HS101	Professional Communication	HSMC	4	2	0	2	3
6	22CS202	Java Programming	ESC	5	3	0	2	4
7	22IT202	Database Management System	PCC	5	3	0	2	4
LABORATORY COURSES								
8	22GE211	Product Development Lab - 2	EEC	2	0	0	2	1
MANDATORY COURSES								
9	22CH104	Environmental Sciences and Sustainability (Non Credit)	MC	2	2	0	0	0
AUDIT COURSES								
10	22AC201	Yoga for Stress Management	AC	1	1	0	0	0
			TOTAL	**35**	**21**	**0**	**14**	**25**

SEMESTER – III								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	22MA301	Discrete Mathematics	BSC	4	3	1	0	4
2	22GE301	Universal Human Values II: Understanding Harmony	HSMC	3	3	0	0	3
3	22CS302	Computer Organization and Architecture	ESC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4	22CS307	Advanced Java Programming	PCC	5	3	0	2	4
5	22CS306	Design and Analysis of Algorithms	PCC	5	3	0	2	4
6	22CS304	Operating Systems	PCC	4	2	0	2	3
LABORATORY COURSES								
7	22GE311	Product Development Lab - 3	EEC	2	0	0	2	1
EMPLOYABILITY ENHANCEMENT COURSES								
8	22CS311	Aptitude and Coding Skills I	EEC	2	0	0	2	1
9	22CS312	Internship and Seminar	EEC	2	0	0	2	1
AUDIT COURSES								
10	22AC301	Value Education (Non Credit)	AC	1	1	0	0	0
			TOTAL	**31**	**18**	**1**	**12**	**24**

*2 weeks for one credit. Internship during 4 Semester Summer Vacation

SEMESTER – IV								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	22EC441	Microcontrollers and Embedded Systems	ESC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
2	22MA401	Probability and Statistics	BSC	5	3	0	2	4
3	22CS401	Distributed and Cloud Computing	PCC	4	2	0	2	3
4	22IT403	Web Development Frameworks	PCC	5	3	0	2	4
5	22AM301	Artificial Intelligence	PCC	5	3	0	2	4
6		Professional Elective I	PEC	4	2	0	2	3
LABORATORY COURSES								
7	22GE411	Product Development Lab - 4	EEC	2	0	0	2	1
EMPLOYABILITY ENHANCEMENT COURSES								
8	22CS411	Aptitude and Coding Skills II	EEC	2	0	0	2	1
AUDIT COURSES								
9	22AC401	Yoga/Personality Development (Non Credit)	AC	1	1	0	0	0
			TOTAL	**31**	**17**	**0**	**14**	**23**

SEMESTER – V								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1		Professional Elective II	PEC	3	3	0	0	3
2		Open Elective I	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3	22CS501	Computer Networks	PCC	5	3	0	2	4
4	22CS502	Theory of Computation	PCC	4	2	0	2	3
5	22CS503	Machine Learning	PCC	5	3	0	2	4
6		Professional Elective III	PEC	4	2	0	2	3
EMPLOYABILITY ENHANCEMENT COURSES								
7	22CS511	Advanced Aptitude and Coding Skills I	EEC	2	0	0	2	1
8	22CS512	Internship/Seminar	EEC	2	0	0	2	1
AUDIT COURSES								
8		Indian Constitution (Non Credit)	MC	1	1	0	0	0
			TOTAL	**29**	**17**	**0**	**12**	**22**

*2 weeks for one credit. Internship during 4th Semester Summer Vacation

SEMESTER – VI								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	22CS603	Professional Ethics	HSMC	3	3	0	0	3
2		Professional Elective IV	PEC	3	3	0	0	3
3		Professional Elective V	PEC	3	3	0	0	3
4		Open Elective II	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
5	22CS601	Compiler Design	PCC	5	3	0	2	4
6	22CS602	Object Oriented Software Engineering	PCC	4	2	0	2	3
EMPLOYABILITY ENHANCEMENT COURSES								
7	22CS611	Advanced Aptitude and Coding Skills II	EEC	2	0	0	2	1
			TOTAL	**23**	**17**	**0**	**6**	**20**

SEMESTER – VII								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1		Professional Elective VI	PEC	3	3	0	0	3
2		Open Elective III	OEC	3	3	0	0	3
3		Open Elective IV	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4	22CS701	Cryptography and Cyber Security	PCC	5	3	0	2	4
5	22CS702	Data Analytics	PCC	5	3	0	2	4
MANDATORY COURSES								
6		Essence of Indian Knowledge Tradition (Non Credit)	MC	1	1	0	0	0
			TOTAL	**20**	**16**	**0**	**4**	**17**

SEMESTER – VIII								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
EMPLOYABILITY ENHANCEMENT COURSES								
1	22CS811	Project Work	EEC	16	0	0	16	8
			TOTAL	**16**	**0**	**0**	**16**	**8**

Credit Summary

S. No.	Subject Area	Credits Per Semester								Credit Total	Percentage
		I	II	III	IV	V	VI	VII	VIII		
1	HSMC	1	4	3			3			11	6.75%
2	BSC	8	8	4	4					24	14.72%
3	ESC	14	8	3	3					28	17.18%
4	PCC		4	11	11	11	7	8		52	31.90%
5	PEC				3	6	6	3		18	11.04%
6	OEC					3	3	6		12	7.36%
7	EEC	1	1	3	2	2	1		8	18	11.04%
8	MC/AC										
Total		24	25	24	23	22	20	17	8	163	

HSMC - Humanities and Social Sciences including Management Course; BSC – Basic Science Course; ESC – Engineering Science Course; PCC – Professional Core Course; PEC – Professional Elective Course; OEC – Open Elective Course; EEC – Employability Enhancement Course; MC – Mandatory Course; AC – Audit Course.

PROFESSIONAL ELECTIVE COURSES: VERTICALS

Cyber Security	Cloud Computing	Full Stack Technology	Data Science	Artificial Intelligence	Emerging Technologies
Ethical Hacking	Cloud Foundations	UI/UX Design	Data Science using Python*	Soft Computing*	Industrial IoT
Social Network Security	Cloud Architecting	MERN Stack Development	Data Exploration and Visualization*	Neural Networks and Deep Learning*	GPU Computing
Blockchain Technology	Virtualization	Mobile Architecture and	Text and Speech Analytics*	Reinforcement and Ensemble Learning*	Introduction to Augmented and Virtual Reality

		Application Development			
Cloud and Data Security	DevOps	Microservice Architecture	Image and Video Analytics*	Applied AI and ML*	Digital Marketing
Digital and Mobile Forensics	Data Engineering in Cloud	Web Application Security	Stream Processing and Analytics*	Recommender Systems*	Quantum Computing
Vulnerability Analysis and Penetration Testing	Cloud Security Foundations	Blockchain Technology	Cognitive Science and Analytics*	Generative AI Fundamentals*	Intelligent Robots
Engineering Secure software systems*	Software Defined Networks*	Software Testing and Automation*			Game Development
Network Design and Programming*	Storage Technologies*	DevOps *			Principles of 3D Printing and Design
Fault Tolerant Computing*	Machine Learning for NLP in Cloud*	Usability Design of Software Application*			Computer Vision
Enterprise Cyber Security*	Cloud Services Management*	Generative AI Fundamentals*			
Rest Application Development Using Spring Boot and JPA	Rest Application Development Using Spring Boot and JPA	Rest Application Development Using Spring Boot and JPA			
Capstone Project*	Capstone Project*	Capstone Project*	Capstone Project*	Capstone Project*	

* **Minimum of 2 advanced courses with Capstone Project for Honours Degree**

ELECTIVE VERTICALS

CYBER SECURITY								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
1	22CS901	Ethical Hacking	PEC	4	2	0	2	3
2	22CS902	Social Network Security	PEC	4	2	0	2	3
3	22CS903	Blockchain Technology	PEC	3	3	0	0	3
4	22CS904	Cloud and Data Security	PEC	3	3	0	0	3
5	22CS905	Digital and Mobile Forensics	PEC	3	3	0	0	3
6	22CS906	Vulnerability Analysis and Penetration Testing	PEC	3	3	0	0	3
7	22CS927	Engineering Secure software systems*	PEC	3	3	0	0	3
8	22CS928	Network Design and Programming*	PEC	3	3	0	0	3
9	22CS929	Fault Tolerant Computing*	PEC	3	3	0	0	3

10	22CS930	Enterprise Cyber Security*	PEC	3	3	0	0	3
11	22IT910	Rest Application Development Using Spring Boot and JPA	PEC	4	2	0	2	3
12	22CS812	Capstone Project*	EEC	12	0	0	12	6
CLOUD COMPUTING								
1	22CS907	Cloud Foundations	PEC	4	2	0	2	3
2	22CS908	Cloud Architecting	PEC	4	2	0	2	3
3	22CS909	Virtualization	PEC	3	3	0	0	3
4	22CS910	DevOps	PEC	3	3	0	0	3
5	22CS911	Data Engineering in Cloud	PEC	3	3	0	0	3
6	22CS912	Cloud Security Foundations	PEC	3	3	0	0	3
7	22CS931	Software Defined Networks*	PEC	3	3	0	0	3
8	22CS932	Storage Technologies*	PEC	3	3	0	0	3
9	22CS933	Machine Learning for NLP in Cloud*	PEC	3	3	0	0	3
10	22CS934	Cloud Services Management*	PEC	3	3	0	0	3
11	22IT910	Rest Application Development Using Spring Boot and JPA	PEC	4	2	0	2	3
12	22CS812	Capstone Project*	EEC	12	0	0	12	6
FULL STACK TECHNOLOGY								
1	22CS913	UI/UX Design	PEC	4	2	0	2	3
2	22CS914	MERN Stack Development	PEC	4	2	0	2	3
3	22CS915	Mobile Architecture and Application Development	PEC	4	2	0	2	3
4	22CS916	Microservice Architecture	PEC	3	3	0	0	3
5	22CS917	Web Application Security	PEC	3	3	0	0	3
6	22CS903	Blockchain Technology	PEC	3	3	0	0	3
7	22CS918	Software Testing and Automation*	PEC	3	3	0	0	3
8	22CS910	DevOps*	PEC	3	3	0	0	3
9	22CS919	Usability Design of Software Application*	PEC	3	3	0	0	3
10	22CS920	Generative AI Fundamentals*	PEC	3	3	0	0	3
11	22IT910	Rest Application Development Using Spring Boot and JPA	PEC	4	2	0	2	3
12	22CS812	Capstone Project*	EEC	12	0	0	12	6
DATA SCIENCE								
1	22AM901	Data Science using Python	PEC	4	2	0	2	3
2	22CS935	Data Exploration and Visualization	PEC	3	3	0	0	3
3	22AM904	Text and Speech Analytics	PEC	3	3	0	0	3
4	22AM905	Image and Video Analytics	PEC	3	3	0	0	3
5	22AM906	Stream Processing and	PEC	3	3	0	0	3

		Analytics						
6	22CS938	Cognitive Science and Analytics	PEC	3	3	0	0	3
7	22CS812	Capstone Project	EEC	12	0	0	12	6
ARTIFICIAL INTELLIGENCE								
1	22AM921	Soft Computing	PEC	4	2	0	2	3
2	22CS936	Neural Networks and Deep Learning	PEC	3	3	0	0	3
3	22CS937	Reinforcement and Ensemble Learning	PEC	3	3	0	0	3
4	22AM922	Applied AI and ML	PEC	3	3	0	0	3
5	22AM923	Recommender Systems	PEC	3	3	0	0	3
6	22CS920	Generative AI Fundamentals	PEC	3	3	0	0	3
11	22CS812	Capstone Project	EEC	12	0	0	12	6
EMERGING TECHNOLOGIES								
1	22CS921	Industrial IoT	PEC	3	3	0	0	3
2	22AM912	GPU Computing	PEC	3	3	0	0	3
3	22CS922	Introduction to Augmented and Virtual Reality	PEC	3	3	0	0	3
4	22CS923	Digital Marketing	PEC	3	3	0	0	3
5	22CS924	Quantum Computing	PEC	3	3	0	0	3
6	22AM909	Intelligent Robots	PEC	3	3	0	0	3
7	22CS925	Game Development	PEC	3	3	0	0	3
8	22CS926	Principles of 3D Printing and Design	PEC	3	3	0	0	3
9	22AM702	Computer Vision	PEC	3	3	0	0	3

OPEN ELECTIVE OFFERED TO OTHER DEPARTMENTS

1	22CS001	Ethical Hacking	OEC	3	3	0	0	3
2	22CS002	Cloud Foundations	OEC	3	3	0	0	3
3	22CS003	Digital and Mobile Forensics	OEC	3	3	0	0	3
4	22CS004	Blockchain Technology	OEC	3	3	0	0	3
5	22CS005	UI/UX Design	OEC	3	3	0	0	3
6	22CS006	Introduction to Computer Networks	OEC	3	3	0	0	3

R2022 CURRICULUM
B.E (HONOURS) IN COMPUTER SCIENCE AND ENGINEERING
WITH SPECIALIZATION IN

Sl. No.	NAME OF THE HONOURS DEGREE WITH SPECIALIZATION
1	Cyber Security
2	Cloud Computing

3	Full Stack Development
4	Data Science
5	Artificial Intelligence

Additional 18 credits to be completed from the courses offered in the specific Professional Elective Verticals

R2022 CURRICULUM

B. E. (HONOURS) IN COMPUTER SCIENCE AND ENGINEERING

Additional 18 credits to be completed from the courses offered in the ProfessionalElective Verticals.

B.E. CSE WITH MINOR DEGREE

Sl. No.	Name of the Minor Degree	Offering Department
1.	Internet of Things	Electronics and Communication Engineering

R2022 CURRICULUM

MINOR DEGREE CURRICULUM OFFERED BY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(FOR OTHER B.E. / B.Tech PROGRAMMES)

Sl. No.	NAME OF THE MINOR DEGREE WITH SPECIALIZATION
1	Cyber Security
2	Cloud Computing
3	Full Stack Development

CYBER SECURITY

1	22CS501	Computer Networks	PCC	4	2	0	2	3
2	22CS701	Cryptography and Cyber Security	PCC	4	2	0	2	3
3	22CS901	Ethical Hacking	PEC	4	2	0	2	3
4	22CS905	Digital and Mobile Forensics	PEC	3	3	0	0	3
5	22CS812	Capstone Project	EEC	12	0	0	12	6

CLOUD COMPUTING

1	22CS304	Operating Systems	PCC	4	2	0	2	3
2	22CS401	Distributed and Cloud Computing	PEC	4	2	0	2	3
3	22CS907	Cloud Foundations	PEC	4	2	0	2	3
4	22CS908	Cloud Architecting	PEC	4	2	0	2	3
5	22CS812	Capstone Project	EEC	12	0	0	12	6

FULL STACK DEVELOPMENT

1	22CS402	Web Development Frameworks	PCC	5	3	0	2	4
2	22CS914	MERN Stack Development	PEC	4	2	0	2	3
3	22CS915	Mobile Architecture and Application Development	PEC	4	2	0	2	3
4	22CS918	Web Application Security	PEC	3	3	0	0	3
5	22CS812	Capstone Project	EEC	12	0	0	12	6

SEMESTER – I

22GE101	HERITAGE OF TAMILS (Common to All Branches)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Recognize Tamil literature and its significance in Tamil culture.
- Introduce the Tamils' rich artistic and cultural legacy.
- Familiarize the different types of folk and martial arts that are unique to Tamil Nadu.
- Acquaint the concept of Thinai in Tamil literature and culture.
- Comprehend the significance of Tamil in developing Indian culture.

UNIT I	LANGUAGE AND LITERATURE	3

Language Families in India - Dravidian Languages – Tamil as a Classical Language - Classical Literature in Tamil – Secular Nature of Sangam Literature – Distributive Justice in Sangam Literature - Management Principles in Thirukural - Tamil Epics and Impact of Buddhism & Jainism in Tamil Land - Bakthi Literature Azhwars and Nayanmars - Forms of minor Poetry – Development of Modern literature in Tamil - Contribution of Bharathiyar and Bharathidhasan.

UNIT II	HERITAGE - ROCK ART PAINTINGS TO MODERN ART – SCULPTURE	3

Hero stone to modern sculpture - Bronze icons - Tribes and their handicrafts - Art of temple car making -- Massive Terracotta sculptures, Village deities, Thiruvalluvar Statue at Kanyakumari, Making of musical instruments - Mridhangam, Parai, Veenai, Yazh and Nadhaswaram - Role of Temples in Social and Economic Life of Tamils.

UNIT III	FOLK AND MARTIAL ARTS	3

Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leatherpuppetry, Silambattam, Valari, Tiger dance - Sports and Games of Tamils.

UNIT IV	THINAI CONCEPT OF TAMILS	15

Flora and Fauna of Tamils & Aham and Puram Concept from Tholkappiyam and Sangam Literature - Aram Concept of Tamils - Education and Literacy during Sangam Age - Ancient Cities and Ports of Sangam Age - Export and Import during Sangam Age - Overseas Conquest of Cholas.

UNIT V	CONTRIBUTION OF TAMILS TO INDIAN NATIONAL MOVEMENT AND INDIAN CULTURE	3

Contribution of Tamils to Indian Freedom Struggle – The Cultural Influence of Tamils over the other parts of India – Self-Respect Movement – Role of Siddha Medicine in Indigenous Systems of Medicine – Inscriptions & Manuscripts – Print History of Tamil Books.

TOTAL: 15 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: State the role of Tamil literature in shaping Tamil Cultural roots.

CO2: Express the cultural and religious significance of Tamil art and sculptures.

CO3: Identify and describe the techniques of folk and martial arts.
CO4: Classify the role of Thinai concept in Tamil culture and literature.
CO5: Compare the idea of cultural and intellectual contributions of Tamils.

TEXT BOOKS & REFERENCES:

1. தமிழக வரலாறு – மக்களும் பண்பாடும் – கே.கே. பிள்ளை (வெளியீடு: தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்).
2. கணினித தமிழ – முனைவர இல. சுநதரம. (விகடன பிரசுரம).
3. கீழடி – வைகை நதிககரையில சஙககால நகர நாகரிகம (தொலலியல துறை வெளியீடு)
4. பொருநை – ஆற்றங்கரை நாகரிகம். (தொல்லியல் துறை வெளியீடு)
5. Social Life of Tamils (Dr.K.K. Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
6. Social Life of the Tamils - The Classical Period (Dr.S .Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D.Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by: International Institute of Tamil Studies.)
9. Keeladi - ‗Sangam City Civilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
11. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
12. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL) – Reference Book

22MA101	MATRICES & CALCULUS (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Explain the concepts of matrix algebra techniques needed for practical applications.
- Determine the curvature of the curves.
- Illustrate the simple applications of multivariable calculus and vector calculus.
- Elaborate the concept and application of multiple integrals.

UNIT I	MATRICES	15

Eigenvalues and Eigenvectors of a real matrix – Properties of Eigenvalues and Eigenvectors – Statement and applications of Cayley-Hamilton Theorem – Diagonalization of matrices by orthogonal transformation – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms.

(Theory: 9)

Experiments using SCILAB:

1. Introduction to SCILAB through matrices and general syntax.
2. Finding the Eigenvalues and Eigenvectors.
3. Plotting the graph of a quadratic form.

(Laboratory: 6)

UNIT II	SINGLE VARIABLE CALCULUS	15

Curvature in Cartesian and Polar Co-ordinates – Centre and radius of curvature – Circle of curvature–Evolutes.

(Theory: 9)

Experiments using SCILAB:

1. Evaluating the radius of curvature.
2. Finding the coordinates of the center of curvature.
3. Tracing of Curves.

(Laboratory: 6)

UNIT III	MULTIVARIABLE CALCULUS	15

Partial derivatives (excluding Euler's theorem) – Total derivative – Differentiation of implicit functions – Jacobian and properties – Taylor's series for functions of two variables – Maxima and minima of functions of two variables.

(Theory: 9)

Experiments using SCILAB:

1. Evaluating the maxima of functions of several variables.
2. Evaluating the minima of functions of several variables.
3. Evaluation of Jacobians.

(Laboratory: 6)

UNIT IV	MULTIPLE INTEGRALS	15

Double integrals – Change of order of integration – Area enclosed by plane curves – Triple integrals – Volume of solids.

(Theory: 9)

Experiments using SCILAB:

1. Evaluating area under a curve.
2. Evaluating area using double integral.
3. Evaluation of volume by integrals.

(Laboratory: 6)

UNIT V	VECTOR CALCULUS	15

Gradient, divergence and curl (excluding vector identities) – Directional derivative – Irrotational and Solenoidal vector fields – Vector integration – Green's theorem in a plane and Gauss divergence theorem (Statement only) – Simple applications involving cubes and rectangular parallelopipeds.

(Theory: 9)

Experiments using SCILAB:

1. Evaluating gradient.
2. Evaluating directional derivative.
3. Evaluating divergent and curl.

(Laboratory: 6)

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Use the matrix algebra methods to diagonalize the matrix.

CO2: Determine the evolute of the curve.

CO3: Apply differential calculus ideas on the function of several variables.

CO4: Evaluate the area and volume by applying the concept of multiple integration.

CO5: Utilize the concept of vector calculus in evaluating integrals.

TEXT BOOKS:

1. Erwin Kreyszig, ―Advanced Engineering Mathematics‖, John Wiley and Sons, 10th Edition, New Delhi, 2016.
2. B.S. Grewal, ―Higher Engineering Mathematics‖, Khanna Publishers, New Delhi, 43rd Edition, 2014.

REFERENCES:

1. M. K. Venkataraman, ―Engineering Mathematics‖, Volume I, 4th Edition, The National Publication Company, Chennai, 2003.
2. Sivaramakrishna Dass, C. Vijayakumari, ―Engineering Mathematics‖, Pearson Education India, 4th Edition 2019.
3. H. K. Dass, and Er. Rajnish Verma, ―Higher Engineering Mathematics‖, S. Chand Private Limited, 3rd Edition 2014.
4. B.V. Ramana, ―Higher Engineering Mathematics‖, Tata McGraw Hill Publishing Company, 6th Edition, New Delhi, 2008.
5. S.S. Sastry, ―Engineering Mathematics‖, Vol. I & II, PHI Learning Private Limited, 4th Edition, New Delhi, 2014.

LIST OF EQUIPMENTS:

1. SCILAB- Open source

22CH101	ENGINEERING CHEMISTRY (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To understand the water quality criteria and interpret its applications in water purification.
- To gain insights into the basic concepts of electrochemistry and implement its applications in chemical sensors.
- To acquire knowledge on the fundamental principle of energy storage devices and relate it to electric vehicles.
- To identify the different types of smart materials and explore their applications in Engineering and Technology.
- To assimilate the preparation, properties and applications of nanomaterials in various fields.

UNIT I	WATER TECHNOLOGY	15

Sources of water –Impurities - Drinking water quality parameters –Hardness and its types, problems - Municipal water treatment and disinfection (chlorination- break-point

chlorination,UV, Ozonation). Boiler troubles- Scales and sludges, Boiler feed water: Requirements - Internal treatment (phosphate, colloidal, sodium aluminate and Calgon conditioning). External treatment –Ion exchange demineralization - Principle, process and fouling.

Desalination of brackish water: Reverse osmosis –principle-types of membranes, process and fouling.

(Theory-9)

List of Experiments:

1. Determination of total, temporary and permanent hardness of water by EDTA method.
2. Determination of chloride content of water sample by argentometric method.
3. Determination of alkalinity in water sample

(Laboratory-6)

UNIT II	ELECTROCHEMISTRY AND SENSORS	15

Introduction- Conductance- factors affecting conductance – Electrodes- origin of electrode potential – single electrode potential, standard electrode potential – measurement of single electrode potential –over voltage - reference electrodes (standard hydrogen electrode, calomel electrode)-ion selective electrode- glass electrode - Nernst equation (derivation), numerical problems, Electrochemical series and its applications.

Chemical sensors – Principle of chemical sensors – Breath analyzer- Gas sensors – CO2 sensors-Sensor for health care – Glucose sensor.

(Theory-9)

List of Experiments:

1. Determination of the amount of NaOH using a conductivity meter.
2. Determination of the amount of acids in a mixture using a conductivity meter.
3. Determination of the amount of given hydrochloric acid using a pH meter.

(Laboratory-6)

UNIT III	ENERGY STORAGE DEVICES AND ENERGY SOURCES	15

Batteries –Primary alkaline battery - Secondary battery - Pb-acid battery, Fuel cell - H2 – O2 fuel cell.

Batteries used in E- vehicle: Ni-metal hydride battery, Li-ion Battery, Li-air Battery Nuclear Energy – Nuclear fission, fusion, differences, characteristics – nuclear chain reactions – light water nuclear reactor – breeder reactor.

(Theory-9)

List of Experiments:

1. Determination of single electrode potential of the given electrode.
2. Estimation of the iron content of the given solution using apotentiometer.
3. Determination of electrochemical cell potential (using different electrodes/ different concentrations of electrolytes)

(Laboratory-6)

UNIT IV	SMART MATERIALS FOR ENGINEERING APPLICATIONS	15

Polymers – Definition – Classification – smart polymeric materials - Preparation, properties and applications of Piezoelectric polymer - Polyvinylidene fluoride (PVDF),

Electroactive polymer- Polyaniline (PANI) and Biodegradable polymer - Polylactic acid (PLA).Polymer composites: Definition, Classification – FRP's – Kevlar.
Shape Memory Alloys: Introduction, Shape memory effect – Functional properties of SMAs – Types of SMA - Nitinol (Ni-Ti) alloys - applications.
Chromogenic materials:Introduction – Types - applications.

(Theory-9)

List of Experiments:

1. Determination of themolecular weight of polymer using Ostwald viscometer.
2. Application of polymeric fibers in 3D printing.

(Laboratory-6)

UNIT V	NANO CHEMISTRY	15

Introduction – synthesis – top-down process (laser ablation, chemical vapor deposition), bottom-up process (precipitation, electrochemical deposition) – properties of nanomaterials – types – nanotubes -carbon nanotubes, applications of CNT - nanocomposites - General applications of nanomaterials in electronics, information technology, medical and healthcare, energy, environmental remediation, construction and transportation industries.

(Theory-9)

List of Experiments:

1. Determination of concentration of $BaSO_4$ nanoparticles by conductometric titrations.
2. Preparation of ZnO nanocrystal by precipitation method.

(Laboratory-6)

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Interpret the water quality parameters and explain the various water treatment methods.

CO2: Construct the electro chemical cells and sensors.

CO3: Compare different energy storage devices and predict their relevance in electric vehicles.

CO4: Classify different types of smart materials, their properties and applications in Engineering and Technology.

CO5: Integrate the concepts of nano chemistry and enumerate its applications in various fields.

TEXT BOOKS:

1. P. C. Jain and Monika Jain, ¯Engineering Chemistry‖, 17th Edition, Dhanpat Rai PublishingCompany Pvt. Ltd., New Delhi, 2022.
2. SivasankarB., ¯Engineering Chemistry‖, Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2012.

REFERENCES:

1. S.S. Daraand S.S. Umare, ¯A Textbook of Engineering Chemistry‖,12thEdition, S.Chand&Company, NewDelhi, 2013.
2. V.R. Gowarikar, Polymer Science, 2nd edition, New Age International Publishers, 2021.
3. J. C. Kuriacose and J. Rajaram, ¯Chemistry in Engineering and Technology‖, Volume -1&Volume -2, Tata McGraw-Hill Education Pvt. Ltd., 2010.
4. Geoffrey A. Ozin, Andre C. Arsenault and Ludovico Cademartiri, ¯Nanochemistry: A Chemical ApproachtoNanomaterials‖,2ndEdition,RSC publishers,2015.
5. Prasanna Chandrasekhar, ¯Conducting polymers, fundamentals and applications-Including Carbon Nanotubes and Graphene‖, Second Edition, Springer Science & Business Media, New York, 2019.
6. J. Mendham, R. C. Denney, J. D. Barnes, M. J. K. Thomas and B. Sivasankar, ¯Vogel's Quantitative Chemical Analysis‖, 6th edition, Pearson Education Pvt. Ltd., 2019.

LIST OF EQUIPMENTS:

1. Conductivity meter – 20 Nos.
2. pH meter - 19 Nos.
3. Potentiometer - 20 Nos.

22CS101	PROBLEM SOLVING USING C++ (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To learn problem solving and programming fundamentals.
- To gain knowledge on pointers and functions.
- To apply the principles of object orientated programming.
- To understand operator overloading, inheritance and polymorphism.
- To use the functionalities of I/O operations, files build C++ programs using exceptions.

UNIT I	PROBLEM SOLVING AND PROGRAMMING FUNDAMENTALS	15

Computational thinking for Problem solving – Algorithmic thinking for Problem solving - Building Blocks - Problem Solving and Decomposition - Dealing with Error – Evaluation. Overview of C – Data types – Identifiers – Variables – Storage Class Specifiers – Constants – Operators - Expressions – Statements – Arrays and Strings – Single-Dimensional – Two-Dimensional Arrays – Arrays of Strings – Multidimensional Arrays.

List of Exercise/Experiments:

1. Write C/C++ programs for the following:
    a. Find the sum of individual digits of a positive integer.
    b. Compute the GCD of two numbers.
    c. Find the roots of a number (Newton_s method)
2. Write C/C++ programs using arrays:

a. Find the maximum of an array of numbers.
b. Remove duplicates from an array of numbers.
c. Print the numbers in an array after removing even numbers.

3. Write C/C++ programs using strings:
   a. Checking for palindrome.
   b. Count the occurrences of each character in a given word.

UNIT II	POINTERS AND FUNCTIONS	15

Pointers -Variables – Operators – Expressions – Pointers and Arrays – Functions - Scope Rules – Function Arguments – return Statement – Recursion – Structures – Unions – Enumerations.

List of Exercise/Experiments:

1. Generate salary slip of employees using structures and pointers. Create a structure Employee with the following members:
   EID, Ename, Designation, DOB, DOJ, Basicpay
   Note that DOB and DOJ should be implemented using structure within structure.
2. Compute internal marks of students for five different subjects using structures and functions.

UNIT III	CLASSES AND OBJECTS	15

Concepts of Object Oriented Programming – Benefits of OOP – Simple C++ program - Classes and Objects - Member functions - Nesting of member functions - Private member functions - Memory Allocation for Objects - Static Data Members - Static Member functions - Array of Objects - Objects as function arguments - Returning objects - friend functions – Const Member functions - Constructors – Destructors.

List of Exercise/Experiments:

1. Write a program Illustrating Class Declarations, Definition, and Accessing Class Members.
2. Program to illustrate default constructor, parameterized constructor and copy constructors.

UNIT IV	OPERATOR OVERLOADING, INHERITANCE AND POLYMORPHISM	15

Operator Overloading - Overloading Using Friend functions – Inheritance – Types of inheritance – Virtual Base Class - Abstract Class – Constructors in Derived Classes - member class: nesting of classes.
Pointer to objects – this pointer- Pointer to derived Class - Virtual functions – Pure Virtual Functions – Polymorphism.

List of Exercise/Experiments:

1. Write a Program to Demonstrate the i) Operator Overloading. ii) Function Overloading.
2. Write a Program to Demonstrate Friend Function and Friend Class.
3. Program to demonstrate inline functions.
4. Program for Overriding of member functions.
5. Write C++ programs that illustrate how the following forms of inheritance are supported:

a) Single inheritance b)Multiple inheritance c)Multi level inheritance d)Hierarchical inheritance.

UNIT V	I/O, FILES AND EXCEPTIONS	15

C++ Streams – Unformatted I/O - Formatted Console I/O – Opening and Closing File – File modes - File pointers and their manipulations – Templates – Class Templates – Function Templates - Exception handling.

List of Exercise/Experiments:

1. Program to demonstrate pure virtual function implementation.
2. Count the number of account holders whose balance is less than the minimum balance using sequential access file.
3. Write a Program to Demonstrate the Catching of all Exceptions.
4. Mini project.

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Solve problems using basic constructs in C.
CO2: Implement C programs using pointers and functions.
CO3: Apply object-oriented concepts and solve real world problems.
CO4: Develop C++ programs using operator overloading and polymorphism.
CO5: Implement C++ programs using Files and exceptions.

TEXT BOOKS:

1. Herbert Schildt, ―The Complete Reference C++‖, 4th edition, MH, 2015. (Unit 1 & 2)
2. E Balagurusamy,‖Object Oriented Programming with C++‖, 4th Edition, Tata McGraw-Hill Education, 2008. (Unit 3, 4 & 5)

REFERENCES:

1. Karl Beecher,‖Computational Thinking: A beginner's guide to problem-solving and programming‖, BCS Learning & Development Ltd, 2017. (Unit 1)
2. Nell Dale, Chip Weems, ―Programming and Problem Solving with C++‖, 5th Edition, Jones and Barklett Publishers, 2010.
3. John Hubbard, ―Schaum's Outline of Programming with C++‖, MH, 2016.
4. Yashavant P. Kanetkar, ―Let us C++‖, BPB Publications, 2020
5. ISRD Group, ―Introduction to Object-oriented Programming and C++‖, Tata McGraw-Hill Publishing Company Ltd., 2007.
6. D. S. Malik, ―C++ Programming: From Problem Analysis to Program Design‖, Third Edition, Thomson Course Technology, 2007.
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01297200240671948837_shared/overview

LIST OF EQUIPMENTS:

1. Standalone desktops with C/C++ compiler (or) Server with C/C++ compiler.

22CS102	SOFTWARE DEVELOPMENT PRACTICES (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To discuss the essence of agile development methods.
- To set up and create a GitHub repository.
- To create interactive websites using HTML
- To design interactive websites using CSS.
- To develop dynamic web page using Java script.

UNIT I	AGILE SOFTWARE DEVELOPMENT AND Git and GitHub	15

Software Engineering Practices - Waterfall Model - Agility - Agile Process - Extreme Programming - Agile Process Models – Adaptive Software Development – Scrum – Dynamic Systems Development Method - Crystal - Feature Driven Development - Lean Software Development – Agile Modeling – Agile Unified Process – Tool set for Agile Process.

Introduction to Git -Setting up a Git Repository - Recording Changes to the Repository - Viewing the Commit History - Undoing Things - Working with Remotes -Tagging - Git Aliases - Git Branching - Branches in a Nutshell - Basic Branching and Merging - Branch Management - Branching Workflows - Remote Branches - Rebasing.

Introduction to GitHub - Set up and Configuration - Contribution to Projects, Maintaining a Project – Scripting GitHub.

List of Exercise/Experiments:

1. Form a Team, Decide on a project:
    a) Create a repository in GitHub for the team.
    b) Choose and follow a Git workflow
        - Each team member can create a StudentName.txt file with contents about themselves and the team project
        - Each team member can create a branch, commit the file with a proper commit message and push the branch to remote GitHub repository.
        - Team members can now create a Pull request to merge the branch to master branch or main development branch.
        - The Pull request can have two reviewers, one peer team member and one faculty. Reviewers can give at least one comment for Pull Request updating.
        - Once pull request is reviewed and merged, the master or main development branch will have files created by all team members.
2. Create a web page with at least three links to different web pages. Each of the web pages is to be designed by a team member. Follow Git workflow, pull request and peer reviews.
3. Form a Team, Decide on a project:
    c) Create a repository in GitHub for the team.
    d) Choose and follow a Git workflow
        - Each team member can create a StudentName.txt file with contents about themselves and the team project

- Each team member can create a branch, commit the file with a proper commit message and push the branch to remote GitHub repository.
- Team members can now create a Pull request to merge the branch to master branch or main development branch.
- The Pull request can have two reviewers, one peer team member and one faculty. Reviewers can give at least one comment for Pull Request updation.
- Once pull request is reviewed and merged, the master or main development branch will have files created by all team members.

4. Create a web page with at least three links to different web pages. Each of the web pages is to be designed by a team member. Follow Git workflow, pull request and peer reviews.

UNIT II	HTML	15

Introduction – Web Basics – Multitier Application Architecture – Cline-Side Scripting versus Server-side Scripting – HTML5 – Headings – Linking – Images – Special Characters and Horizontal Rules – Lists – Tables – Forms – Internal Linking – meta Elements - Form input Types – input and datalist Elements - Page-Structure Elements.

List of Exercise/Experiments:

1. Create web pages using the following:
   - Tables and Lists
   - Image map
   - Forms and Form elements
   - Frames

UNIT III	CSS	15

Inline Styles – Embedded Style Sheets – Conflicting Styles – Linking External Style Sheets - Positioning Elements - Backgrounds - Element Dimensions - Box Model and Text Flow – Media Types and Media Queries – Drop-Down Menus – Text Shadows – Rounded Corners – Colour – Box Shadows – Linear Gradients – Radial Gradients – Multiple Background Images – Image Borders – Animations – Transitions and Transformations – Flexible Box Layout Module – Multicolumn Layout.

List of Exercise/Experiments:

1. Apply Cascading style sheets for the web pages created.

UNIT IV	JAVASCRIPT BASICS	15

Introduction to Scripting – Obtaining user input – Memory Concepts – Arithmetic – Decision Making: Equality and Relational Operators - JavaScript Control Statements - Functions - Program Modules - Programmer-defined functions - Scope rules - functions – Recursion – Arrays – Declaring and Allocating Arrays – References and Reference Parameters – Passing Arrays to Functions – Multidimensional arrays.

List of Exercise/Experiments:

1. Form Validation (Date, Email, User name, Password and Number validation) using JavaScript.

UNIT V	JAVASCRIPT OBJECTS	15

Objects – Math, String, and Date, Boolean and Number, document Object – Using JSON to Represent objects – DOM: Objects and Collections – Event Handling.

List of Exercise/Experiments:

Implement Event Handling in the web pages.

Mini Projects-Develop any one of the following web applications (not limited to one) using above technologies.

a. Online assessment system
b. Ticket reservation system
c. Online shopping
d. Student management system
e. Student result management system
f. Library management
g. Hospital management
h. Attendance management system
i. Examination automation system
j. Web based chat application

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Apply agile development methods in software development practices.
CO2: Set up and create a GitHub repository.
CO3: Develop static and dynamic webpages using HTML.
CO4: Design interactive personal or professional webpages using CSS.
CO5: Develop web pages using Java script with event-handling mechanism.

TEXT BOOKS:

1. Roger S. Pressman, ―Software Engineering: A Practitioner‟s Approach‖, McGraw Hill International Edition, Nineth Edition, 2020.
2. Scott Chacon, Ben Straub, ―Pro GIT‖, Apress Publisher, 3rd Edition, 2014.
3. Deitel and Deitel and Nieto, ―Internet and World Wide Web - How to Program‖, Pearson, 5th Edition, 2018.

REFERENCES:

1. Roman Pichler, ―Agile Product Management with Scrum Creating Products that Customers Love‖, Pearson Education, 1 st Edition, 2010.
2. Jeffrey C and Jackson, ―Web Technologies A Computer Science Perspective‖, Pearson Education, 2011.
3. Stephen Wynkoop and John Burke, ―Running a Perfect Website‖, QUE, 2nd Edition, 1999.
4. Chris Bates, ―Web Programming – Building Intranet Applications‖, 3rd Edition, Wiley Publications, 2009.
5. Gopalan N.P. and Akilandeswari J., ―Web Technology‖, Second Edition, Prentice Hall of India, 2014.
6. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013094421427470336209 9_shared/overview

LIST OF EQUIPMENTS:

1. Systems with either Netbeans or Eclipse
2. Java/JSP/ISP Webserver/Apache
3. Tomcat / MySQL / Dreamweaver or
4. Equivalent/ Eclipse, WAMP/XAMP

22EC101	DIGITAL PRINCIPLESAND SYSTEMS DESIGN (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To acquire the knowledge in Digital fundamentals and its simplification methods.
- To familiarize the design of various combinational digital circuits using logic gates.
- To realize various sequential circuits using flip flops.
- To interpret various clocked sequential circuits.
- To elucidate various semiconductor memories and related technology.
- To build various logic functions using Programmable Logic Devices.

UNIT I	BOOLEAN ALGEBRA AND LOGIC GATES	9

Review of number systems-representation-conversions, Review of Boolean algebra-theorems, sum of product and product of sum simplification, canonical forms, min term and max term, Simplification of Boolean expressions-Karnaugh map,Implementation of Boolean expressions using logic gates and universal gates.

List of Exercise/Experiments:

1. Implementation of Boolean expression using logic gates.

UNIT II	COMBINATIONAL LOGIC CIRCUITS	9

Design of combinational circuits - Half and Full Adders, Half and Full Subtractors, Binary Parallel Adder – Carry look ahead Adder, Magnitude Comparator, Decoder, Encoder, Priority Encoder, Mux/De-mux, Parity Generator/Checker

List of Exercise/Experiments:

1. Design of adders
2. Design ofsubtractors.
3. Design of binary adder using IC7483
4. Design of Multiplexers &Demultiplexers.
5. Design of Encoders and Decoders.
6. Implementation of a boolean function using a multiplexer.

UNIT III	SEQUENTIAL CIRCUITS	9

Flip flops - SR, JK, T, D, Master/Slave FF - operation and excitation tables, Asynchronous and Synchronous Counters Design - Shift registers, Universal Shift Register

List of Exercise/Experiments:

1. Design and implementation of 3 bit ripple counters.
2. Design and implementation of 3 bit synchronous counter
3. Design and implementation of shift registers

UNIT IV	SYNCHRONOUSSEQUENTIAL CIRCUITS DESIGN	9

Design of clocked sequential circuits - Moore/Mealy models, state minimization, state assignment, circuit implementation

UNIT V	MEMORY AND PROGRAMMABLE LOGIC DEVICES	9

Basic memory structure ROM: PROM – EPROM – EEPROM -RAM – Static and dynamic RAM - Programmable Logic Devices: Programmable Logic Array (PLA) - Programmable Array Logic (PAL) - Implementation of combinational logic circuits using PLA, PAL.

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Implement digital circuits using simplified Boolean functions.
CO2: Realize Combinational circuits for a given function using logic gates.
CO3: Demonstrate the operation of various counters and shift registers using Flip Flops.
CO4: Analyze Synchronous Sequential circuits.
CO5: Summarize the various types of memory devices.
CO6: Design the Combinational circuits using Programmable Logic Devices.
CO7: Perform practical exercises as an individual and / or team member to manage the task in time.
CO8: Express the experimental results with effective presentation and report.

TEXT BOOKS:

1. M. Morris Mano and Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6th Edition, Pearson, 2018.
2. S.Salivahanan and S.Arivazhagan,Digital Circuits and Design, 5th Edition, Oxford University Press, 2018.

REFERENCES:

1. A.Anandkumar, Fundamental of digital circuits, 4th Edition, PHI Publication,2016.
2. WilliamKleitz, Digital Electronics-A Practical approach to VHDL, Prentice Hall International Inc, 2012.
3. CharlesH.Roth, Jr. andLarry L. Kinney, Fundamentals of Logic Design, 7th Edition, Thomson Learning, 2014.
4. Thomas L. Floyd, Digital Fundamentals, 11th Edition, Pearson Education Inc, 2017.
5. John.M Yarbrough, Digital Logic: Applications and Design, 1st Edition, Cengage India, 2006.

NPTEL LINK: https://nptel.ac.in/courses/108/105/108105132/

LIST OF EQUIPMENTS:

22GE202	COMPUTER AIDED ENGINEERING GRAPHICS (Common to All Branches)	L	T	P	C
		1	0	2	2

OBJECTIVES:

The Course will enable learners to:

- To help students understand universal technical drawing standards.
- To provide training on drafting software to draw part models.
- To demonstrate the concepts of orthographic and isometric projections.
- To use drawing skills for communicating concepts, ideas for engineering product design.
- Use pictorial views to visualize and draw the isometric view of the objects.

UNIT I	INTRODUCTION TO CONVENTIONS IN ENGINEERING DRAWING AND CONIC SECTIONS	9

Introduction to Engineering Drawing - Importance of graphics in engineering applications – Use of drafting instruments – BIS conventions and specifications – Size, layout and folding of drawing sheets – Lettering and dimensioning. Conic curves - Ellipse, Parabola and Hyperbola by Eccentricity method.

(Theory - 3)

List of Experiments:

1. Drawing of a title block with necessary text, projection symbol and lettering using drafting software.
2. Drafting of Conic curves - Ellipse, Parabola and Hyperbola

(Laboratory - 6)

UNIT II	ORTHOGRAPHIC PROJECTION	9

Visualization concepts and Orthographic Projection - Layout of views – Orthographic Projection- Conversion of pictorial diagram into orthographic views.

(Theory - 3)

List of Experiments:

1. Drawing orthographic view of simple solids like Prism, Pyramids, Cylinder, Cone, etc, and dimensioning.
2. Drawing of orthographic views from the given pictorial diagram.

(Laboratory -6)

UNIT III	PROJECTION OF PLANES	9

Projection of planes (polygonal and circular surfaces) inclined to both the planes by rotating object method.

(Theory - 3)

List of Experiments:

1. Drawing of plane Surface inclined to HP.
2. Drawing of plane Surface inclined to VP.

(Laboratory -6)

UNIT IV	PROJECTION OF SOLIDS	9

Projection of simple solids like Prisms, Pyramids, Cylinder and Cone when the axis is inclined to HP by rotating object method.

(Theory - 3)

List of Experiments:

1. Drawing of simple solids like prism and pyramids when the axis is inclined to HP.
2. Drawing of simple solids like cylinder and cone when the axis is inclined to HP.

(Laboratory -6)

UNIT V	ISOMETRIC DRAWING	9

Principles of isometric view – Isometric view of simple solids – Prism, Pyramid, Cylinder and Cone.

(Theory - 3)

List of Experiments:

1. Drawing isometric projection of simple solids.
2. Modeling of 2D to 3D objects using drafting software.

(Laboratory -6)

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Explain the various engineering standards required for drafting and explore knowledge in conic sections.

CO2: Draw the orthographic views of 3Dprimitive objects.

CO3: Describe the projection of plane surfaces by the rotating plane method.

CO4: Apply the projection concepts and drafting tools to draw projections of solids.

CO5: Sketch the pictorial views of the objects using CAD tools.

TEXT BOOKS:

1. Natarajan K.V., ¯A text book of Engineering Graphics‖, Dhanalakshmi Publishers, Chennai, 33rd Edition, 2020.
2. Venugopal K. and Prabhu Raja V., ¯Engineering Graphics‖, New Age International (P) Limited, 15th Edition, 2019.

REFERENCES:

1. Bhatt N.D. ¯Engineering Drawing‖, Charotar Publishing House, 53rd edition ,2019.
2. BasantAgarwal and Agarwal C.M., ¯Engineering Drawing‖, Tata McGraw Hill Publishing Company Limited, New Delhi, 3rd Edition, 2019.
3. Engineering Drawing Practice for Schools and Colleges BIS SP46:2003 (R2008), Published by Bureau of Indian Standards (BIS), 2008.
4. Parthasarathy. N.S and Vela Murali, ¯Engineering Graphics‖, Oxford University, Press, New Delhi, 2019.
5. Gopalakrishna. K.R., Engineering Drawing Vol. 1 & 2, Subhas Publications, 27th Edition, 2017.

22GE111	PRODUCT DEVELOPMENT LAB - 1 (Common to All Branches)	L	T	P	C
		0	0	2	1

The students may be grouped into 3 to 4 and work under a project supervisor. The device/system/component/prototype Idea to be developed by the students and a final presentation to be done by the students about the idea generated at the end of the semester.

OBJECTIVES:

The Course will enable learners to:

- Understand the functionalities and limitation of various machine/equipment
- Demonstrate various operations that can be performed to machines
- Summarize the basic principles of machines to convert their ideas into products

I 1.Study of Manufacturing Processes(Carpentry, Plumbing, Machines and Welding).
2. Study of fundamental operations of 3D Printer and Scanner with Software.
3. Study of Smart Machining (CNC and Laser cutting) and Engraving Techniques.

II 1. Study of Fundamental of Circuit Design.
2. Study of PCB Milling Machine.
3. Study of Soldering and Desoldering.

III 1. Study of Computer Peripheral Devices (Processing Information Devices)

IV 1. Present the Product Idea Presentation - Phase – I.

TOTAL: 30 PERIODS

Note:

The students can select the prototype to be made of their choice after learning the above exercises.

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the concept of manufacturing processes.

CO2: Describe the working of the machine element.

CO3: Discuss the various applications of engineering materials

CO4: Summarize the basics of core engineering concepts.

CO5: Describe the process for converting ideas into products

LIST OF EQUIPMENTS:

1. CNC Router – 1 No.
2. 3D Printer – 1 No.
3. 3D Scanner – 1 No.
4. Laser cutting Machine – 1 No.
5. Centre lathe – 2 Nos.
6. Arc welding transformer with cables and holders – 2 Nos.
7. Plumbing tools – 2 Nos.
8. Carpentry tools – 2 Nos.
9. Multimeter – 10 Nos.
10. Drilling Machine – 1 No.
11. Solder Stations 5 Sets
12. Desoldering Machine – 1 No.
13. PCB Milling Machine – 1 No.
14. Variable Power Supply – 1 No.
15. Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc. – 10 Sets
16. Personal Desktop Computers – 30 Nos.

SEMESTER – II

22GE201	TAMILS AND TECHNOLOGY (Common to All Branches)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Recognize the historical significance of weaving and pottery technologies in ancient Tamil civilization.
- Highlight the concepts of design and construction technology during the Sangam age.
- Provide an overview of manufacturing technology and its role in Tamil society.
- Illustrate the agricultural and irrigation techniques employed in ancient Tamil society.
- Promote scientific Tamil and Tamil computing.

UNIT I	WEAVING AND CERAMIC TECHNOLOGY	3

Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries (BRW) – Graffiti on Potteries.

UNIT II	DESIGN AND CONSTRUCTION TECHNOLOGY	3

Designing and Structural construction House & Designs in household materials during Sangam Age - Building materials and Hero stones of Sangam age – Details of Stage Constructions in Silappathikaram - Sculptures and Temples of Mamallapuram – Great Temples of Cholas and other worship places - Temples of Nayaka Period - Type study (Madurai Meenakshi Temple)- Thirumalai Nayakar Mahal - Chetti Nadu Houses, Indo - Saracenic architecture at Madras during British Period.

UNIT III	MANUFACTURING TECHNOLOGY	3

Art of Ship Building - Metallurgical studies - Iron industry - Iron smelting,steel -Copper and gold- Coins as source of history - Minting of Coins – Beads making-industries Stone beads - Glass beads - Terracotta beads -Shell beads/ bone beats - Archeological evidences - Gem stone types described in Silappathikaram.

UNIT IV	AGRICULTURE AND IRRIGATION TECHNOLOGY	3

Dam, Tank, ponds, Sluice, Significance of Kumizhi Thoompu of Chola Period, Animal Husbandry - Wells designed for cattle use - Agriculture and Agro Processing - Knowledge of Sea - Fisheries – Pearl - Conche diving - Ancient Knowledge of Ocean - Knowledge Specific Society.

UNIT V	SCIENTIFIC TAMIL & TAMIL COMPUTING	3

Development of Scientific Tamil - Tamil computing – Digitalization of Tamil Books – Development of Tamil Software – Tamil Virtual Academy – Tamil Digital Library – Online Tamil Dictionaries – Sorkuvai Project.

TOTAL: 15 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Identify the role of weaving and ceramic technology in ancient Tamil Culture.

CO2: Assess the design and construction technology ideas in the current Tamil society.

CO3: Identify the different types of manufacturing technology used in Tamil society and their significance.

CO4: Classify agricultural and irrigation technologies in ancient Tamil society and its current relevance.

CO5: Discuss the fundamentals of scientific Tamil and Tamil computing.

TEXT BOOKS & REFERENCE BOOKS:

1. தமிழக வரலாறு – மக்களும் பண்பாடும் – கே.கே. பிள்ளை (வெளியீடு: தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்).
2. கணினித் தமிழ் – முனைவர் இல. சுந்தரம். (விகடன் பிரசுரம்).
3. கீழடி – வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம் (தொல்லியல் துறை வெளியீடு)
4. பொருநை – ஆற்றங்கரை நாகரிகம். (தொல்லியல் துறை வெளியீடு)
5. Social Life of Tamils (Dr.K.K. Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
6. Social Life of the Tamils - The Classical Period (Dr.S .Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D.Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by: International Institute of Tamil Studies.)
9. Keeladi - _Sangam City Civilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
11. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
12. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL) – Reference Book

22MA201	TRANSFORMS AND NUMERICAL METHODS (Common to CSE / IT)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Introduce the concepts of Laplace transforms and Z-transforms.
- Illustrate the application of transforms in solving differential and difference equations.
- Explain the Numerical methods for handling algebraic and transcendental equations.
- Introduce the numerical techniques for interpolation, differentiation and integration.

UNIT I	LAPLACE TRANSFORMS	15

Laplace transforms – Sufficient condition for existence – Transform of elementary functions – Basic properties – Transforms of derivatives and integrals of functions – Derivatives and integrals of transforms -Transforms of unit step function and impulse

functions – Transform of periodic functions. Inverse Laplace transform – Convolution theorem (Statement only).

Theory: 9

Experiments using SCILAB:

1. Finding Laplace transform of a function.
2. Finding inverse Laplace Transforms.
3. Determine the input for given output function of Laplace Transform.

Laboratory: 6

UNIT II	Z – TRANSFORMS	15

Z-transforms – Elementary properties – Inverse Z-transforms – partial fractions method – residues method – Convolution theorem.

Theory: 9

Experiments using SCILAB:

1. Finding Z –transform of a sequence.
2. Finding convolution of two sequences.
3. Plotting the input and output function of Z transform.

Laboratory: 6

UNIT III	SOLUTION OF DIFFERENTIAL AND DIFFERENCE EQUATIONS	15

Solution of linear ordinary differential equation of second order with constant coefficients and first order simultaneous equations with constant coefficients using Laplace transform. Formation of difference equations – Solution of first and second order difference equations with constant coefficients using Z-transform.

Theory: 9

Experiments using SCILAB:

1. Solving second order Ordinary Differential Equation.
2. Finding the Laplace transform and its inverse of a function numerically.
3. Finding the Z-transform numerically

Laboratory: 6

UNIT IV	SOLUTION OF EQUATIONS AND EIGENVALUE PROBLEMS	15

Solution of algebraic and transcendental equations by Newton Raphson method - Solution of linear system of equations – Gauss elimination method – Gauss Jordan method – Gauss Seidel Iterative method- Eigenvalues of a matrix by Power method.

Theory: 9

Experiments using SCILAB:

1. Finding the real roots of algebraic and transcendental equations using Newton Raphson method.
2. Finding the largest Eigenvalue by power method.
3. Solving system of linear equations using Gauss Seidel Method.

Laboratory: 6

UNIT V	NUMERICAL DIFFERENTIATION AND INTEGRATION	15

Finite differences – Forward and Backward differences – Interpolation – Newton‘s forward and backward interpolation formulae - Lagrange‘s interpolation for unequal intervals - Numerical Differentiation - Newton‘s and Lagrange‘s formulae - Numerical integration using Trapezoidal and Simpson‘s 1/3 rules - Evaluation of double integrals by Trapezoidal and Simpson‘s 1/3 rules.

Theory: 9

Experiments using SCILAB:

1. Finding approximately the missing value using Lagrange interpolation.
2. Evaluating line integrals by trapezoidal rule.
3. Evaluating line integrals by Simpson's rule.

Laboratory: 6

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Determine Laplace transform and inverse transform of simple functions.

CO2: Determine Z- transform and inverse transform of simple functions.

CO3: Solve ordinary differential equations using Laplace transform and difference equations using Z-Transform.

CO4: Compute the solutions of algebraic, transcendental and the system of equations.

CO5: Appreciate the numerical techniques of interpolation in various intervals and apply the numerical techniques of differentiation and integration for engineering problems.

TEXT BOOKS:

1. Bali N., Goyal M. and Watkins C., ―Advanced Engineering Mathematics‖, Firewall Media (An imprint of Lakshmi Publications Pvt., Ltd.,), New Delhi, 7th Edition, 2009.
2. Grewal, B.S., and Grewal, J.S., ―Numerical Methods in Engineering and Science‖, Khanna Publishers, 10th Edition, New Delhi, 2015.

REFERENCES:

1. Erwin. Kreyszig, ―Advanced Engineering Mathematics‖, John Wiley and Sons, 10th Edition, New Delhi, 2016.
2. Jain R.K. and Iyengar S. R. K., ―Advanced Engineering Mathematics‖, Narosa Publications, New Delhi, 3rd Edition, 2007.
3. Wylie, R.C. and Barrett, L.C., ―Advanced Engineering Mathematics‖, Tata McGraw Hill Education Pvt. Ltd, 6th Edition, New Delhi, 2012.
4. Mathews, J.H. ―Numerical Methods for Mathematics, Science and Engineering‖, 2nd Edition, Prentice Hall, 1992.
5. Sastry S.S, ―Introductory Methods of Numerical Analysis‖, PHI Learning Pvt. Ltd, 5th Edition, 2015.

LIST OF EQUIPMENTS:

1. SCILAB - Open source

22CS201	DATA STRUCTURES	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To understand the concepts of List ADT.
- To learn linear data structures – stacks and queues ADTs.
- To understand and apply Tree data structures.
- To understand and apply Graph structures.
- To analyze sorting, searching and hashing algorithms.

UNIT I	LINEAR DATA STRUCTURES – LIST	15

Algorithm analysis - running time calculations - 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 Manipulation – All operations (Insertion, Deletion, Merge, Traversal).

List of Exercise/Experiments:

- Array implementation of List ADTs.
- Linked list implementation of List ADTs.

UNIT II	LINEAR DATA STRUCTURES – STACKS, QUEUES	15

Stack ADT – Stack Model - Implementations: Array and Linked list - Applications - Balancing symbols - Evaluating arithmetic expressions - Conversion of Infix to postfix expression - Queue ADT – Queue Model - Implementations: Array and Linked list - applications of queues - Priority Queues – Binary Heap – Applications of Priority Queues.

List of Exercise/Experiments:

- Array implementation of Stack and Queue ADTs.
- Linked list implementation of Stack and Queue ADTs.
- Applications of List – Polynomial manipulations
- Applications of Stack – Infix to postfix conversion and expression evaluation.

UNIT III	NON LINEAR DATA STRUCTURES – TREES	15

Tree ADT – tree traversals - Binary Tree ADT – expression trees – applications of trees – binary search tree ADT– AVL Tree.

List of Exercise/Experiments:

- Implementation of Binary Trees and operations of Binary Trees.
- Implementation of Binary Search Trees.
- Implementation of Heaps using Priority Queues.

UNIT IV	NON LINEAR DATA STRUCTURES - GRAPHS	15

Definition – Representation of Graph – Types of graph - Breadth-first traversal - Depth-first traversal – Topological Sort – Applications of graphs – BiConnectivity – Euler circuits.

List of Exercise/Experiments:

- Graph representation and Traversal algorithms.

UNIT V	SEARCHING, SORTING AND HASHING TECHNIQUES	15

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

List of Exercise/Experiments:

- Implement searching and sorting algorithms.

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Implement abstract data types for list.

CO2: Solve real world problems using appropriate linear data structures.

CO3: Apply appropriate tree data structures in problem solving.

CO4: Implement appropriate Graph representations and solve real-world applications.

CO5: Implement various searching and sorting algorithms.

TEXT BOOKS:

1. Mark Allen Weiss, ¯Data Structures and Algorithm Analysis in C++‖, 4th Edition, Pearson Education, 2014.
2. Sartaj Sahni, ¯Data Structures, Algorithms and Applications in C++‖, Silicon paper publications, 2004.

REFERENCES:

1. Rajesh K. Shukla, ¯Data Structures using C and C++‖, Wiley India Publications, 2009.
2. Narasimha Karumanchi, ¯Data Structure and Algorithmic Thinking with Python: Data Structure and Algorithmic Puzzles‖, CareerMonk Publications, 2020.
3. Jean-Paul Tremblay and Paul Sorenson, ¯An Introduction to Data Structures with Application‖, McGraw-Hill, 2017.
4. Mark Allen Weiss, ¯Data Structures and Algorithm Analysis in Java‖, Third Edition, Pearson Education, 2012.
5. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, ¯Fundamentals of Data Structures in C‖, Second Edition, University Press, 2008.
6. Ellis Horowitz, Sartaj Sahni, Dinesh P Mehta, ¯Fundamentals of Data Structures in C++‖, Second Edition, Silicon Press, 2007.
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501578165051392105 84/overview

LIST OF EQUIPMENTS:

1. Systems with Linux/Ubuntu Operating System with gnu C++ compiler

22PH201	PHYSICS FOR COMPUTER SCIENCE AND INFORMATION TECHNOLOGY (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Learn the fundamental concepts of Physics and apply this knowledge to scientific, engineering and technological problems.
- Make the students enrich basic knowledge in electronics and quantum concepts and apply the same in computing fields.

UNIT I	LASER AND FIBRE OPTICS	15

Population of energy levels – Einstein's A and B coefficients derivation - Resonant cavity - Optical amplification (qualitative) - Semiconductor lasers: homojunction and heterojunction- Engineering applications of lasers in data storage (qualitative).

Fibre optics: Principle and propagation of light through optical fibre - V-number - Types of optical fibres (Material, refractive index and mode) - Losses in optical fibre - Fibre optic communication - Fibre optic sensors (pressure and displacement).

(Theory -9)

List of Experiments:

- Determination of divergence of laser beam
- Determination of acceptance angle and numerical aperture of an optical fibre

(Laboratory -6)

UNIT II	ELECTRON THEORIES OF MATERIALS	15

Classical free electron theory - Expressions for electrical conductivity and thermal conductivity - Wiedemann-Franz law - Success and failures of CFT- Effect of temperature on Fermi function- Density of energy states and average energy of electron at 0 K - Energy bands in solids.

(Theory -9)

List of Experiments:

- Determination of thermal conductivity of a bad conductor by Lee's disc method
- Measurement of the internal resistance using potentiometer

(Laboratory -6)

UNIT III	SEMICONDUCTOR PHYSICS	15

Intrinsic Semiconductors – E-kdiagram-Direct and indirect band gap semiconductors - Carrier concentration in intrinsic semiconductors- Band gap determination-Extrinsic semiconductors - Carrier concentration in n-type and p-type semiconductors -Electrical conductivity of intrinsic and extrinsic semiconductors -Variation of Fermi level with temperature and impurity concentration - Hall effect and its applications.

(Theory-9)

List of Experiments:

- Bandgap determination of intrinsic semiconductor.
- Determination of wavelength of semiconductor laser (Laboratory -6)

UNIT IV	INTRODUCTION TO NANO DEVICES AND QUANTUM COMPUTING	15

Introduction to nanomaterial -Electron density in a bulk material - Size dependence of Fermi energy - Quantum confinement - Quantum structures - Density of states in quantum well, quantum wire and quantum dot structures - Band gap of nanomaterial. Quantum computing: Quantum states - classical bits - quantum bits or qubits - CNOT gate - multiple qubits - Bloch sphere - quantum gates - advantages of quantum computing over classical computing. (Theory - 9)

List of Experiments:

- Synthesis of nanoparticles by sol-gel method
- Determination of particle size using laser source (Laboratory - 6)

UNIT V	MAGNETIC AND SUPERCONDUCTING MATERIALS	15

Introduction- Bohr magneton -magnetic dipole moment - origin of magnetic moments - types of magnetic materials-Ferromagnetism: Domain Theory - antiferromagnetism - ferrimagnetism - magnetic principle in computer data storage - Magnetic hard disc (GMR sensor) - Introduction to spintronics.

Superconducting materials – properties, types of superconductors, applications – SQUID and MAGLEV trains - superconducting qubits in quantum computing.

(Theory -9)

List of Experiments:

- Determination of hysteresis loss using B-H loop
- Determination of magnetic susceptibility of a paramagnetic liquid using Quincke's apparatus

(Laboratory -6)

	TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Discuss the basic principles of working of laser and their applications in fibre optic communication

CO2: Summarize the classical and quantum electron theories and energy band structures

CO3: Describe the conductivity in intrinsic and extrinsic semiconductors and importance of Hall effect measurements

CO4: Associate the properties of nanoscale materials and their applications in quantum computing

CO5: Interpret the properties of magnetic and superconducting materials and their applications in computer data storage

TEXT BOOKS:

1. **S.O. Kasap**,Principles of Electronic Materials and Devices, McGraw-Hill Education (Indian Edition) 2020.
2. **Jasprit Singh**, Semiconductor Devices: Basic Principles, Wiley (Indian Edition) 2007.
3. **Parag K Lala**, Quantum Computing: A Beginner‘s Introduction, McGraw-Hill Education (Indian Edition) 2020.

REFERENCES:

1. **R.P. Feynman**, The Feynman Lectures on Physics - Vol. II, The New Millennium Edition, 2012.
2. **M.A.Wahab**, Solid State Physics, 3rd Edition, Narosa Publishing House Pvt. Ltd., 2015.
3. **B.Rogers, J. Adams and S.Pennathur**, Nanotechnology: Understanding Small System, CRC Press, 2014.
4. **C.P. Williams,** Explorations in Quantum Computing, Springer-Verlag London, 2011.
5. **Wilson J.D. and Hernandez C.A.,**Physics Laboratory Experiments, Houghton Mifflin Company, New York 2005.
6. **Department of Physics,** Physics laboratory manual, R.M.K. Group of Institutions, 2021.
7. **R.P. Feynman**, The Feynman Lectures on Physics - Vol. II, The New Millennium Edition, 2012.

LIST OF EQUIPMENTS:

1. Semiconductor Laser - 6 Nos.
2. Determination of optical fibre parameters - 6 Nos.
3. Lee‘s disc apparatus - 6 Nos.
4. Potentiometer - 6 Nos.
5. Bandgap determination set up - 6 Nos.
6. Sol-gel synthesis - 2 Nos.
7. B-H loop set-up - 5 Nos.
8. Quincke‘s apparatus - 2 Nos.

22HS101	PROFESSIONAL COMMUNICATION (Common to All Branches)	L	T	P	C
		2	0	2	3

OBJECTIVES:
The Course will enable learners to:

- Strengthen basic reading and writing skills.
- Comprehend listening contexts competently.
- Inculcate reading habit and develop effective reading skills.
- Improve active and passive vocabulary.
- Acquire speech clarity with right pronunciation.
- Develop vocabulary of a general kind and enhance grammatical accuracy.
- Imbibe Content and Language Integrated Learning (CLIL).

UNIT I	FORMAL AND INFORMAL COMMUNICATION	12

Listening: Short Texts, Short Formal and Informal Conversations
Speaking: Self Introduction, Exchanging Personal Information
Reading: Practice in Skimming, Scanning and Predicting, Reading Comprehension
Writing: Free Writing, Hints Development
Grammar: Parts of Speech, Prepositions.
Vocabulary: Compound Nouns, Technical Words.

(Theory 6)

1. Familiarization of Vowel Sounds-Monophthongs, Diphthongs and Consonant Sounds
2. Listening to Formal Conversations in British and American Accents
3. Guided Writing(Laboratory 6)

UNIT II	GRAMMAR AND LANGUAGE DEVELOPMENT	12

Listening: Telephonic Conversations.
Speaking: Sharing information of a personal kind - Greetings – Taking leave.
Reading: Short comprehension passages - Pre-reading and Post-reading (multiple choice questions shortquestions / open and close ended questions)
Writing: Instructions, Recommendations, Checklists
Grammar: Tenses, Framing _Wh' & _Yes' or _No' questions
Vocabulary: Numerical Adjectives, Collocations

.

(Theory 6)

1. Communication Etiquettes
2. Self -Introduction using SWOT Analysis

(Laboratory 6)

UNIT III	BASIC TECHNICAL WRITING AND STUDY SKILLS	12

Listening: Listening to longer texts and filling up the tables
Speaking: Asking about routine actions and expressing opinions
Reading: Short texts (Cloze Test)
Writing: Formal letters, E-mail writing, Interpretation of Charts and Graphs
Grammar: Cause and Effect expressions**,** Conditional Clauses
Vocabulary: Often misspelled and confusing words

(Theory 6)

Mechanics of Reading Skills
News Reading-Cloze Tests

(Laboratory 6)

UNIT IV	GROUP DISCUSSION AND JOB APPLICATIONS	12

Listening: Listening to recorded dialogues of conversations and completing exercises based on them
Speaking: Discussion on Social issues.
Reading: Reading text from magazines
Writing: Purpose Expressions, Letter of Application, Minutes of Meeting.
Grammar: Modal Verbs, Subject-Verb agreement
Vocabulary : Sequence Words

(Theory 6)

1. Group Presentation, Group Discussion: Do‘s and Don‘ts of GroupDiscussion
2. Discussions on failure and success in interviews of famous personalities Spotting Errors

(Laboratory 6)

UNIT V	ART OF REPORTING	12

Listening: Listening to TED talks
Speaking: Debate & Presentations
Reading: Biographies
Writing: Definitions (Single line & Extended), Report Writing (Industrial visit, Accident and Feasibilityreports)
Grammar: Reported speech
Vocabulary : Verbal Analogies (Theory 6)

1. Writing based on listening to academic lectures and discussions
2. Leadership skills, Negotiation skills
3. Mechanics of Report Writing

(Laboratory 6)

LIST OF PROJECTS

1. Create a podcast on a topic that will be interesting to college students
2. Read and Review (Movie/Book/Technical Article)
3. Presentation on Social Issues
4. Submit a report on ¯Global English: A study‖

TOTAL: 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Comprehend conversations and short talks delivered in English
CO2: Participate efficiently in informal conversations and develop an awareness of the self and apply well-defined techniques
CO3: Read articles of a general kind in magazines and newspapers efficiently
CO4: Write short general essays, personal letters and E-mails in English CO5: Develop vocabulary of a general kind by enriching reading skills

TEXT BOOKS:

1. **Kumar, Suresh E, & Sreehari, P.** *Communicative English*. Orient Black Swan, 2007.
2. **Richards, Jack C.** Interchange Students‘ Book-2 New Delhi: CUP,2015.

REFERENCES:

1. **Bailey, Stephen.** Academic Writing: *A practical guide for students*. New York: Rutledge,2011.
2. **Dhanavel, S P.** English and Soft Skills, Volume Two, Orient Black Swan.
3. **Elbow, Peter.** *Writing Without Teachers*. London: Oxford University Press, 1973.
4. **Larsen, Kristine.** *Stephen Hawking*: *A Biography*, Greenwood: Publishing Group, 2005.
5. **Redston, Chris & Gillies Cunningham.** Face2Face (Pre- intermediate Students'Book &Workbook) Cambridge University Press, New Delhi: 2005.
6. Lewis, Norman. Word Power Made Easy, Latest Edition: Penguin Random House India: 2015

WEB REFERENCES:

1. Basics of Business Communication
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_012688768083632128308_shared/overview
2. communicating to Succeed
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_012686653619175424640_shared/overview
3. Business English
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_012683227498151936279_shared/overview
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013267708367904768573/overview (lab support)
4. Business Writing
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012689477601009664 33_shared/overview
5. Email Etiquettes
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013294623865561088 17682_shared/overview
6. Email Writing Skills
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_012689543630135296 66_shared/overview
7. Time Management
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_012985921210736640721_shared/overview
8. Understanding Body Language
   https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_01297973765144

576024689_shared/overview

9. ONLINE RESOURCES:
10. https://infyspringboard.onwingspan.com/web/en/page/home

LIST OF EQUIPMENTS:

22CS202	JAVA PROGRAMMING (Common to All Branches)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To explain object oriented programming concepts and fundamentals of Java
- To apply the principles of packages, interfaces and exceptions
- To develop a Java application with I/O streams, threads and generic programming
- To build applications using strings and collections.
- To apply the JDBC concepts

UNIT I	JAVA FUNDAMENTALS	15

An Overview of Java - Data Types, Variables, and Arrays – Operators - Control Statements - Class Fundamentals - Declaring objects - Methods – Constructors - this keyword – Overloading methods - Overloading constructors - Access Control – Static – Final

List of Exercise/Experiments:

1. Develop a Java application to generate Electricity bill. You must use one super class called EB Bill and must have two sub classes namely Domestic Bill and Commercial Bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection (i.e domestic or commercial). Compute the bill amount using the following tariff

If the type of the EB connection is domestic, calculate the amount to be paid as follows: First 100 units - Rs. 1 per unit

101-200 units - Rs. 2.50 per unit 201 -500 units - Rs. 4 per unit

> 501 units - Rs. 6 per unit

If the type of the EB connection is commercial, calculate the amount to be paid as follows: First 100 units - Rs. 2 per unit

101-200 units - Rs. 4.50 per unit 201 -500 units - Rs. 6 per unit

> 501 units - Rs. 7 per unit

2. Arrays Manipulations: (Use Methods for implementing these in a Class)

a. Find kth smallest element in an unsorted array

b. Find the sub array with given sum

c. Matrix manipulations – Addition, Subtraction, Multiplication

d. Remove duplicate elements in an Array

e. Accept an integer value N and print the Nth digit in the integer sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 and so on till infinity.

Example: The 11th digit in the sequence 12345678910111213…. is 0.

UNIT II	INHERITANCE, INTERFACES AND EXCEPTION HANDLING	15

Inheritance: Inheritance basics, Using super, Method Overriding, Using Abstract Classes, Using final with Inheritance - Package and Interfaces: Packages, Packages and member access, Importing Packages, Interfaces, Static Methods in an Interface – Exception Handling: Exception- Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions.

List of Exercise/Experiments:

1. Develop a Java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages.
2. Develop a Java application with Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.
3. Design a Java interface for ADT Stack. Implement this interface using array and built-in classes. Provide necessary exception handling in both the implementations.
4. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains the methods print Area () that prints the area of the given shape and Numberofsides() that prints the number of sides of the given shape.
5. Write a Java program to apply built-in and user defined exceptions.

UNIT III	MULTITHREADING, I/O AND GENERIC PROGRAMMING	15

Multithreaded Programming: Creating a Thread, Thread Priorities, Synchronization, Interthread Communication – I/O: I/O Basics, Reading Console Input, Writing Console Output, Reading and Writing Files – Generics: Introduction, Generic class, Bounded Types, Generic Methods, Generic Interfaces, Generic Restrictions.

List of Exercise/Experiments:

1. Write a Java program to read and copy the content of one file to other by handling all file related exceptions.

UNIT IV	STRING HANDLING AND COLLECTIONS	15

Lambda Expressions - String Handling – Collections: The Collection Interfaces, The Collection Classes – Iterator – Map - Regular Expression Processing.

List of Exercise/Experiments:

1. String Manipulation:

a. Reversing a set of words and count the frequency of each letter in the string.

b. Pattern Recognition - Find the number of patterns of form 1[0]1 where [0] represents any number of zeroes (minimum requirement is one 0) there should not be any other character except 0 in the [0] sequence in a given binary string.
c. Remove all the occurrences of string S2 in string S1 and print the remaining.
d. Find the longest repeating sequence in a string
e. Print the number of unique string values that can be formed by rearranging the letters in the string S.

2. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
3. Collections:

a. Write a program to perform string operations using ArrayList. Write functions for the following

i. Append - add at end
ii. Insert – add at particular index
iii. Search
iv. List all string starts with given letter

b. Find the frequency of words in a given text.

UNIT V	JDBC CONNECTIVITY	15

JDBC – DataSource, Configurations, Connection, Connection Pools, Driver Types, ResultSet, Prepared Statement, Named Parameter, Embedded SQL (Insert, Update, Delete, Join, union etc), ResultSet Navigation, Connection Close and Clean up.

List of Exercise/Experiments:

- Mini Project (using JDBC)

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the object oriented programming concepts and fundamentals of Java.

CO2: Develop Java programs with the packages, interfaces and exceptions.

CO3: Build Java applications with I/O streams, threads andgenerics programming.

CO4: Apply strings and collections in developing applications.

CO5: Implement the concepts of JDBC.

TEXT BOOKS:

1. Herbert Schildt, ―Java: The Complete Reference‖, 11th Edition, McGraw Hill Education, 2019.

REFERENCES:

1. Cay S. Horstmann, Gary Cornell, ¯Core Java Volume – I Fundamentals‖, 11th Edition, Prentice Hall, 2019.
2. Paul Deitel, Harvey Deitel, Java SE 8 for programmers, 3rd Edition, Pearson, 2015.
3. Steven Holzner, Java 2 Black book, Dream tech press, 2011.
4. Timothy Budd, Understanding Object-oriented programming with Java, Third Edition, Pearson Education, 2008.
5. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29959473947367270000_shared/overview

LIST OF EQUIPMENTS:

1. Java and Eclipse / NetBeans IDE or Equivalent

22IT202	DATABASE MANAGEMENT SYSTEM	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- To understand the basic concepts of Data modeling and Database Systems.
- To understand SQL and effective relational database design concepts.
- To learn relational algebra, calculus and normalization.
- To know the fundamental concepts of transaction processing, concurrency control techniques, recovery procedure and data storage techniques.
- To understand query processing, efficient data querying and advanced databases.

UNIT I	DATABASE CONCEPTS	15

Concept of Database and Overview of DBMS - Characteristics of databases - Data Models, Schemas and Instances - Three-Schema Architecture - Database Languages and Interfaces - Introductions to data models types - ER Model- ER Diagrams - Enhanced ER Model - reducing ER to table Applications: ER model of University Database Application – Relational Database Design by ER- and EER-to-Relational Mapping.

List of Exercise/Experiments:

1. Data Definition Commands, Data Manipulation Commands for inserting, deleting, updating and retrieving Tables and Transaction Control statements

UNIT II	STRUCTURED QUERY LANGUAGE	15

SQL Data Definition and Data Types – Constraints – Queries – INSERT, UPDATE, and DELETE in SQL - Views - Integrity Procedures, Functions, Cursor and Triggers - Embedded SQL - Dynamic SQL.

List of Exercise/Experiments:

1. Database Querying – Simple queries, Nested queries, Sub queries and Joins
2. Views, Sequences, Synonyms
3. Database Programming: Implicit and Explicit Cursors

UNIT III	RELATIONAL ALGEBRA, CALCULUS AND NORMALIZATION	15

Relational Algebra – Operations - Domain Relational Calculus- Tuple Relational Calculus - Fundamental operations.

Relational Database Design - Functional Dependency – Normalization (1NF, 2NF 3NF and BCNF) – Multivalued Dependency and 4NF – Joint Dependencies and 5NF - De-normalization.

List of Exercise/Experiments:

1. Procedures and Functions
2. Triggers

UNIT IV	TRANSACTIONS, CONCURRENCY CONTROL AND DATA STORAGE	15

Transaction Concepts – ACID Properties – Schedules based on Recoverability, Serializability – Concurrency Control – Need for Concurrency – Locking Protocols – Two Phase Locking – Transaction Recovery – Concepts – Deferred Update – Immediate Update.

Organization of Records in Files – Unordered, Ordered – Hashing Techniques – RAID – Ordered Indexes – Multilevel Indexes - B+ tree Index Files – B tree Index Files.

List of Exercise/Experiments:

1. Exception Handling
2. Database Design using ER modeling, normalization and Implementation for any application
3. Database Connectivity with Front End Tools

UNIT V	QUERY OPTIMIZATION AND ADVANCED DATABASES	15

Query Processing Overview – Algorithms for SELECT and JOIN operations – Query optimization using Heuristics.

Distributed Database Concepts – Design – Concurrency Control and Recovery – NOSQL Systems – Document-Based NOSQL Systems and MongoDB.

List of Exercise/Experiments:

1. Case Study using real life database applications anyone 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

- Build Entity Model diagram. The diagram should align with the business and functional goals stated in the application.
- Apply Normalization rules in designing the tables in scope.
- Prepared applicable views, triggers (for auditing purposes), functions for enabling enterprise grade features.
- Build PL SQL / Stored Procedures for Complex Functionalities, ex EOD Batch Processing for calculating the EMI for Gold Loan for each eligible Customer.
- Ability to showcase ACID Properties with sample queries with appropriate settings

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Map ER model to Relational model to perform database design effectively.
CO2: Implement SQL and effective relational database design concepts.
CO3: Apply relational algebra, calculus and normalization techniques in database design.
CO4: Understand the concepts of transaction processing, concurrency control,

recovery procedure and data storage techniques.

CO5: Apply query optimization techniques and understand advanced databases.

TEXT BOOKS:

1. Elmasri R. and S. Navathe, ¯Fundamentals of Database Systems‖, Pearson Education, 7th Edition, 2016.
2. Abraham Silberschatz, Henry F.Korth, ¯Database System Concepts‖, Tata McGraw Hill , 7th Edition, 2021.

REFERENCES:

1. Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education, 2013.Raghu Ramakrishnan, Gehrke ¯Database Management Systems‖, MCGraw Hill, 3rd Edition 2014.
2. Plunkett T., B. Macdonald, ¯Oracle Big Data Hand Book‖ , McGraw Hill, First Edition, 2013
3. Gupta G K , ¯Database Management Systems‖ , Tata McGraw Hill Education Private Limited, New Delhi, 2011.
4. C. J. Date, A.Kannan, S. Swamynathan, ¯An Introduction to Database Systems‖, Eighth Edition, Pearson Education, 2015.
5. Maqsood Alam, Aalok Muley, Chaitanya Kadaru, Ashok Joshi, Oracle NoSQL Database: Real-Time Big Data Management for the Enterprise, McGraw Hill Professional, 2013.
6. Thomas Connolly, Carolyn Begg, ¯Database Systems: A Practical Approach to Design, Implementation and Management‖, Pearson, 6th Edition, 2015.
7. Database Management System Part - 1
8. Database Management System Part – 2
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_aut94241_shared/overv
9. Online Resources:
   https://infyspringboard.onwingspan.com/web/en/page/home

LIST OF EQUIPMENTS:

1. MySql and Eclipse / NetBeans IDE or Equivalent

22GE211	PRODUCT DEVELOPMENT LAB - 2 (Common to All Branches)	L	T	P	C
		0	0	2	1

The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. The student batches should study the device/system/component and will do literature review to develop prototype idea. Further at the end of the semester they will make a final presentation to exhibit the conceptual design skills and the process to develop a product.

OBJECTIVES:

The Course will enable learners to:

- Use the innovative design methodology to articulate the product concepts.
- Summarize the requisite Engineering Principles for transforming concepts into products.
- Conduct basic tests to extract the qualitative and quantitative performance factors.

LIST OF EXERCISE/EXPERIMENTS

1. Study of Basic Engineering Design Concepts.
2. Conduct a literature survey on the implementation of the design concepts.
3. Prepare the design concepts for an identified literature gap.
4. Present the Product Idea Presentation – Phase II.

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the working and capacity of various engineering systems.
CO2: Infer the outcomes in the product development process.
CO3: Perform basic engineering and material characterization tests.
CO4: Demonstrate the ability to provide conceptual design strategies for a product.
CO5: Implement the Science, Engineering, Technology and Mathematics (STEM) for product design.

LIST OF EQUIPMENTS:

22CH104	ENVIRONMENTAL SCIENCE AND SUSTAINABILITY (Common to All Branches)	L	T	P	C
		2	0	0	MC

OBJECTIVES:

The Course will enable learners to:

- To gain knowledge of the environment and various natural resources.
- To identify the Scientific and Technological solutions to pollution issues and waste management.
- To understand the significance of the conservation of biodiversity.
- To recognize the needs and benefits of sustainability and its management.
- To comprehend the effects of human population on the environment.

UNIT I	NATURAL RESOURCES	7

Definition, scope and importance of environment – need for public awareness. Introduction to natural resources - Types - Forest resources: Use and over-exploitation, deforestation and its impacts, Food resources: effects of modern agriculture, organic farming, Renewable energy sources - Solar, Wind, Geothermal, Tidal, OTE and Biomass.

Field activity -Tree plantation

UNIT II	POLLUTION AND WASTE MANAGEMENT	7

Pollution - Definition -causes, effects and control measures of (a) Air pollution (b) Water pollution (c) Soil pollution (d) Noise pollution (e) Nuclear hazards - nuclear accidents and holocaust -Role of an individual in prevention of pollution -Case studies.

Waste management- Municipal solid wastes, e- waste, plastic waste.

Field study – Solid waste management of the institution

UNIT III	BIODIVERSITY AND ITS CONSERVATION	6

Biodiversity: types – values of biodiversity, India as a mega-diversity nation – hot-spots of biodiversity – threats to biodiversity – endangered and endemic species, extinct, rare, vulnerable species of India – conservation of biodiversity: In-situ and ex-situ method.

Field study – Biodiversity of the institution

UNIT IV	SUSTAINABILITY AND MANAGEMENT	5

Sustainability-concept, needs and challenges-Circular economy -Sustainable Development Goals- Concept of Carbon footprint, Environmental Impact Assessment, Clean Development Mechanism, solutions.

Field study – Carbon footprint of the institution

UNIT V	HUMAN POPULATION	5

Introduction - Population growth, variation among nations,population explosion, Environment and human health – endemic/epidemic/pandemic- Role of information technology in environment and human health.

Case Study – Pandemics of 21st century

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Investigate and use conservational practices to protect natural resources.

CO2: Identify the causes of pollutants and illustrate suitable methods for pollution abatement.

CO3: Adapt the values of biodiversity and its conservation methods.

CO4: Recognize suitable sustainable development practices and apply it in day-to-day life.

CO5: Assess the impacts of human population and suggest suitable solutions.

TEXT BOOKS:

1. Anubha Kaushik and C.P. Kaushik, ¯Perspectives in environmental studies‖,New Age International Publishers,2nd edition, 2021.
2. Benny Joseph, Environmental Science and Engineering, Tata McGraw-Hill, New Delhi, 2017.
3. Gilbert M.Masters, Introduction to Environmental Engineering and Science, 3rd edition, Pearson Education, 2014.
4. Erach Bharuch, Textbook of Environmental Studies for Undergraduate Courses, Third Edition, Universities Press(I) Pvt. Ltd., Hyderabad, 2021.

REFERENCES:

1. William P.Cunningham & Mary Ann Cunningham Environmental Science: A Global Concern, McGraw Hill, 14th edition, 2017.
2. Rajagopalan, R, Environmental Studies-From Crisis to Cure, Oxford University Press, 2015.
3. G. Tyler Miller and Scott E. Spoolman, ¯Environmental Science, Cengage Learning India Pvt, Ltd., Delhi, 2014.
4. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall, 2012.
5. Bradley. A.S; Adebayo, A.O., Maria, P. Engineering applications in sustainable design and development, Cengage learning, 2015.
6. Environment Impact Assessment Guidelines, Notification of Government of India, 2006 and subsequent amendments, 2022

SEMESTER III

22MA301	DISCRETE MATHEMATICS	L	T	P	C
		3	1	0	4

OBJECTIVES:

The Course will enable learners to:

- Describe the arguments using connectives and rules of inference.
- Introduce the basic concept of counting and generating functions.
- Define the graphs and it's models.
- Understand the concept of group theory, lattices and Boolean algebra.

UNIT I	LOGIC AND PROOFS	12

Propositional logic - Propositional equivalences - Predicates and quantifiers - Nested quantifiers - Rules of inference - Introduction to proofs - Proof methods and strategy.

UNIT II	COMBINATORICS	12

Mathematical induction - Strong induction and well ordering The basics of counting - The pigeonhole principle - Permutations and combinations - Recurrence relations - Solving linear recurrence relations - Generating functions - Inclusion and exclusion principle and its applications.

UNIT III	GRAPHS	12

Graphs and graph models - Graph terminology and special types of graphs - Matrix representation of graphs and graph isomorphism - Connectivity - Euler and Hamilton paths.

UNIT IV	ALGEBRAIC STRUCTURES	12

Algebraic systems - Semi groups and monoids - Groups - Subgroups - Homomorphism_s - Normal subgroup and cosets - Lagrange_s theorem - Definitions and examples of Rings and Fields.

UNIT V	LATTICES AND BOOLEAN ALGEBRA	12

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

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Examine the validity of the arguments.

CO2: Apply various proof techniques and principles using analytic and combinatorial methods.

CO3: Develop the recurrence relation for the given problems.

CO4: Implement graph theory techniques to solve real time problems.

CO5: Understand groups, Rings and Fields.

CO6: Solve problems in Lattices and Boolean algebra.

TEXT BOOKS:

1. Rosen, K.H., "Discrete Mathematics and its Applications", 8th Edition, Tata McGraw Hill Pub. Co. Ltd., New Delhi, Special Indian Edition, 2021.
2. Tremblay, J.P. and Manohar.R, " Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2017.

REFERENCES:
1. Grimaldi, R.P. "Discrete and Combinatorial Mathematics: An Applied Introduction", 5th Edition, Pearson Education Asia, Delhi, 2014.
2. Lipschutz, S. and Mark Lipson., "Discrete Mathematics", Schaum's Outlines, Tata McGraw Hill Pub. Co. Ltd., New Delhi, 3rd Edition, 2010.
3. Koshy, T. "Discrete Mathematics with Applications", Elsevier Publications, 2006

22GE301	UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY	L	T	P	C
		2	1	0	3

OBJECTIVES:

The Course will enable learners to:
- Development of a holistic perspective based on self-exploration about themselves (human being), family, society and nature/existence.
- Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
- Strengthening of self-reflection.
- Development of commitment and courage to act.

UNIT I	NEED, BASIC GUIDELINES, CONTENT AND PROCESS FOR VALUE EDUCATION	12

- Purpose and motivation for the course, recapitulation from Universal Human Values-I
- Self-Exploration-what is it? - Its content and process; _Natural Acceptance‘ and Experiential Validation- as the process for self-exploration
- Continuous Happiness and Prosperity- A look at basic Human Aspirations
- Right understanding, Relationship and Physical Facility- The basic requirements for fulfilment of aspirations of every human being with their correct priority
- Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario
- Method to fulfil the above human aspirations: Understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking

UNIT II	UNDERSTANDING HARMONY IN THE HUMAN BEING – HARMONY IN MYSELF!	12

- Understanding human being as a co-existence of the sentient _I‘ and the material _Body‘
- Understanding the needs of Self (_I‘) and _Body‘ - happiness and physical facility
- Understanding the body as an instrument of _I‘ (I being the doer, seer 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 • Programs to ensure Sanyam and Health. Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss programs for ensuring health vs dealing with disease		
UNIT III	**UNDERSTANDING HARMONY IN THE FAMILY AND SOCIETY- HARMONY IN HUMAN-HUMAN RELATIONSHIP**	**12**
• 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. Include practice sessions to reflect on relationships in family, hostel and institutes extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.		
UNIT IV	**UNDERSTANDING HARMONY IN THE NATURE AND EXISTENCE - WHOLE EXISTENCE AS COEXISTENCE**	**12**
• Understanding the harmony in 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.		
UNIT V	**IMPLICATIONS OF THE ABOVE HOLISTIC UNDERSTANDING OF HARMONY ON PROFESSIONAL ETHICS**	**12**
• 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
- Sum up.

Include practice exercises and case studies will be taken up in practice (tutorial) sessions eg. To discuss the conduct as an engineer or scientist etc.

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Would become more aware of themselves, and their surroundings (family, society, nature).

CO2: Would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.

CO3: Would have better critical ability.

CO4: Would become sensitive to their commitment towards what they have understood (human values, human relationship, and human society).

CO5: Would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

TEXT BOOK:

1. R R Gaur, R Sangal, G P Bagaria, ¯Human Values and Professional Ethics‖, Excel Books, New Delhi, Second Edition 2019.

REFERENCES:

1. A Nagaraj, ¯Jeevan Vidya: Ek Parichaya‖, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. E. F Schumacher, "Small is Beautiful", Vintage classics, London, 1993.
3. A.N. Tripathi, ¯Human Values‖, New Age Intl. Publishers, New Delhi, Third Edition 2020.
4. Maulana Abdul Kalam Azad, "India Wins Freedom", Oriental blackswan private limited, Hyderabad, 2020.
5. Mahatma Gandhi, ¯Hind Swaraj or Indian Home Rule‖, Maheswari Publications, Delhi 2020.
6. Romain Rolland, "The life of Vivekananda and the universal gospel", Publication house of Ramakrishna Math, Kolkata, Thirty second edition 2018.
7. Romain Rolland, "Mahatma Gandhi: The man who become one with the universal being ¯, Srishti Publishers & Distributors, New Delhi, Sixth Edition 2013.
8. Heaton, Dennis P. "The story of stuff." (2010): 553-556.
9. Gandhi, Mohandas Karamchand, ¯The story of my experiments with truth: An autobiography‖, Om Books International, 2018.
10. Andrews, Cecile, ¯Slow is beautiful: new visions of community, leisure, and joie de vivre‖, New society publishers, 2006.
11. Kumarappa, Joseph Cornelius, ¯The economy of permanence. CP‖, All India Village Industries Assn., 1946.

12.	Vivekananda-Romain Rolland (English)
13.	Gandhi-Romain Rolland (English)

22CS302	COMPUTER ORGANIZATION AND ARCHITECTURE	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Describe the basic principles and operations of digital computers.
- Design arithmetic and logic unit for various fixed and floating point operations
- Construct pipeline architectures for RISC processors.
- Explain various memory systems & I/O interfacings
- Discuss parallel processor and multi-processor architectures

UNIT I	COMPUTER FUNDAMENTALS	9

Computer Types - Functional Units — Basic Operational Concepts — Number Representation and Arithmetic Operations - Performance Measurement — Instruction Set Architecture - Memory Locations and Addresses - Instructions and Instruction Sequencing - Addressing Modes.

UNIT II	COMPUTER ARITHMETIC	9

Addition and Subtraction of Signed Numbers - Design of Fast Adders - Multiplication of Unsigned Numbers - Multiplication of Signed Numbers - Fast Multiplication - Integer Division - Floating-Point Numbers and Operations.

UNIT III	BASIC PROCESSING UNIT AND PIPELINING	9

Basic Processing Unit: Concepts - Instruction Execution - Hardware Components - Instruction Fetch and Execution Steps -Control Signals - Hardwired Control.
Pipelining: Basic Concept - Pipeline Organization- Pipelining Issues - Data Dependencies - Memory Delays - Branch Delays - Resource Limitations - Performance Evaluation -Superscalar Operation.

UNIT IV	I/O AND MEMORY	9

Input/Output Organization: Bus Structure - Bus Operation - Arbitration - The Memory System: Basic Concepts - Semiconductor RAM Memories - Read-only Memories - Direct Memory Access - Memory Hierarchy - Cache Memories - Performance Considerations - Virtual Memory - Memory Management Requirements - Secondary Storage.

UNIT V	PARALLEL PROCESSING AND MULTICORE COMPUTERS	9

Parallel Processing: Use of Multiple Processors - Symmetric Multiprocessors - Multithreading and Chip Multiprocessors - Clusters - Nonuniform Memory Access Computers Vector Computation - Multicore Organization.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Explain the basic principles and operations of digital computers.
CO2: Design Arithmetic and Logic Unit to perform fixed and floating-point operations.
CO3: Analyze and Implement Instruction Execution.
CO4: Evaluate Pipelining and Superscalar Operations.
CO5: Understand I/O and Memory Systems.

CO6: Explore Parallel Processing and Multicore Computers.
TEXT BOOKS: 1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization, Tata McGraw Hill, Sixth edition, 2012. 2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface 5th edition, Morgan Kaufmann, 2013.
REFERENCES: 1. John P.Hayes, Computer Architecture and Organization, Third Edition, TataMcGraw Hill, 2012. 2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface, 6th edition, Morgan Kaufmann, 2021. 3. John L. Hennessy and David A. Patterson, Computer Architecture – A Quantitate Approach, Morgan Kaufmann / Elsevier Publishers, Fifth Edition,2012.

22CS307	**ADVANCED JAVA PROGRAMMING (Lab Integrated)**	**L**	**T**	**P**	**C**
		3	**0**	**2**	**4**

OBJECTIVES:

The Course will enable learners to:

- Gain a comprehensive understanding of the Java Collections Framework and its various interfaces and implementations.
- Learn the details of Java I/O streams and utility classes for managing dates, numbers, and currencies.
- Develop a thorough understanding of the Stream API introduced in Java 8 and its various operations.
- Explore advanced object serialization and string tokenizing techniques, including pattern matching with regular expressions.
- Understand advanced Stream API features and gain proficiency in using regular expressions for text processing.

UNIT I	**COLLECTIONS FRAMEWORK AND UTILITY CLASSES**	**9+6**

Introduction to Collections Framework - Collection Interface- Methods in Collection Interface - Iterable and Iterator Interfaces - List Interface- ArrayList - LinkedList - Set Interface - HashSet- LinkedHashSet - TreeSet - Map Interface - HashMap - LinkedHashMap- TreeMap - Queue Interface -PriorityQueue - Deque Interface - Utility Classes.

List of Experiments

1. Write a program that measures the time taken for insertion, deletion, and search operations on ArrayList, LinkedList, HashSet, and TreeSet for varying sizes of input data.
2. Implement a custom data structure that combines features of a list and a set.
3. Write a Java program to create a HashMap where the keys are strings, and the values are integers Add five key-value pairs to the map. Print all the keys and values in the

map. Remove an entry by key. Update the value associated with a specific key. Check if the map contains a specific key and a specific value.

UNIT II	I/O OPERATIONS, SERIALIZATION, AND DATE HANDLING	9+6

Date – Calendar – Comparable interface – Observer Interface – Streams - Types of Streams - The Byte-stream I/O hierarchy - Character Stream Hierarchy – Random Access File class – the java.io. Console Class – Serialization – Dates - Numbers, and Currency - Working with Dates - Numbers and Currencies.

List of Experiments

1. Create a class representing a complex object with nested data structures. Serialize the object to a file, then deserialize it back and verify that the object remains intact.
2. Write a program that formats dates and currencies according to different locales.
3. Implement a java program that allows users to open a text file, navigate through it using random access, insert, delete, and modify text at specific positions within the file.

UNIT III	STREAM API AND FUNCTIONAL PROGRAMMING PARADIGMS	9+6

Overview of Stream API - Importance of Stream API in Java 8 and Beyond – Functional Programming Concepts - Creating Streams - Stream Interface Methods - Stream Operations - Intermediate Filtering (filter)-Mapping (map, flatMap)-Sorting (sorted)-Distinct (distinct) - Limit and Skip (limit, skip) - Terminal Operations -Collecting Results (collect) - Reducing and Summarizing (reduce, summaryStatistics)-Iterating (forEach) - Matching and Finding (anyMatch, allMatch, noneMatch, findFirst, findAny) -Counting (count).

List of Experiments

1. Write a program that performs stream operations like filtering, mapping, and reducing.
2. Create an infinite stream generator that generates prime numbers. Implement methods
to check for primality and generate the next prime number.
3. Write a program that reads a text file containing sentences. Tokenize each sentence into words, filter out stopwords, and print the remaining words.

UNIT IV	ADVANCED STRING PROCESSING, OBJECT SERIALIZATION, AND I/O TECHNIQUES	9+6

String Tokenizer – Parsing - Tokenizing and Formatting - Locating Data via Pattern Matching, Tokenizing - Object Serialization - Serializable Interface - Writing and Reading Serializable Objects -Transient Keyword- SerialVersionUID - Advanced I/O - Piped Streams (PipedInputStream and PipedOutputStream) – SequenceInputStream - PushbackInputStream and PushbackReader.

List of Experiments

1. Write a program that reads a text file and tokenizes it into sentences using the StringTokenizer class.
2. Create a class hierarchy representing different types of objects (e.g., Person, Employee). Serialize instances of these classes to a file using object serialization.
3. Implement a program that uses advanced I/O techniques like PipedInputStream, PipedOutputStream, SequenceInputStream, and PushbackInputStream.

UNIT V	ADVANCED STREAM FEATURES AND REGULAR EXPRESSIONS	9+6

Importance and Use Cases of Advanced Stream Features - Creating Custom Streams - Stream Generators (Stream.generate, Stream.iterate) - Infinite Streams - Using Spliterators – Advanced Stream Operations - FlatMapping - Chaining Stream Operations - Stream Peeking (peek) - Advanced Filtering Techniques - Introduction to Regular Expressions - Character Classes - Quantifiers - Pattern Matching - Groups and Capturing - Regex in Java - java.util.regex Package Pattern Class - Matcher Class - String Manipulation with Regex - Splitting Strings - Replacing Text (replaceAll, replaceFirst) - Replacing with Backreferences.

List of Experiments

1. Implement custom stream generators using Stream.generate and Stream.iterate methods.
2. Write a program that demonstrates advanced stream operations like flatMapping, chaining stream operations, and peeking.
3. Develop a program that utilizes regular expressions to perform string manipulation tasks such as splitting strings, replacing text, and extracting specific patterns.

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Utilize the Java Collections Framework to solve complex data structure problems.

CO2: Demonstrate proficiency in Java I/O operations and manage data efficiently.

CO3: Learn to utilize the Stream API for complex data processing by applying functional programming techniques.

CO4: Understand and implement advanced object serialization techniques.

CO5: Effectively use regular expressions for advanced text processing tasks.

CO6: Build simple applications using advanced java programming concepts.

TEXT BOOK:

1. Cay S. Horstmann, "Core Java Volume I--Fundamentals," 12th Edition, 2019.
2. Joshua Bloch, "Effective Java," 3rd Edition, 2018.
3. Raoul-Gabriel Urma, "Java 8 in Action: Lambdas, Streams, and Functional-Style Programming," 1st Edition, 2014.
4. Herbert Schildt, "Java: The Complete Reference," 11th Edition, 2018.
5. Alan Mycroft and Martin Odersky, "Programming in Scala," 4th Edition, 2020.

REFERENCES:

1. Bruce Eckel, "Thinking in Java," 4th Edition, 2006.
2. Herbert Schildt, "Java: A Beginner's Guide," 8th Edition, 2019.
3. Richard Warburton, "Java 8 Lambdas: Pragmatic Functional Programming," 1st Edition, 2014.

LIST OF EQUIPMENTS:

JDK/Eclipse

22CS306	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Critically analyse the efficiency of alternative algorithmic solutions for the same problem
- Illustrate brute force and divide and conquer design techniques.
- Explain dynamic programming for solving various problems.
- Apply greedy technique and iterative improvement technique to solve optimization problems
- Examine the limitations of algorithmic power and handling it in different problems.

UNIT I	INTRODUCTION	9+6

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving –Fundamentals of the Analysis of Algorithmic Efficiency – Asymptotic Notations and their properties. Analysis Framework – Mathematical analysis for Recursive and Non-recursive algorithms

List of Exercise/Experiments:

1. Perform the recursive algorithm analysis.
2. Perform the non-recursive algorithm analysis.

UNIT II	BRUTE FORCE AND DIVIDE AND CONQUER	9+6

Brute Force - String Matching - Exhaustive Search - Knapsack Problem - Divide and Conquer Methodology – Binary Search – Merge sort – Quick sort - Multiplication of Large Integers – Closest-Pair and Convex Hull Problems - Transform and Conquer Method: Heap Sort

List of Exercise/Experiments:

1. Write a program to search an element using binary search
2. Write a program to sort the elements using merge sort and find time complexity.
3. Write a program to sort the elements using quick sort and find time complexity.
4. Write a program to sort the elements using heap sort

UNIT III	DYNAMIC PROGRAMMING	9+6

Dynamic programming – Principle of optimality – Floyd_s algorithm – Multi stage graph - Optimal Binary Search Trees - Longest common subsequence - Matrix-chain multiplication – Travelling Salesperson Problem – Knapsack Problem and Memory functions.

List of Exercise/Experiments:

1. Solve Floyd‘s algorithm
2. Write a program to find optimal binary search tree for a given list of keys.
3. Solve the multi-stage graph to find shortest path using backward and forward approach
4. Write a program to find the longest common subsequence

UNIT IV	GREEDY TECHNIQUE AND ITERATIVE IMPROVEMENT	9+6

Greedy Technique – Prim_s algorithm and Kruskal‘s Algorithm – Huffman Trees. The Maximum-Flow Problem – Maximum Matching in Bipartite Graphs - The Stable marriage Problem

List of Exercise/Experiments:

1. Write a program to find minimum spanning tree using Prim's algorithm
2. Implement Kruskal's algorithm to find minimum spanning tree
3. Write a program to solve maximum flow problem

UNIT V	BACKTRACKING AND BRANCH AND BOUND	9+6

P, NP NP- Complete and NP Hard Problems. Backtracking – N-Queen problem - Subset Sum Problem. Branch and Bound – LIFO Search and FIFO search - Assignment problem – Knapsack Problem - Approximation Algorithms for NP-Hard Problems – Travelling Salesman problem

List of Exercise/Experiments:

1. Write a program to implement sum of subset problem.
2. Write a program to solve N-Queen problem
3. Solve the assignment problem using branch and bound technique
4. Solve knapsack problem using branch and bound technique

TOTAL: 45+30=75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Evaluate the efficiency of algorithms using asymptotic notations.

CO2: Design, implement, and analyze the time complexity of various sorting algorithms

CO3: Develop solutions for optimization problems using dynamic programming techniques.

CO4: Implement and analyze greedy algorithms for optimization problems.

CO5: Develop solutions for complex problems using backtracking and branch and bound techniques.

CO6: Analyze NP-complete problems and implement approximation algorithms for these problems.

TEXT BOOKS:

1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2019.

REFERENCES:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, PHI Learning Private Limited, 2012.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford university press, 2014.
3. http://nptel.ac.in/

LIST OF EQUIPMENTS:

1. Standalone PC with C/C++/Java

22CS304	OPERATING SYSTEMS	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Explain the basic concepts of operating systems and process.
- Discuss threads and analyse various CPU scheduling algorithms.
- Describe the concept of process synchronization and deadlocks.
- Analyse various memory management schemes.
- Describe I/O management and file systems.

UNIT I	INTRODUCTION TO OPERATING SYSTEMS AND PROCESSES	6+6

Introduction: Computer system organization - architecture – Resource management - Protection and Security – Virtualization - Operating System Structures: Services - User and Operating-System Interface - System Calls - System Services - Design and Implementation - Building and Booting an Operating System – Processes: Process Concept - Process Scheduling - Operations on Processes – Inter process Communication - IPC in Shared-Memory Systems - IPC in Message-Passing Systems

List of Exercise/Experiments:

1. Basic Unix file system commands such as ls, cd, mkdir, rmdir, cp, rm, mv, more, lpr, man, grep, sed, etc..
2. Programs using Shell Programming.
3. Implementation of Unix System Calls.
4. Implementation of IPC using message queue
   a. Get the input data (integer value) from a process called sender
   b. Use Message Queue to transfer this data from sender to receiver process
   c. The receiver does the prime number checking on the received data
   d. Communicate the verified/status result from receiver to sender process, this status should be displayed in the Sender process.

Note: Simultaneously execute two or more processes. Don‘t do it as a single process

UNIT II	THREADS AND CPU SCHEDULING	6+6

Threads & Concurrency: Overview - Multicore Programming - Multithreading Models - Thread Libraries - Implicit Threading - Threading Issues - CPU Scheduling: Basic Concepts – Scheduling Criteria - Scheduling Algorithms - Thread Scheduling - Multi-Processor Scheduling - Real-Time CPU Scheduling

List of Exercise/Experiments:

1. Write a program to implement the following actions using pthreads
   a. Create a thread in a program and called Parent thread, this parent thread creates another thread (Child thread) to print out the numbers from 1 to 20. The Parent thread waits till the child thread finishes
   b. Create a thread in the main program, this program passes the 'count' as arguments to that thread function and this created thread function has to print your name 'count' times.
2. Write C programs to implement the various CPU Scheduling Algorithms.

UNIT III	PROCESS SYNCHRONISATION AND DEADLOCKS	6+6

Process Synchronization: The critical-section problem – Peterson‘s Solution, Synchronization hardware, Mutex locks, Semaphores, monitors - Classic problems of synchronization: Bounded Buffer Problem - Reader‘s & Writer Problem, Dinning Philosopher Problem. Deadlock: System model - Deadlock characterization, Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock.

List of Exercise/Experiments:

1. Process Synchronization using Semaphores. A shared data has to be accessed by two categories of processes namely A and B. Satisfy the following constraints to access the data without any data loss.
   a. When a process A1 is accessing the database another process of the same category is permitted.
   b. When a process B1 is accessing the database neither process A1 nor another 74 process B2 is permitted.
   c. When a process A1 is accessing the database process B1 should not be allowed to access the database. Write appropriate code for both A and B satisfying all the above constraints using semaphores.

   Note: The time-stamp for accessing is approximately 10 sec.
2. Bankers Algorithm for Deadlock Avoidance

UNIT IV	MEMORY MANAGEMENT	6+6

Memory Management: Contiguous Memory Allocation - Paging - Structure of the Page Table – Swapping - Virtual Memory: Demand Paging – Copy-on write – Page Replacement – Allocation of frames – Thrashing – Memory Compression

List of Exercise/Experiments:

1. Analysis and Simulation of Memory Allocation and Management Techniques
   i. First Fit ii. Best Fit iii. Worst Fit
2. Implementation of Page Replacement Techniques
   i. FIFO ii. LRU iii. Optimal page replacement

UNIT V	STORAGE MANAGEMENT	6+6

Mass Storage Structure: Overview of Mass Storage Structure- HDD scheduling – Swap Space Management, I/O systems: I/O Hardware, Application I/O interface, Kernel I/O Subsystem, File System Interface: File Concept – Access Methods – Directory Structure – Protection, File-System Implementation: File-System Structure- File-System Operations - Directory Implementation - Allocation Methods - Free-Space Management, - Case Study-Linux

List of Exercise/Experiments:

1. Simulation of File Allocation Techniques
   i. Sequential ii. Linked list iii. indexed
2. Implementation of File Organization Strategies
   Single level directory ii. Two level directory iii. Hierarchical level directory

TOTAL: 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Describe computer system organization, operating system structures, and services.

CO2: Demonstrate basic Unix commands and develop programs using shell scripting.

CO3: Analyze process concepts and scheduling, and implement interprocess communication using message queues.

CO4: Implement CPU scheduling algorithms and use threading models to manage concurrency.

CO5: Address process synchronization issues and implement deadlock avoidance techniques.

CO6: Understand and Apply Memory and Storage Management Techniques

TEXTBOOKS:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, ¯Operating System Concepts‖ II, 10th Edition, John Wiley and Sons Inc., 2018.

2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.

REFERENCES:

1. William Stallings, "Operating Systems: Internals and Design Principles", 7th Edition, Prentice Hall, 2018.
2. Achyut S.Godbole, Atul Kahate, ¯Operating Systems‖, McGraw Hill Education, 2016.

LIST OF EQUIPMENTS:

1. Standalone desktops with C/C++/Java/Equivalent compiler

22GE311	PRODUCT DEVELOPMENT LAB – 3 (Design and Analysis Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- To provide an adequate understanding of project/product concepts and creative design process.
- Create a methodology to develop solutions to complex systems.

The students can form a team of 3 or 4 to work on the approved topic by the faculty in-charge. All approved product/process topics should have the following stages as listed under activities. The faculty in-charge conducts a periodic review to endorse the work process and during the review, the faculty shall provide suggestions/ideas to improvise the project towards completion. An interim report (consisting of literature, photographs, proof of the work done, etc..) for all listed activities should be submitted by the team during periodic review for evaluation. A final project report is required at the end of the semester for evaluation.

LIST OF ACTIVITIES:

1. Develop the design stage for a product from the concept.
   - Researching it in-depth.
   - Ideating possible solutions.
   - Selecting a promising solution.
   - Make a mock-up model
   - Comprehend the design features of the mock-up model.
2. Evaluate the pros-cons of the mock-up (& with the existing product).
3. Generate the Design for Manufacturing and Assembly (DFMA) process route for the product with necessary interdisciplinary collaborations.

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Enhance their skills in design concepts, rules and procedures.
CO2: Develop their cognitive strategy to think, organize, learn and behave.
CO3: Demonstrate the ability to provide conceptual design strategies for a product.
CO4: Describe the procedure for designing a Mock-up model.
CO5: Recognize and apply appropriate interdisciplinary and integrative strategies for solving complex problems

LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS:

1. CNC Router - 1 No
2. 3D Printer - 1 No
3. 3D Scanner - 1 No
4. Laser cutting Machine - 1 No
5. Centre lathe - 2 Nos
6. Arc welding transformer with cables and holders - 2 Nos
7. Plumbing tools - 2 Sets
8. Carpentry tools - 2 Sets
9. Multimeter - 10 Nos
10. Drilling Machine - 1 No
11. Solder Stations - 5 Sets
12. Desoldering Machine - 1 No
13. PCB Milling Machine - 1 No
14. Variable Power Supply - 1 No
15. Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc. - 10 Sets
16. Personal Desktop Computers - 30 Nos
17. 3D Modelling software – Creo/ AutoCAD/ etc., - 30 Licence

22CS311	APTITUDE AND CODING SKILLS – I (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Develop vocabulary for effective communication and reading skills.
- Build the logical reasoning and quantitative skills.
- Develop error correction and debugging skills in programming.

List of Exercises:

1. English – Phase I

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase I

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern

recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase I

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase I

Logical, Compilation and Code reuse

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop vocabulary for effective communication and reading skills.
CO2: Build the logical reasoning and quantitative skills.
CO3: Develop error correction and debugging skills in programming.

SEMESTER IV

22EC441	MICROCONTROLLERS AND EMBEDDED SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Explain the Architecture of 8051 microcontroller
- Explain the assembly language programming of 8051.
- Demonstrate a microcontroller based system
- Understand about PIC microcontroller
- Be familiar with the basic components of embedded System.
- Learn the architecture and programming of ARM processor.

UNIT I	MICROCONTROLLER	9

Architecture of 8051 – Special Function Registers(SFRs) - I/O Pins, Ports and Circuits - Instruction set - Addressing modes

UNIT II	PROGRAMMING MICROCONTROLLER	9

Introduction to 8051 assembly language programming Data types & directives- Programs: addition, subtraction, multiplication, and division- Sum and Average of natural numbers - Programs: Logical AND, OR, EX-OR and NOT - Programs for Switching register banks -Programs for 1's Complement and 2's Complement - Programs for Nested loop.

UNIT III	INTERFACING MICROCONTROLLER	9

Programming 8051 Timers - Serial Port Programming - Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing - Stepper Motor and Waveform generation – Introduction to PIC16X Microcontroller - Comparison of Microprocessor, Microcontroller, PIC microcontroller.

UNIT IV	INTRODUCTION TO EMBEDDED SYSTEM	9

Embedded Systems Basics: Introduction to Embedded systems, Examples of embedded systems, Embedded system design process, Embedded System Hardware and Embedded System Software. ARM Processor Fundamentals: Registers, Current

Program Status Register, Pipeline, Exceptions and Interrupts, Major applications of embedded systems

UNIT V	ARM PROCESSOR AND EMBEDDED PROGRAMMING	9

ARM 7 Architecture, ARM Instruction Set: Data Processing Instructions, Programme Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions. Components for embedded programs- Models of programs- Assembly, linking and loading.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Acquire knowledge on the architecture of 8051 Microcontroller

CO2: Apply programming techniques in developing the assembly language program for microcontroller applications.

CO3: Analyze various types of interfacing devices with other peripheral devices

CO4: Design and construct Microcontroller based systems

CO5: Acquire knowledge on basic components of embedded system

CO6: Describe the architecture and programming of ARM processor

TEXTBOOKS:

1. Mohamed Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay, The 8051 Microcontroller and Embedded Systems: Using Assembly and C, 2nd Edition, Pearson Education, 2011.
2. Marilyn Wolf, Computers as Components, Principles of Embedded Computing System Design, Fourth Edition, Morgan Kaufmann Publisher (An imprint from Elsevier), 2016.

REFERENCES:

1. Scott MacKenzie, Raphael Chung-Wei Phan, The 8051 Microcontroller, 4th Edition, Pearson Education, 2008.
2. Martin P Bates, Programming 8 – bit PIC Microcontroller in C with Interactive Hardware Simulation, Newnes, 2008.
3. LylaB. Das, Embedded Systems: An Integrated Approach Pearson Education, 2013.

22MA401	PROBABILITY AND STATISTICS	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Provide the necessary basic concepts of random variables and to introduce some standard distributions.
- Test the hypothesis for small and large samples.
- Introduce the concepts of Analysis of Variances.
- Understand the concept of statistical quality control.

UNIT I	ONE-DIMENSIONAL RANDOM VARIABLES	15

Basic probability definitions- Independent events- Conditional probability (revisit) - Random variable - Discrete and continuous random variables - Moments - Moment generating functions - Binomial, Poisson, Geometric, Uniform, Exponential and Normal distributions.

List of Exercise/Experiments using R Programming:

1. Finding conditional probability.

2. Finding mean, variance and standard deviation.

UNIT II	TWO-DIMENSIONAL RANDOM VARIABLES	15

Joint distributions - Marginal and conditional distributions - Covariance - Correlation and linear regression - Transformation of random variables.

List of Exercise/Experiments using R Programming:

1. Finding marginal density functions for discrete random variables.
2. Calculating correlation and regression.

UNIT III	TESTING OF HYPOTHESIS	15

Sampling distributions - Estimation of parameters - Statistical hypothesis - Large sample tests based on Normal distribution for single mean and difference of means - Tests based on t and F distributions for mean and variance - Chi-square test- Contingency table (test for independent) - Goodness of fit.

List of Exercise/Experiments using R Programming:

1. Testing of hypothesis for given data using Z - test.
2. Testing of hypothesis for given data using t - test.

UNIT IV	DESIGN OF EXPERIMENTS	15

One way and Two-way classifications - Completely randomized design - Randomized blockdesign - Latin square design.

List of Exercise/Experiments R Programming:

1. Perform one-way ANOVA test for the given data.
2. Perform two-way ANOVA test for the given data.

UNIT V	STATISTICAL QUALITY CONTROL	15

Control charts for measurements ($\bar{X}$and R charts) - Control charts for attributes (p, c and npcharts) - Tolerance limits.

List of Exercise/Experiments using R Programming:

1. Interpret the results for $\bar{X}$Chart for variable data.
2. Interpret the results for R-Chart for variable data.

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Calculate the statistical measures of standard distributions.
CO2: Compute the correlation & regression for two dimensional random variables.
CO3: Apply the concept of testing the hypothesis.
CO4: Implement the concept of analysis of variance for various experimental designs.
CO5: Demonstrate the control charts for variables and attributes**.**

TEXT BOOKS:

1. R.A. Johnson, I. Miller and J. Freund, "Miller and Freund‘s Probability and Statistics for Engineers", Pearson Education, Asia, 8th Edition, 2015.
2. J.S. Milton and J.C. Arnold, "Introduction to Probability and Statistics", Tata McGrawHill, 4th Edition, 2017.

REFERENCES:

1. J.L. Devore, "Probability and Statistics for Engineering and the Sciences", Cengage Learning, New Delhi, 9th Edition, 2016.
2. S.M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 6th Edition, Elsevier, 2020.
3. M.R. Spiegel, J. Schiller and R.A. Srinivasan, "Schaum_s Outline of Theory and Problems of Probability and Statistics", Tata McGraw Hill Edition, 2004.
4. R.E.Walpole, R.H.Myers, S.L. Myers and K.Ye, "Probability and Statistics for Engineers and Scientists‖. Pearson Education, Asia, 9th Edition, 2012

LIST OF EQUIPMENTS:

22CS401	DISTRIBUTED AND CLOUD COMPUTING (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

- ✓ To articulate the concepts and models underlying distributed computing
- ✓ To maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms.
- ✓ To learn different distributed mutual exclusion algorithms.
- ✓ To develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud.
- ✓ To explain the high-level automation and orchestration systems that manage the virtualized infrastructure.

UNIT I	INTRODUCTION	6 + 6

Definition - Relation to computer system components - Message-passing systems versus shared memory systems - Primitives for distributed communication - Synchronous versus asynchronous executions. A model of distributed computations: A distributed program - A model of distributed executions - Models of communication networks - Global state of a distributed system.

List of Exercise/Experiments:

1. Implement a simple distributed program that communicates between two nodes using Java's RMI (Remote Method Invocation) API.
2. Develop a distributed program that uses Java's messaging API (JMS) to communicate between nodes. Explore the different messaging paradigms (pub/sub, point-to-point) and evaluate their performance and scalability.
3. Develop a model of a distributed program using Java's concurrency and synchronization primitives.

UNIT II	LOGICAL TIME, GLOBAL STATE, AND SNAPSHOT ALGORITHMS	6 + 6

Logical time – Scalar Time – Vector Time - Efficient implementations of vector clocks – Virtual Time. Global state and snapshot recording algorithms: System model - Snapshot algorithms for FIFO channels and non-FIFO channels.

List of Exercise/Experiments:

1. Develop a program in Java that implements vector clocks to synchronize the order of events between nodes in a distributed system.
2. Implement a snapshot algorithm for recording the global state of the distributed system using vector clocks, for both FIFO and non-FIFO channels. Test the algorithm by recording snapshots at various points in the system's execution and analyzing the resulting global state.

UNIT III	DISTRIBUTED MUTUAL EXCLUSION ALGORITHMS	6 + 6

Introduction - Lamport‘s algorithm - Ricart-Agrawala algorithm - Quorum-based mutual exclusion algorithms - Maekawa‘s algorithm - Suzuki-Kasami‘s broadcast algorithm.

List of Exercise/Experiments:

1. Implement Lamport's algorithm for mutual exclusion in a distributed system using Java's RMI API.
2. Develop a program in Java that implements Maekawa's algorithm for mutual exclusion in a distributed system.
3. Implement Suzuki-Kasami's broadcast algorithm in Java to achieve reliable message delivery in a distributed system.

UNIT IV	CLOUD INFRASTRUCTURE AND VIRTUALIZATION	6 + 6

Data Center Infrastructure and Equipment – Virtual Machines – Containers – Virtual Networks - Virtual Storage.

List of Exercise/Experiments:

1. Set up a virtualized data center using a hypervisor like VMware or VirtualBox and create multiple virtual machines (VMs) on it. Configure the VMs with different operating systems, resources, and network configurations, and test their connectivity and performance.
2. Deploy a containerized application on a virtual machine using Docker or Kubernetes.

UNIT V	AUTOMATION AND ORCHESTRATION	6 + 6

Automation - Orchestration: Automated Replication and Parallelism - The MapReduce Paradigm: The MapReduce Programming Paradigm – Splitting Input – Parallelism and Data size – Data access and Data Transmission – Apache Hadoop – Parts of Hadoop – HDFS Components – Block Replication and Fault Tolerance – HDFS and MapReduce - Microservices.

List of Exercise/Experiments:

1. Set up and configure a single-node Hadoop cluster.
2. Run the word count program in Hadoop.
3. Deploy a microservices architecture using a container orchestration tool like Kubernetes or Docker Swarm.

TOTAL: 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Articulate the main concepts and models underlying distributed computing.

CO2: Learn how to maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms.

CO3: Learn different distributed mutual exclusion algorithms

CO4: Develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud.

CO5: Explain the high-level automation and orchestration systems that manage the virtualized infrastructure.

CO6: Develop Containerized Applications

TEXT BOOKS:

1. Ajay D. Kshemkalyani, Mukesh Singhal, ¯Distributed Computing: Principles, Algorithms, and Systems‖, Cambridge University Press, 2011. (Unit 1, 2, 3)
2. Douglass E. Comer, ¯The Cloud Computing Book: The future of computing explained‖, CRC Press, 2021. (Unit 4, 5)

REFERENCES:

1. Arshdeep Bahga, Vijay Madisetti, ¯Cloud Computing: A Hands-on Approach‖, Universities Press Private Limited, 2014.
2. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, ¯Mastering Cloud Computing‖, Tata Mcgraw Hill, 2017.
3. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, ¯Distributed and Cloud Computing, From Parallel Processing to the Internet of Things‖, Morgan Kaufmann Publishers, 2012.
4. Hagit Attiya, Jennifer Welch, ¯Distributed Computing: Fundamentals, Simulations and Advanced Topics‖, John Wiley & Sons, Inc., 2004.

LIST OF EQUIPMENTS:

Java, VMWare/VirtualBox, Hadoop, AWS/GCP/Azure/Any cloud platform, Eclipse/Equivalent IDE.

22IT403	WEB DEVELOPMENT FRAMEWORKS (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Understand web semantics and related tools and framework
- Able to get hands on latest JS based web frameworks
- Develop a scalable and responsive web application
- Develop an industry ready application web enterprise feature

UNIT I	ADVANCED JAVASCRIPT	9+6

Introduction to HTML5 and CSS3, Media Queries, JS, DOM, BootStrap, Variables, Loops, Operators, Scope, Hoisting, Arrays, Spread, REST, DeStructuring

List of Exercise/Experiments

1) Create a JS Object for Bank Account (w attributes like à customer name, account type, balance, data of creation, bank name, branch name, pan card number). Using JS Object keyword, try to perform following activities
   - List down all the entries of the bank object
   - Check the existence of a key
   - If key found, get the value for the key
2) Spread Operator
   - Merge Customer and Account Arrays
   - Update the Customer Object with the new values
   - Develop a function that takes an Spread Argument and calculates total balance.

UNIT II	INTRODUCTION TO REACTJS	

Class-Inheritance, Methods, Extended Class-Map, filter and Reduce Functions, Functions - Arrow Functions, Lambda Expressions, REST - Introduction, Why JSX, Hello World Apps, Project Structure

List of Exercise/Experiments

1) Create a list of Bank Objects (same kind of object you used in above lab, but in a array format)
   - Display the banks where balance is greater than 200
   - deduct 10% of the Bank account balance, as part of monthly service fees
   - Display the banks where balance is greater than 200 and branch code is ¯Chennai‖
   - Add a new Bank to the given array
   - Delete a bank from the array (use splice operator)
   - Calculate the total balance of all bank accounts
2) Develop a Scientific calculator that does following operations
   - Rounded Value
   - Area of Circle
   - Calculating of Sin, Cos and Tan functions
   - Permiter of an Rectangle
   - Employ Arrow functions

Employ HOC

UNIT III	REACT COMPONENTS	9+6

Class vs Functional Components, React Class Based Components – component DidMount, WillUpdate, shouldupate, didcatchetc - State - UseState, UseRef, USeEffect,UseHistory Usage and Props(difference, when to use what, mutable or immutabilty,direction of flow), PropTypes, Auxillary Components, Controlled and Uncontrolled Components,Component Interaction (Parent to Child and Child to Parent), Iteration & Conditional Response

List of Exercise/Experiments

1) Create a collection of Customer by using
   - Weak Map and Map Collection in JS
   - Show Case the different feature set of the same.
2) Add Login Page, Dash Board Page, Admin Page
   - Enable React Routing

- Add React Protected Route, for authorization

UNIT IV	REACT LIBRARY - I	9+6

Event Bubbleup - Component Wrapper - Integration of CSS Modules - Forms Validations(YUP, Formik, Standard), Events Handling, Data Binding

List of Exercise/Experiments

1) Develop a React application that has User Registration Form w field level validations, data submission to a rest api end point, boot strap for responsive.
2) Use YUP or Formik to implement the same

UNIT V	REACT HOOKS	

Custom Hooks, HTTP - Fetch, Axios, Services, Behaviour Subjects - StateLess, StateFulll and Container Components, Error Handling - Build, Env, CORS, Unit Testing w React Testing Library - Introduction to react-native - Introduction to StoryBook

List of Exercise/Experiments

1) Employ backend API for Login Page functionality (authentication). Post login, store the user context (received from the back end server) in browser's session storage as objects. And use the same as creds during protected route verification
   - On the dashboard page, have a grid of Students. The data has to be bought from backend api
   - Employ useref, useeffect & usestate, and useHistory
   1) Enable Exception Handling
   2) Enable HOC and Aux Components
   3) Implement React-Testing Library

Business Use Case Implementations

1) Student Management System
2) Retail Bank System
3) eCommerce System
4) Student LMS Management System

TOTAL: 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Personalize web pages using text formatting, graphics, audio, and video.

CO2: Hands on knowledge on Rest API, propTypes

CO3: Able to develop a web application using latest React Framework

CO4: Apply various React features including functions, components, and services.

CO5: Able to develop application using ReactJshooks.

CO6: Design and integrate complex web components to enhance user interface and user experience in React Native and Storybook applications.

TEXTBOOKS:

1. David Flanagan, Javascript The Definitive Guide, Paperback, 7th Edition, 2020.
2. **David Choi ,Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL Paperback – Import, 18 December 2020**
3. Mehul Mohan, Advanced Web Development with React Paperback – 1 January 2020

E-RESOURCES:

1. Parental Website - https://reactjs.org/
2. The Road to Learn React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
3. Learning React: Functional Web Development with React and Redux by Alex Banks and Eve Porcello
4. Learning React by KirupaChinnathambi
5. "React Up & Running" by StoyanStefanov
6. https://www.edureka.co/reactjs-redux-certification-training
7. CodePen
8. CodeSandbox (Preferred)
9. Stackblitz

LIST OF EQUIPMENTS:

- NodeJS (v9.11.2)
- Github as code repository
- Visual studio code as IDE
- RTL as unit testing framework
- Responsive design w bootstrap
- ReactJS installation (v17)
- Chrome / FIreFox Browsers (latest)
- Responsive using Media Queries & Bootstrap Material & Ant design
- Design based Apps

22AM301	ARTIFICIAL INTELLIGENCE (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

- To understand the various Intelligent agents and search strategies in AI.
- To learn about different problem-solving strategies using heuristic function.
- To learn about knowledge-based agents and first order logics.
- To understand knowledge representation and planning.
- To know about the expert system.

UNIT I	ARTIFICIAL INTELLIGENCE AND INTELLIGENT AGENTS	9+6

Introduction to AI - Foundations of Artificial Intelligence - Intelligent Agents – Agents and Environments - Concept of rationality – Nature of environments – Structure of agents - Problem solving agents – Example Problems - Search Algorithms – Uninformed Search Strategies.

Lab Programs:

1. Implement basic search strategies – 8-Puzzle, 8 - Queens problem.
2. Implement Breadth First Search & Depth first Search Algorithm
3. Implement Water Jug problem.
4. Solve Tic-Tac-Toe problem.

UNIT II	PROBLEM SOLVING	9+6

Heuristic search strategies – heuristic functions- Game Playing – Mini-max Algorithm - Optimal decisions in games – Alpha-beta search -Monte-Carlo search for Games -

Constraint satisfaction problems – Constraint propagation – Backtracking search for CSP – Local search for CSP – Structure of CSP

Lab Programs:

1. Implement A* and memory bounded A* algorithms.
2. Implement Minimax algorithm & Alpha-Beta pruning for game playing.
3. Constraint Satisfaction Problem
4. Mini Project – Chess. Sudoku.

UNIT III	**LOGICAL AGENTS**	**9+6**

Knowledge-based agents – Logic - Propositional logic – Propositional theorem proving – Propositional model checking – Agents based on propositional logic

First-Order Logic – Syntax and semantics – Using First-Order Logic - Knowledge representation and engineering – Inferences in first-order logic – Propositional Vs First-Order Inference - Unification and First-Order Inference - Forward chaining – Backward chaining – Resolution.

Lab Programs:

1. Implement Unification algorithm for the given logic.
2. Implement forward chaining and backward chaining using Python.

UNIT IV	**KNOWLEDGE REPRESENTATION AND PLANNING**	**9+6**

Ontological engineering – Categories and objects – Events – Mental objects and modal logic – Reasoning systems for categories – Reasoning with default information Classical planning – Algorithms for classical planning – Heuristics for planning – Hierarchical planning – non-deterministic domains – Time, schedule, and resources – Analysis

Lab Programs:

1. Implementation of object detection.
2. Implement classical planning algorithms.

UNIT V	**LEARNING AND EXPERT SYSTEMS**	**9+6**

Forms of Learning – Developing Machine Learning systems – Statistical Learning - Deep Learning: Simple feed-forward network - Neural Networks – Reinforcement Learning: Learning from rewards – Passive and active Reinforcement learning.

Expert Systems: Functions – Main structure – if-then rules for representing knowledge – developing the shell – Dealing with uncertainty

Lab Programs:

1. Develop an Expert system.
2. Mini-Project – Develop Machine Learning based classification Models.

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Illustrate the structure of agents and to implement various Intelligent agents.

CO2: Apply search strategies in problem solving and game playing using heuristic function.

CO3: Implement logical agents and first-order logic problems.

CO4: Apply problem-solving strategies with knowledge representation mechanism for solving hard problems.

CO5: Demonstrate the basics of expert systems and to develop models using machine learning techniques.

CO6: Develop miniprojects using artificial intelligence concepts.

TEXT BOOKS:

1. Peter Norvig and Stuart Russel, Artificial Intelligence: A Modern Approach, Pearson, 4th Edition, 2020.

4. Bratko, Prolog: Programming for Artificial Intelligence‖, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.

REFERENCES:

1. Elaine Rich, Kevin Knight and B.Nair, Artificial Intelligence 3rd Edition, McGraw Hill, 2017.
2. Melanie Mitchell, Artificial Intelligence: A Guide for Thinking Humans. Series: Pelican Books, 2020
3. Ernest Friedman-Hill, Jess in Action, Rule-Based Systems in Java, Manning Publications, 2003
4. Nils J. Nilsson, The Quest for Artificial Intelligence, Cambridge University Press, 2009.
5. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems,1st Edition by Patterson, Pearson, India, 2015.
6. NPTEL Courses:
   a. An Introduction to Artificial Intelligence - https://onlinecourses.nptel.ac.in/noc23_cs05/preview
   b. Artificial Intelligence: Knowledge Representation And Reasoning - https://onlinecourses.nptel.ac.in/noc23_cs09/preview

22GE411	PRODUCT DEVELOPMENT LAB - 4 (Prototype Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Analyze the real-time problems in product development from an engineering perspective.
- Implement the DFMA process route to make and assemble the product.
- Test and qualify the product or a system with acquired knowledge.
- Identify the business opportunities for the developed product or process.

The student batch of PDD Lab 3 shall continue their product/ process design work under the guidance of the faculty incharge. All batches should cover the following stages of prototyping work as listed under activities. The faculty incharge shall conduct periodic reviews to endorse the work progress and during the review, the faculty shall provide suggestions/ideas to improvise the project towards completion. An interim report (consisting of BoM, Stages of Prototyping, photographs, proof of work done, etc..) for all listed activities should be submitted by the team during periodic review for evaluation. A final project report is required at the end of the semester and the evaluation is based on an oral presentation in front of the examiner panel constituted by the Head of the Department.

LIST OF ACTIVITIES:

1. Develop Engineering BoM for the approved industrial Mock-up from Phase III. Transform the Engineering BoM to develop a Prototype.
2. Devise / Plan an economically efficient manufacturing process to make the Prototype and testing.
3. Deliberation of the Product / Process outcome – Phase IV. Preparation and submission of a project report.

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1 Identify the real-time problems through literature.

CO 2 Develop feasible solutions for the problems.

CO 3 Evaluate the methods to develop solutions to the problem.

CO 4 Analyze the business opportunities for a new product.

CO 5 Prepare a detailed report for the experimental dissemination.

LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS:

1. CNC Router - 1 No
2. 3D Printer - 1 No
3. 3D Scanner - 1 No
4. Laser cutting Machine - 1 No
5. Centre lathe - 2 Nos
6. Arc welding transformer with cables and holders - 2 Nos
7. Plumbing tools - 2 Sets
8. Carpentry tools - 2 Sets
9. Multimeter - 10 Nos
10. Drilling Machine - 1 No
11. Solder Stations - 5 Sets
12. Desoldering Machine - 1 No
13. PCB Milling Machine - 1 No
14. Variable Power Supply - 1 No
15. Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc. - 10 Sets
16. Personal Desktop Computers - 30 Nos
17. Numerical Simulation Tools - 30 Licence
18. Test bench:
    Mech: Digital Micrometre/ Vernier/ Bore gauge/ etc EEE : (Based on the electrical components)
    ECE : (Based on the electronic components) - 5 Nos

22CS411	APTITUDE AND CODING SKILLS – II (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Develop advanced vocabulary for effective communication and reading skills.
- Build an enhanced level of logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.

- To apply data structures and algorithms in problem solving.

List of Exercises:

1. English – Phase II

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase II

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase II

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase II

Logical, Compilation and Code reuse

5. Automata - Phase II

Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching

Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop advanced vocabulary for effective communication and reading skills.
CO2: Build an enhanced level of logical reasoning and quantitative skills.
CO3: Develop error correction and debugging skills in programming.
CO4: Apply data structures and algorithms in problem solving.

SEMESTER V

22CS501	COMPUTER NETWORKS (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Study the fundamental concepts of computer networks and physical layer.
- Apply the knowledge of various protocols and techniques used in the data link layer.
- Implement the services of network layer and network layer protocols.
- Illustrate different protocols used in the transport layer.
- Build applications using the application layer protocols.

UNIT I	INTRODUCTION AND PHYSICAL LAYER	9+6

Data Communications – Network Types – Protocol Layering – Network Models (OSI, TCP/IP) Networking Devices: Hubs, Bridges, Switches – Performance Metrics – Transmission media - Guided media -Unguided media- Switching-Circuit Switching - Packet Switching.

List of Exercise/Experiments:

1. Practice different network commands available in Windows and Linux Operating Systems and troubleshoot the network.
2. Network configuration commands using Linux.

UNIT II	DATA LINK LAYER	9+6

Introduction – Link-Layer Addressing- Error Detection and Correction - Wired LANs: Ethernet - Wireless LANs – Introduction – IEEE 802.11, Bluetooth

List of Exercise/Experiments:

1. Error detection and correction mechanisms.
2. Flow control mechanisms.

UNIT III	NETWORK LAYER	9+6

Network Layer Services – IPV4 Addresses – Forwarding of IP Packets - Network Layer Protocols: IP, ICMP v4 – Unicast Routing Algorithms – Protocols – Multicasting Basics - IPV6 Addressing – IPV6 Protocol.

List of Exercise/Experiments:

1. Multi-client chatting in TCP and UDP using Socket programming
2. Implementation of HTTP, Web Caching, FTP using socket programming.

UNIT IV	TRANSPORT LAYER	9+6

Introduction – Transport Layer Protocols – Services – Port Numbers – User Datagram Protocol –Transmission Control Protocol – SCTP.

List of Exercise/Experiments:

1. Develop a DNS client server to resolve the given host name or IP address.
2. Simulation of unicast routing protocols.

UNIT V	APPLICATION LAYER	9+6

Application layer-WWW and HTTP – FTP – Email –Telnet –SSH – DNS – SNMP

List of Exercise/Experiments:

1. Observing Packets across the network and Performance Analysis of various Routing protocols.
2. Simulation of Transport layer Protocols and analysis of congestion control techniques in the network.

TOTAL: 45 +30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the fundamental concepts of computer networks.

CO2: Apply the various routing protocols to solve real-world problems.

CO3: Build simple applications to solve societal problems.

CO4: Apply the simulation tools to implement various protocols used in the various layers.

CO5: Analyze the various application layer protocols.

CO6: Apply the mathematical knowledge to do performance analysis of various routing protocols.

TEXT BOOK:

1. Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill Education, 5th Ed., 2017.

REFERENCES:

1. Computer Networking- A Top Down Approach, James F. Kurose, University of Massachusetts and Amherst Keith Ross, 8th Edition, 2021.
2. Computer Networks, Andrew S. Tanenbaum, Sixth Edition, Pearson, 2021.
3. Data Communications and Computer Networks, P.C. Gupta, Prentice-Hall of India, 2006.
4. Computer Networks: A Systems Approach, L. L. Peterson and B. S. Davie, Morgan Kaufmann, 3rd ed., 2003.

LIST OF EQUIPMENTS:

C/Java, Ubuntu OS, NS2 simulation tool

22CS502	THEORY OF COMPUTATION (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Learn about automata construction
- Learn equivalence of automata and regular expressions
- Design a context free grammar for any given language
- Understand the concepts of Turing Machine
- Understand undecidable problems and NP class problems

UNIT I	FINITE AUTOMATA	6 + 6

Introduction to Formal Proof- Additional Forms of Proof - Inductive Proofs- Finite Automata - Deterministic Finite Automata-Nondeterministic Finite Automata -Finite Automata with Epsilon Transitions.

List of Exercise/Experiments:

1. Design a Finite State Machine (FSM) that accepts all strings over input symbols {0,1} having consecutive 1's as a substring**.**
2. Construct epsilon closure from the given NFA.

UNIT II	REGULAR EXPRESSIONS	6 + 6

Regular Expressions - Finite Automata and Regular Expressions - Properties of Regular Languages - Proving Languages not to be regular using Pumping Lemma - Closure Properties of Regular Languages - Equivalence and Minimization of Automata.

List of Exercise/Experiments:

1. Convert the given Regular expression to NFA.
2. Design the DFA minimization from the given transition table.

UNIT III	CONTEXT FREE GRAMMAR	

CFG – Parse trees- Application of CFG - Ambiguity in Grammars and Languages - Pushdown Automata-Definition of the Pushdown Automaton- The Languages of a PDA- Equivalence of PDA‘s and CFG‘s.

List of Exercise/Experiments:

1. Design a PDA that accepts all string having equal number of 0‘1 and 1‘s over input symbol {0,1} for a language 0^n1^n where n>=1.
2. Design a PDA to accept WCWR where w is any binary string and W^R is reverse of that string and C is a special symbol.
3. Design a Program to create PDA machine that accept the well-formed parenthesis.

UNIT IV	TURING MACHINE	6 + 6

Normal Forms for CFG- Pumping Lemma for CFL - Closure Properties of CFL - Turing Machines and its types - Programming Techniques for Turing Machine – Turing machines as a computer of integer functions.

List of Exercise/Experiments:

1. Design a Turing Machine that‘s accepts the following language $a^nb^nc^n$ where n>0.
2. Design a Turing Machine to accept W^R where w is any binary string and WR is reverse of that string.

UNIT V	UNDECIDABILITY	

Non-Recursive Enumerable Language - Undecidable Problem with Regular Expressions - Undecidable Problems about Turing Machine - Universal Turing Machine- Post's Correspondence Problem and modified PCP- The Classes P and NP - An NP Complete Problems.

List of Exercise/Experiments:

1. Design the travelling salesman problem for NP complete problems.

TOTAL: 30 +30 = 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the fundamentals of finite automata and their applications in solving problems.

CO2: Analyze the behavior of formal languages and regular expressions to solve computational problems.

CO3: Understand the concept of context-free grammars (CFG) and their applications to construct language.

CO4: Evaluate the computational power and limitations of Turing machines and their relevance to modern computing systems.

CO5: Identify undecidable problems and analyze their implications for engineering practice and theoretical computer science.

CO6: Develop critical thinking skills to analyze and reason about language properties and their computational complexity.

TEXT BOOK:

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education,2014.

REFERENCES:

1. Introduction to Languages and The Theory of Computation, 4th Edition, John C Martin, TMH,2010.
2. Introduction to Computer Theory, Dniel I.A. Cohen, John Wiley.
3. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
4. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

LIST OF EQUIPMENTS:

C/C++

22CS503	MACHINE LEARNING (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

- To discuss the basics of Machine Learning and model evaluation.
- To study dimensionality reduction techniques.
- To understand the various classification algorithms.
- To elaborate on unsupervised learning techniques.
- To discuss the basics of neural networks and various types of learning.

UNIT I	INTRODUCTION	9+6

Machine Learning – Types – Applications – Preparing to Model – Activities – Data – Exploring
structure of Data – Data Quality and Remediation – Data Pre-processing – Modelling and Evaluation: Selecting a Model -Training a Model – Model representation and Interpretability –
Evaluating Performance of a Model – Improving Performance.

Lab Programs:

1. Implementation of Candidate Elimination algorithm
2. Implementation of ML model evaluation techniques (R-Squared/Adjusted R-Squared/Mean Absolute Error/Mean Squared Error)
3. Implementation of ML model evaluation techniques (Confusion Matrix/F1 Score/AUC-ROC Curve)

UNIT II	FEATURE ENGINEERING AND DIMENSIONALITY REDUCTION	9+6

Feature Engineering – Feature Transformation – Feature Subset Selection - Principle Component Analysis – Feature Embedding – Factor Analysis – Singular value decomposition and Matrix Factorization - Multidimensional scaling - Linear Discriminant Analysis - Canonical Correlation Analysis - Isomap - Locally linear Embedding - Laplacian Eigenmaps.

Lab Programs:

1. Write python code to identify feature co-relations (PCA)
2. Interpret Canonical Covariates with Heatmap
3. Feature Engineering is the way of extracting features from data and transforming them into formats that are suitable for Machine Learning algorithms. Implement python code for Feature Selection/ Feature Transformation/ Feature Extraction.
4. Mini Project – Feature Subset Selection

UNIT III	SUPERVISED LEARNING	9+6

Linear Regression -Relation between two variables - Steps - Evaluation – Logistic Regression - Decision Tree - Algorithms - Construction - Classification using Decision Tree – Issues – Rule - based Classification – Pruning the Rule Set – Support Vector Machines - Linear SVM - Optimal Hyperplane - Radial Basis Functions - Naïve Bayes Classifier - Bayesian Belief Networks.

Lab Programs:

1. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select the appropriate data set for your experiment and draw graphs.
2. Implement and demonstrate the working of the decision tree-based ID3 algorithm
3. Build a Simple Support Vector Machines using a data set

UNIT IV	UNSUPERVISED LEARNING	9+6

Clustering – Types – Applications - Partitioning Methods – K-means Algorithm – K-Medoids – Hierarchical methods – Density based methods DBSCAN – Finding patterns using Association Rules – Hidden Markov Model.

Lab Programs:

1. Implement a k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions
2. Implement market basket analysis using association rules
3. Mini Project using Clustering analysis.

UNIT V	NEURAL NETWORKS AND TYPES OF LEARNING	9+6

Biological Neuron – Artificial Neuron – Types of Activation function – Implementations of ANN –Architectures of Neural Networks – Learning Process in ANN – Back propagation – Deep Learning – Representation Learning – Active Learning – Instance based Learning – Association Rule Learning – Ensemble Learning Algorithm – Regularization Algorithm- Reinforcement Learning – Elements- Model-based- Temporal Difference Learning.

Lab Programs:

1. Build an ANN by implementing the Single-layer Perceptron. Test it using appropriate data sets.
2. Implement Multi-layer Perceptron and test the same using appropriate data sets.
3. Build a RBF Network to calculate the fitness function with five neurons.
4. Mini Project – Face recognition,

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:

At the end of this **course, the students will be able to:**

CO1: Explain the basics of Machine Learning and model evaluation.
CO2: Study dimensionality reduction techniques.
CO3: Understand and implement various classification algorithms.
CO4: Understand and implement various unsupervised learning techniques.
CO5: Build Neural Networks and understand the different types of learning.
CO6: Develop simple projects using machine learning concepts.

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, ¯Machine Learning‖, Pearson,2019. (Unit 1 – chap 1,2,3/ Unit 2 – Chap 4 / Unit 4 – 9 / Unit 5 – Chap 10, 11)
2. Ethem Alpaydin, ¯Introduction to Machine Learning, Adaptive Computation and Machine Learning Series‖, Third Edition, MIT Press, 2014. (Unit 2 – Chap 6 / Unit 4 – chap 8.2.3/ Unit 5 – Chap 18)

REFERENCES:

1. Anuradha Srinivasaraghavan,Vincy Joseph, ¯Machine Learning‖, First Edition, Wiley, 2019.(Unit 3 – Chap 7,8,9,10,11 / Unit 4 – 13, 11.4, 11.5,12)
2. Peter Harrington, ¯Machine Learning in Action‖, Manning Publications, 2012.
3. Stephen Marsland, ¯Machine Learning – An Algorithmic Perspective‖, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
4. Tom M Mitchell, ¯Machine Learning‖, First Edition, McGraw Hill Education, 2013.
5. Christoph Molnar, ¯Interpretable Machine Learning - A Guide for Making Black Box Models Explainable‖, Creative Commons License, 2020.
6. NPTEL Courses:
   a. Introduction to Machine Learning - https://onlinecourses.nptel.ac.in/noc23_cs18/preview

LIST OF EQUIPMENTS:

Systems with Anaconda, Jupyter Notebook, Python, Pytorch, scikit-learn, Tensorflow, Colab

22CS511	ADVANCED APTITUDE AND CODING SKILLS - I	L	T	P	C
		0	0	2	1

OBJECTIVES:

- To develop vocabulary for effective communication and reading skills.
- To build the logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.

LIST OF EXERCISES:

1. English – Phase I Advanced

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase I Advanced

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase I Advanced

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase I

Logical, Compilation and Code reuse

TOTAL: 30 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Develop vocabulary for effective communication and reading skills.

CO2: Build the logical reasoning and quantitative skills.

CO3: Develop error correction and debugging skills in programming.

SEMESTER VI

22CS603	PROFESSIONAL ETHICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Familiarize with Engineering Ethics and Human Values.
- Impart knowledge on codes of ethics, safety, responsibilities and rights of engineers.
- Give awareness on global issues related to environmental ethics, computer ethics, weapons development and corporate social responsibility

UNIT I	HUMAN VALUES	9

Morals, values and Ethics – Integrity – Work ethic – Service learning – Civic virtue – Respect for others – Living peacefully – Caring – Sharing – Honesty – Courage – Valuing time – Cooperation – Commitment – Empathy – Self-confidence – Character – Spirituality – Introduction to Yoga and meditation for professional excellence and stress management.

UNIT II	ENGINEERING ETHICS	9

Senses of _Engineering Ethics' – Variety of moral issues – Types of inquiry – Moral dilemmas – Moral Autonomy – Kohlberg's theory – Gilligan's theory – Consensus and Controversy – Models of professional roles - Theories about right action – Self-interest – Customs and Religion – Uses of Ethical Theories.

UNIT III	ENGINEERING AS SOCIAL EXPERIMENTATION	9

Engineering as Experimentation – Engineers as responsible Experimenters – Codes of Ethics – A Balanced Outlook on Law - The Challenger Case Study.

UNIT IV	SAFETY, RESPONSIBILITIES AND RIGHTS	9

Safety and Risk – Assessment of Safety and Risk – Risk Benefit Analysis and Reducing Risk - Case Studies: Chernobyl and Bhopal Disasters - Respect for Authority – Collective Bargaining – Confidentiality – Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) – Discrimination.

UNIT V	GLOBAL ISSUES	9

Multinational Corporations – Environmental Ethics – Computer Ethics – Weapons Development – Engineers as Managers – Consulting Engineers – Engineers as Expert Witnesses and Advisors – Moral Leadership –Code of Conduct – Corporate Social Responsibility.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Summarize the importance of human values in work place.

CO2: Discuss the senses of engineering ethics, moral dilemmas, moral autonomy and uses of ethical theories.
CO3: Describe the role of engineers as responsible experimenters and necessity of codes of ethics in engineering.
CO4: Explain safety, risk, responsibilities and rights in the society.
CO5: Analyze the global issues related to environmental ethics, computer ethics, weapons development and the role of engineers as expert witnesses and advisors.
CO6: Apply ethics in society and discuss the ethical issues related to engineering.

TEXT BOOK:

1. Mike W. Martin and Roland Schinzinger, ¯Ethics in Engineering‖, Tata McGraw Hill, New Delhi, 2017.
2. Govindarajan M, Natarajan S, Senthil Kumar V. S, ¯Engineering Ethics‖, Prentice Hall of India, New Delhi, 2013.

REFERENCES:

1. Charles B. Fleddermann, ¯Engineering Ethics‖, Pearson Prentice Hall, New Jersey, 2012
2. Charles E. Harris, Michael S. Pritchard and Michael J. Rabins, ¯Engineering Ethics – Concepts and Cases‖, Cengage Learning, 2018.
3. John R Boatright, ¯Ethics and the Conduct of Business‖, Pearson Education, New Delhi, 2017.
4. Edmund G Seebauer and Robert L Barry, ¯Fundamentals of Ethics for Scientists and Engineers‖, Oxford University Press, Oxford, 2008.

22CS601	COMPILER DESIGN (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Study the different phases of compiler
- Understand the techniques for tokenization and parsing
- Understand the conversion of source program into an intermediate representation
- Learn the different techniques used for run time environment and code generation
- Analyze various code optimization techniques

UNIT I	INTRODUCTION TO COMPILERS	9 + 6

Introduction-Structure of a Compiler-Role of the Lexical Analyzer - Input Buffering - Specification of Tokens - Recognition of Tokens-The Lexical Analyzer Generator LEX-Finite Automata - Regular Expressions to NFA-Optimization of DFA based pattern matches -Conversion from NFA to DFA - Minimization of DFA.

List of Exercise/Experiments:

1. Develop a lexical analyzer to recognize a few patterns in C. (Ex. identifiers, constants, comments, operators etc.). Create a symbol table, while recognizing identifiers.

UNIT II	SYNTAX ANALYSIS	9 + 6

Role of the Parser - Context-free grammars – Derivation Trees – Ambiguity in Grammars and Languages- Writing a grammar – Types of parsing - Top-Down Parsing – Predictive parser or LL(1) Parser -Bottom-Up Parsing – Shift Reduce Parser - LR Parsers - SLR, CLR, LALR Parser - Parser Generators YACC.

List of Exercise/Experiments:

1. Design a lexical analyzer for the given language. The lexical analyzer should ignore redundant spaces, tabs and new lines, comments etc.

UNIT III	INTERMEDIATE CODE GENERATION	9 + 6

Syntax Directed Definitions - Evaluation Orders for Syntax Directed Definitions – Application of Syntax Directed Translation - Intermediate Languages - Syntax Tree - Three Address Code – Implementation of Three address code – Declarations - Translation of Expressions - Type Checking.

List of Exercise/Experiments:

1. Implement a Lexical Analyzer using Lex Tool
2. Design Predictive Parser for the given language

UNIT IV	RUN-TIME ENVIRONMENT AND CODE GENERATION	9 + 6

Run Time Environment: Storage Organization-Storage allocation strategies - Access to nonlocal data on stack – Heap management - Parameter Passing - Issues in the design of Code Generator – Design of simple Code Generator -Register allocation and assignment.

List of Exercise/Experiments:

1. Implement an Arithmetic Calculator using LEX and YACC
2. Generate three address code for a simple program using LEX and YACC.

UNIT V	CODE OPTIMIZATION	9 + 6

Principle sources of optimization – Peep hole Optimization – DAG construction -Basic blocks and flow graph - Optimization in Basic blocks – Data flow analysis.

List of Exercise/Experiments:

1. Generate three address code for a simple program using LEX and YACC.
2. Implement simple code optimization techniques (Constant folding, Strength reduction and Algebraic transformation)
3. Implement back-end of the compiler for which the three address code is given as input and the 8086 assembly language code is produced as output.

TOTAL: 45 +30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the structure and role of lexical analyzers in the compilation process.

CO2: Design and implement different parsers and the generators like YACC for automating parser construction.

CO3: Understand syntax-directed definitions and their applications in intermediate code generation.

CO4: Analyze the different techniques used for efficient assembly code generation.
CO5: Design efficient code by implementing different code optimization techniques
CO6: Apply DAG technique to optimize the basic blocks used for data flow analysis

TEXT BOOK:

3. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, ―Compilers: Principles, Techniques and Tools‖, Second Edition, Pearson Education Limited, 2014.

REFERENCES:

5. Randy Allen, Ken Kennedy, ―Optimizing Compilers for Modern Architectures: A Dependence-based Approach‖, Morgan Kaufmann Publishers, 2002.
6. Steven S. Muchnick, ―Advanced Compiler Design and Implementation‖, Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint, 2003.
7. Keith D Cooper and Linda Torczon, ―Engineering a Compiler‖, Morgan Kaufmann Publishers, Elsevier Science, 2004.
8. V. Raghavan, ―Principles of Compiler Design‖, Tata McGraw Hill Education Publishers, 2010.
9. Allen I. Holub, ―Compiler Design in C‖, Prentice-Hall Software Series, 1993.

LIST OF EQUIPMENTS:

C/C++

22CS602	OBJECT ORIENTED SOFTWARE ENGINEERING (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:
The Course will enable learners to:

- Explain software engineering principles and activities involved in building large software programs.
- Describe the process of requirements gathering, analysis and unified modelling
- Illustrate the object oriented design process.
- Analyse various traditional and object oriented testing methods
- Apply estimation techniques, schedule project activities and compute pricing.

UNIT I	PRODUCT AND PROCESS	6+6

The Nature of Software - Defining the Discipline - The Software Process - Process models - Prescriptive Process Models - Product and Process - Agility and Process - What is an Agile Process? - Scrum - Other Agile Frameworks - Kanban - DevOps

List of Exercise/Experiments:

1. Identify a software system that needs to be developed.
2. Document the Software Requirements Specification (SRS) for the identified system.

UNIT II	REQUIREMENTS AND UNIFIED MODELING	6+6

Requirements Engineering - Establishing the Groundwork: Nonfunctional Requirements - Requirements Gathering – Developing Use Cases – Negotiating and Validating Requirements.

Unified Modeling Language – Introduction – Static and Dynamic Models – Modelling – Introduction to the UML - UML Diagrams - UML Class Diagrams - Use-Case Diagram - UML Dynamic Modelling.

List of Exercise/Experiments:

1. Identify use cases and develop the Use Case model.
2. Identify the conceptual classes and develop a Domain Model and also derive a Class Diagram from that.

UNIT III	OBJECT ORIENTED ANALYSIS AND DESIGN	6+6

Object oriented Analysis process – Business object Analysis – Use-case driven OOA – Business process modelling – Use case model. Design Concepts – Design Process – Design Concepts – Design Model: Design Principles and Design Elements. Architectural Design – Designing class-based components - Conducting Component Level Design – User Interface Analysis and Design – Pattern-Based Software Design.

List of Exercise/Experiments:

1. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams
2. Draw relevant State Chart and Activity Diagrams for the same system

UNIT IV	SOFTWARE TESTING	6+6

Component Level: A Strategic Approach to Software Testing – White-Box Testing – Black Box Testing - Object Oriented Testing Integration Level: Integration Testing – AI and Regression Testing – Integration Testing in the OO Context Specialized Testing for Mobility: Web application Testing – Web Testing Strategies – Security Testing – Performance Testing – Real time Testing – Testing AI Systems – Testing Virtual Environments.

List of Exercise/Experiments:

1. Implement the system as per the detailed design
2. Test the software system for all the scenarios identified as per the usecase diagram

UNIT V	SOFTWARE PROJECT MANAGAMENT	

Software Metrics and Analytics: Software Measurement – Product Metrics. Creating a Viable Software Plan: The Project Planning Process – Software Scope and Feasibility – Decomposition and Estimation Techniques – Project Scheduling. Risk Management: Reactive Versus Proactive Risk Strategies – Risk Identification – Risk Projection – The RMMM Plan.

Software Process Improvement: The SPI Process – The CMMI

List of Exercise/Experiments:

1. Improve the reusability and maintainability of the software system by applying appropriate design patterns.
2. Implement the modified system and test it for various scenarios

SUGGESTED DOMAINS FOR MINI-PROJECT:

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

TOTAL: 30 + 30= 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understanding Software and Software Processes

CO2: Analyze and gather software requirements.

CO3: Use UML to create static and dynamic models

CO4: Design software components using object-oriented principles.

CO5: Apply various software testing strategies.

CO6: Develop software projects effectively.

TEXT BOOKS:

1. Roger S. Pressman, ¯Software Engineering: A Practitioner_s Approach‖ , McGraw Hill International Edition, Nineth Edition, 2020.
2. Ali Bahrami, ¯Object Oriented Systems Development‖, McGraw Hill International Edition,2017.

REFERENCES:

1. Micheal Blalh and James Rumbaugh, Object Oriented Modeling and Design with UML, 2nd edition Pearson 2013.
2. Ian Sommerville, ¯Software Engineering‖, Tenth Edition, Pearson Education, 2016.
3. Ivar Jacobson, Harold Bud Lawson, Pan-Wei Ng, Paul E. McMahon, Michael Goedicke, ¯The Essentials of Modern Software Engineering‖, Morgan & Claypool Publishers, 2019.
4. Booch, G, Jacobson I, Rumbaugh J, ¯The Unified Modeling Language User Guide‖, Addison Wesley, 2008.
5. Martin Fowler, ¯UML Distilled: A Brief Guide to the Standard Object Modeling Language‖, 3rd edition, Addison Wesley, 2003.

LIST OF EQUIPMENTS:

ArgoUML, Visual Paradigm

22CS611	ADVANCED APTITUDE AND CODING SKILLS - II	L	T	P	C
		0	0	2	1

OBJECTIVES:

- To develop advanced vocabulary for effective communication and reading skills.
- To build an enhanced level of logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.
- To apply data structures and algorithms in problem solving.

LIST OF EXERCISES:

1. **English – Phase II Advanced**

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. **Logical Reasoning – Phase II Advanced**

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. **Quantitative Ability - Phase II Advanced**

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. **Automata Fix – Phase II**

Logical, Compilation and Code reuse

5. **Automata - Phase II**

Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop advanced vocabulary for effective communication and reading skills.

CO2: Build an enhanced level of logical reasoning and quantitative skills.

CO3: Develop error correction and debugging skills in programming.

CO4: Apply data structures and algorithms in problem solving.

SEMESTER VII

22CS701	CRYPTOGRAPHY AND CYBER SECURITY (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Understand the fundamentals of network security and security architecture.
- Learn the different symmetric key cryptographic algorithms.
- Study the various asymmetric key cryptographic algorithms and techniques.
- Know the importance of message authentication and integrity.
- Learn the various cyber-crimes and cyber security.

UNIT I	INTRODUCTION TO SECURITY	9+6

Computer Security Concepts – The OSI Security Architecture – Security Attacks – Security Services and Mechanisms – A Model for Network Security – Classical encryption techniques: Substitution techniques, Transposition techniques, Steganography – Foundations of modern cryptography: Perfect security - Information Theory - Product Cryptosystem – Cryptanalysis.

List of Exercise/Experiments:

1. Perform encryption, decryption using the following substitution techniques

(i) Ceaser cipher, (ii) playfair cipher iii) Hill Cipher iv) Vigenere cipher

2. Perform encryption and decryption using following transposition techniques

i) Rail fence ii) row & Column Transformation

UNIT II	SYMMETRIC CIPHERS	9+6

Number theory – Algebraic Structures – Modular Arithmetic - Euclid_s algorithm – Congruence and matrices – Group, Rings, Fields, Finite Fields

SYMMETRIC KEY CIPHERS: SDES – Block Ciphers – DES, Strength of DES – Differential and linear cryptanalysis – Block cipher design principles – Block cipher mode of operation – Evaluation criteria for AES – Pseudorandom Number Generators – RC4 – Key distribution.

List of Exercise/Experiments:

1. Apply DES algorithm for practical applications.
2. Apply AES algorithm for practical applications.

UNIT III	ASYMMETRIC CRYPTOGRAPHY	9+6

MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes – Primality Testing – Factorization – Euler‘s totient function, Fermat‘s and Euler‘s Theorem – Chinese Remainder Theorem – Exponentiation and logarithm ASYMMETRIC KEY CIPHERS: RSA cryptosystem – Key distribution – Key management – Diffie Hellman key exchange -– Elliptic curve arithmetic – Elliptic curve cryptography.

List of Exercise/Experiments:

1. Implement RSA Algorithm using HTML and JavaScript.
2. Implement the Diffie-Hellman Key Exchange algorithm for a given problem.

3. Calculate the message digest of a text using the SHA-1 algorithm.

UNIT IV	INTEGRITY AND AUTHENTICATION ALGORITHMS	9+6

Authentication requirement – Authentication function – MAC – Hash function – Security of hash function: HMAC, CMAC – SHA – Digital signature and authentication protocols – DSS – Schnorr Digital Signature Scheme – ElGamal cryptosystem – Entity Authentication: Biometrics, Passwords, Challenge Response protocols – Authentication applications – Kerberos MUTUAL TRUST: Key management and distribution – Symmetric key distribution using symmetric and asymmetric encryption – Distribution of public keys – X.509 Certificates

List of Exercise/Experiments:

1. Implement the SIGNATURE SCHEME - Digital Signature Standard.
2. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w

UNIT V	CYBER CRIMES AND CYBER SECURITY	9+6

Cyber Crime and Information Security – classifications of Cyber Crimes – Tools and Methods – Password Cracking, Keyloggers, Spywares, SQL Injection – Network Access Control – Cloud Security – Web Security – Wireless Security

List of Exercise/Experiments:

1. Automated Attack and Penetration Tools

a.Exploring N-Stalker, a Vulnerability Assessment Tool

2. Defeating Malware

i) Building Trojans ii) Rootkit Hunter

TOTAL: 45+30=75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand and explain the basics of computer security, including security architecture, attacks, services, mechanisms, and classical encryption techniques.

CO2: Use substitution and transposition techniques for encryption and decryption, and understand the basics of modern cryptography.

CO3: Analyze and implement symmetric key ciphers like DES and AES, and understand related math concepts.

CO4: Implement and evaluate asymmetric cryptographic methods like RSA and Diffie-Hellman.

CO5: Develop and apply algorithms for integrity and authentication, including digital signatures and hash functions.

CO6: Identify and analyze cyber crimes and apply security measures for networks, cloud, and wireless systems.

TEXT BOOK:

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson education 8th Edition, 2020.
2. Nina Godbole, Sunit Belapure, ¯Cyber Security: Understanding Cyber crimes, Computer Forensics and Legal Perspectives‖, First Edition, Wiley India, 2011.

REFERENCES:

1. C K Shyamala, N Harini and Dr. T R Padmanabhan: Cryptography and Network Security, Wiley India Pvt.Ltd 1st Edition,2011
2. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, Tata Mc Graw Hill, 2015.
3. Wade Trappe, Lawrence C. Washington: Introduction to Cryptography with Coding Theory, 3rd Edition, 2020.
4. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, "Security in Computing", Fifth Edition, Prentice Hall, New Delhi, 2015.

LIST OF EQUIPMENTS:

C/C++/JAVA

22CS702	DATA ANALYTICS (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Explain the fundamentals of big data and data analytics
- Discuss the Hadoop framework
- Explain about exploratory data analysis and data manipulation tools and use it for developing applications
- Analyse and interpret streaming data
- Discuss various applications of data analytics

UNIT I	INTRODUCTION	9+6

Evolution of Big Data- Definition of Big Data-Challenges with Big Data- Traditional Business Intelligence (BI) versus Big Data- Introduction to big data analytics- Classification of Analytics- Analytics Tools- Importance of big data analytics.

List of Exercise/Experiments:

1. Download, install and explore the features of R/Python for data analytics.
2. Working with Numpy arrays

UNIT II	HADOOP FRAMEWORK	9+6

Introducing Hadoop- RDBMS versus Hadoop- Hadoop Overview-HDFS (Hadoop Distributed File System)- Processing Data with Hadoop- Managing Resources and Applications with Hadoop YARN - Interacting with Hadoop Ecosystem

List of Exercise/Experiments:

1. Working with Pandas data frames
2. Basic plots using Matplotlib

UNIT III	EXPLORATORY DATA ANALYSIS	9+6

EDA fundamentals – Understanding data science – Significance of EDA – Making sense of data – Comparing EDA with classical and Bayesian analysis – Software tools for EDA -Data transformation techniques - Introduction to NoSQL – MongoDB: RDBMS Vs MongoDB – Data Types – Query Language – Hive – Hive Architecture – Data Types – File Formats - Hive Query Language (HQL) – RC File Implementation – User Defined Functions.

List of Exercise/Experiments:

1. Statistical and Probability measures - Frequency distributions, Mean, Mode, Standard Deviation, Variability, Normal curves, Correlation and scatter plots, Correlation coefficient, Regression.
2. Use the standard benchmark data set for performing the following:
   a) Univariate Analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis.
   b) Bivariate Analysis: Linear and logistic regression modelling.
   c) Multiple Regression Analysis
   d) Compare the results of the above analysis for the two data sets.

UNIT IV	MINING DATA STREAMS	9+6

The data stream model – stream queries-sampling data in a stream-general streaming problem filtering streams-analysis of filtering- dealing with infinite streams- Counting Distance Elements in a Stream – Estimating Moments – Counting Ones in Window – Decaying Windows.

List of Exercise/Experiments:

1. Apply and explore various plotting functions on any data set.
2. Implement the following algorithms on real time stream data sets.

Support Vector Machine

Decision tree classifier

Clustering Algorithms

UNIT V	APPLICATIONS	9+6

Application: Sales and Marketing – Industry Specific Data Mining – microRNA Data Analysis Case Study – Credit Scoring Case Study – Data Mining Non tabular Data.

List of Exercise/Experiments:

1. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using the standard Heart Disease Data Set. You can use Java/Python ML library classes/API
2. Mini Project: The project should contain the following components

- Realtime dataset
- Data preparation & Transformation
- Handling missing Data
- Data Storage
- Algorithm for data analytics
- Data visualization: Charts, Heatmap, Crosstab, Treemap

TOTAL: 45 + 30 = 75 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of big data, including its definition, challenges, and differences from traditional BI.

CO2: Use R and Python to analyze data sets and perform basic statistical analyses and visualizations.

CO3: Compare the functionalities of RDBMS and Hadoop in data processing and resource management.

CO4: Conduct exploratory data analysis (EDA) and understand its role compared to classical and Bayesian analysis.

CO5: Implement various data stream mining algorithms for real-time data analysis.

CO6: Develop real-world applications of big data analytics, covering data preparation, storage, analytics, and visualization.

TEXT BOOKS:

1. Subhashini Chellappan, Seema Acharya, ―Big Data and Analytics‖, 2nd edition, Wiley Publications, 2019.
2. Suresh Kumar Mukhiya and Usman Ahmed, ―Hands-on Exploratory Data Analysis with Python‖, Packt publishing, March 2020.
3. Jure Leskovek, Anand Rajaraman and Jefrey Ullman,‖ Mining of Massive Datasets. v2.1‖, Cambridge University Press, 2019.
4. Glenn J. Myatt, Wayne P. Johnson, Making Sense of Data II: A Practical Guide To Data Visualization, Advanced Data Mining Methods, and Applications, Wiley 2009.

REFERENCES:

1. Nelli, F., Python Data Analytics: with Pandas, NumPy and Matplotlib, Apress, 2018.
2. Bart Baesens," Analytics in a Big Data World: The Essential Guide to Data Science and its Applications", John Wiley & Sons, 2014.
3. Min Chen, Shiwen Mao, Yin Zhang, Victor CM Leung, Big Data: Related Technologies, Challenges and Future Prospects, Springer, 2014.
4. Michael Minelli, Michele Chambers, Ambiga Dhiraj, ―Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends‖, John Wiley & Sons, 2013.
5. Marcello Trovati, Richard Hill, Ashiq Anjum, Shao Ying Zhu, ―Big Data Analytics and cloud computing – Theory, Algorithms and Applications‖, Springer International Publishing, 2016.

LIST OF EQUIPMENTS:

R / Python

ELECTIVE VERTICALS
CYBER SECURITY

22CS901	ETHICAL HACKING (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Understand Information Security, Cyber threats, attacks, web security.
- Know about different modes of hacking tools and phases of penetration tests and Methodologies.
- Gain the knowledge of the use and availability of tools to support an ethical hack.
- Gain the knowledge of interpreting the results of a controlled attack.

UNIT I	FUNDAMENTALS OF ETHICAL HACKING	6+6

Overview of Cyber threats – Data and Network Security Attacks – Threats: MAC spoofing – Access control Network protocol and services-Hacking terms - Ethical Hacking overview –Modes of Ethical Hacking – Ethics and Legality.

List of Exercise/Experiments:

1. Setup a honey pot and monitor the honey pot on network

2. Write a script or code to demonstrate SQL injection attacks
3. Write a code to demonstrate DoS attacks

UNIT II	HACKING METHODOLOGY RECONNAISSANCE	6+6

Foot printing: Reconnaissance - Footprinting theory – Penetration test – Phases of Penetration test - Methods of Footprinting – Network Information gathering process – Terminologies of Foot printing -Footprinting through search engine directives - Whois tool - NetCraft - Extract Information from DNS - Foot printing from Email servers - Shodan - Dig - MetaGooFil - Social Engineering.

List of Exercise/Experiments:

1. Performing footprinting using Google Hacking, website information, information about an archived website, to extract contents of a website, to trace any received email, to fetch DNS information.
2. Create a social networking website login page using phishing techniques

UNIT III	SCANNING AND ENUMERATION	6+6

Scanning: Concept of Nmap - - Port scanning with Nmap - Subnet - Scanning IPs with Nmap Pings and Ping sweeps - Port - Three way handshake - NmapSyn scanning - Nmap TCP Scan - Nmap UDP Scan - Bypass of IPS and IDS - Nmap Script Engine Enumeration: Service Fingerprinting – Vulnerability Scanners – Basic Banner Grabbing – Common Network services - SMTP - DNS - RPCBIND Enumeration - SMB - NetBIOS

List of Exercise/Experiments:

1. Implement Passive scanning, active scanning, session hijacking, cookies extraction using Burp suit tool
2. Use port scanning. network scanning tools,IDS tool, sniffing tool and generate reports.

UNIT IV	SYSTEM AND NETWORK VULNERABILITY	6+6

Metasploit - Penetration testing with framework Metasploit - Scan services to identify vulnerabilities - Scan FTP services - Scan HTTP services - Exploitation - Post exploitation techniques - Meterpreter - Rootkit - Backdoor - Password hashes - Privilege Escalation - Scanning vulnerable services with Nessus

List of Exercise/Experiments:

1. Penetration Testing using Metasploit and metasploitable
2. Creating a simple keylogger in python
3. Creating a virus
4. Creating a trojan.
5. Install rootkits and study variety of options

UNIT V	SOFTWARE VULNERABILITY (OWASP 10)	6+6

Fundamentals of OWASP Zed Attack Proxy (ZAP) - Web app vulnerability scan - Code Injection Attacks - Broken Authentication - Sensitive Data Exposure - XML External Entities – Broken Access Control – Security misconfiguration – Website pen testing - Cross Site Scripting (XSS) - Insecure Deserialization - Using Components with known vulnerabilities - Insufficient logging and monitoring.

List of Exercise/Experiments:

1. Install jcrypt tool (or any other equivalent) and demonstrate Asymmetric, Symmetric Crypto algorithm, Hash and Digital/PKI signatures studied in theory Network Security And Management

2. Hacking a website by Remote File Inclusion
3. Disguise as Google Bot to view hidden content of a website
4. To use Kaspersky for Lifetime without Patch

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of information security, threats and its attacks
CO2: Understand the fundamentals of ethical hacking with the hacking methodologies
CO3: Analyze the phases of the penetration test with the methods
CO4: Understand the vulnerabilities and use the frameworks to identify vulnerabilities by service scan
CO5: Understand the web security issues with the fundamentals of OWASP

TEXTBOOKS:

1. McClure, S., Scambray, J. and Kurtz, G., 2012. Hacking Exposed Network Security Secrets and Solutions. New York: McGraw-Hill.
2. Engebretson, P., 2013. The Basics Of Hacking And Penetration Testing. Amsterdam: Syngress, an imprint of Elsevier.

REFERENCES:

1. Zaid Sabih, Learn Ethical Hacking from Scratch, 2018, PACKT publishing, ISBN: 978-1-78862-205-9
2. Harsh Bothra, Hacking be a hacker with ethics, Khanna Publishing, 2016, ISBN: 978-03-86173-05-8

LIST OF SOFTWARE:

1. Metasploit Framework (MSF)
2. WireShar
3. Nmap
4. John the Ripper
5. Burp suite or OWASP ZAP
6. Kali Linux

22CS902	SOCIAL NETWORK SECURITY (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Learn the Concepts of Social Network Security
- Understand the methods of Social Network Anonymization
- Learn the techniques for security and privacy in social networks
- Learn the security challenges in social networks
- Explore the Tools to learn about the social network security implementation

UNIT I	INTRODUCTION TO SOCIAL NETWORK SECURITY	6+6

Introduction – Social Networking Applications – Social media Websites – Social Network Representation -Building Social Authority -Privacy and Information sharing – Controlling Application privacy – Cybercrime – Information Leakage – False information – Content Management in Social Networks

List of Exercise/Experiments:

1. Explore a Social network analysis tools to learn about the users and networks

2. Learn a program / tool to illustrate information leakage

UNIT II	SOCIAL NETWORK ANONYMIZATION	6+6

Social Networks - Privacy in Social Networks – Social Network Representation - Social Network Analysis - Data Anonymization – Challenges in Anonymization – Privacy preservation – Social Network Anonymization Factors – Anonymization Algorithms - Link Anonymization techniques –Background Knowledge Attacks – Anonymity in Modern Social Networks

List of Exercise/Experiments:

1. Experiment a link anonymization technique
2. Explore ARX anonymization tool

UNIT III	ANALYZING AND SECURING SOCIAL NETWORKS	6+6

Supporting Technologies - Aspects of Analyzing and Securing Social Networks - Techniques and Tools for Social Network Analytics - Social Network Analytics and Privacy Considerations - Access Control and Inference for Social Networks - Social Media Integration and Analytics Systems - Social Media Application Systems - Secure Social Media Systems - Secure Social Media Directions.

List of Exercise/Experiments:

1. Implement a program for network access control to illustrate malware attacks
2. Create a simple social network application to show authentication mechanisms
3. Create an application for the following scenario: ¯Social networking users are presented with two apparently similar emails or websites. They must first identify the differences between them and then decide which one is a scam attempting to steal their information or money.‖

UNIT IV	SECURITY CHALLENGES IN SOCIAL NETWORKS	6+6

Identity manipulation – Threats from third party applications - Trust in Social Networking Sites - Viruses, Phishing Attacks and Malwares-Tracking users – Privacy of Data – Identity Federation Challenges –Social media threats – Location disclosure – Spoofing – Profile cloning – Fake product sale – Cyber bullying – Prevention Strategies

List of Exercise/Experiments:

1. Implement a program in python to estimate trust of social network users group
2. Write a SQL injection program in python/JAVA to handle session hijacking
3. Create an application using any social network platform to demonstrate profile cloning concept.

UNIT V	SOCIAL NETWORK SECURITY TOOLS	6+6

Analysis Tools for Social Media - AutoMap – Gephi – ORA Lite – ORA Pro – Wolfram Alpha – Social Media Data Collection –Blog Trackers –Crowd Tangle – MalTego – Pulse – SCRAAWL – Fact and Image Trackers – Google Fact Check Tools – Bot Mitigation – BotSlayer – Social Cyber Security

List of Exercise/Experiments:

1. Perform fact checking of social networking content using google fact checking tools
2. Explore a tool that helps protect websites from bot traffic and bot attacks.
3. Create a fake news tracker program to collect, detect and help visualize fake news data from any social network

TOTAL:30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop security applications of social networks.
CO2: Implement data anonymization techniques
CO3: Analyze and secure social networks
CO4: Handle security challenges in social networks
CO5: Develop security tools for social networks
CO6: Create newer social networking applications

TEXTBOOKS:

1. Brij B. Gupta, Somya Ranjan Sahoo, ―Online Social Networks Security-Principles, Algorithm, Applications, and Perspectives‖, First Edition, 2021.
2. Bhavani Thuraisingham, SatyenAbrol, Raymond Heatherly, Murat Kantarcioglu, Vaibhav Khadilkar, Latifur Khan, ―Analyzing and Securing Social Networks‖, First Edition, 2020.
2. B. K. Tripathy, Kiran Baktha, ―Security, Privacy, and Anonymization in Social Networks: Emerging Research and Opportunities‖, IGI Global Publication, 2019.
3. Michael Cross, ―Social Media Security, Leveraging Social Networking While Mitigating Risk‖, Elsevier Publication, First Edition, 2013.

REFERENCES:

1. El-Sayed M. El-Alfy ; Mohamed Eltoweissy ;Errin W. Fulp ; Wojciech Mazurczyk, ―Nature-Inspired Cyber Security and Resiliency: Fundamentals, Techniques and Applications‖, IET Publication, 2019.
2. https://sites.google.com/view/social-cybersec/tools?pli=1
3. Yaniv Altshuler, ―Security and Privacy in Social Networks‖, Springer , 2013.

LIST OF EQUIPMENTS:

1. **Software Required:**
   Python
2. **Software Tools Required:**
   Shield Square
   BotSlayer
   GOOGLE FACT CHECK TOOLS
   ORA-PRO

22CS903	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand block chain system‘s fundamental components, how they fit together and examine a decentralization using block chain.
- To explain how Crypto currency works.
- To explain the components of Ethereum and Programming Languages for Ethere
- To study the basics of Web3 and Hyper ledger.
- To give an insight of alternative block chains and its emerging trends.

UNIT I	INTRODUCTION TO BLOCKCHAIN	9

History of Blockchain – Types of Blockchain – Consensus – Decentralization using

Blockchain – Blockchain and Full Ecosystem Decentralization – Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography - public and private keys - Elliptic curve cryptography - Discrete logarithm problem in ECC.

UNIT II	INTRODUCTION TO CRYPTOCURRENCY	9

Bitcoin – Digital Keys and Addresses – Transactions – Mining – Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin. Case study - Web3j.

UNIT III	ETHEREUM	9

The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code – Blocks and Blockchain – Fee Schedule – Supporting Protocols - Solidity Language.

UNIT IV	WEB3 AND HYPERLEDGER	9

Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger – Case study - Corda.

UNIT V	ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS	9

Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research - Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand the technology components of Blockchain and how it works behind the scenes.

CO2: Understand the Bitcoin and its limitations by comparing with other alternative coins.

CO3: Develop deep understanding of the Ethereum model, its consensus model, code execution.

CO4: Understand the architectural components of a Hyperledger and its development framework.

CO5: Explore the alternative blockchains and its emerging trends.

CO6: Understand blockchain technology, including Bitcoin, Ethereum, and alternative blockchains.

TEXT BOOKS:

1. Imran Bashir, ―Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained‖, Second Edition, Packt Publishing, 2018.
2. Arshdeep Bahga, Vijay Madisetti, ―Blockchain Applications: A Hands-On Approach‖, VPT, 2017.

REFERENCES:

1. Andreas Antonopoulos, Satoshi Nakamoto, ―Mastering Bitcoin‖, O'Reilly Publishing, 2014.
2. Roger Wattenhofer, ―The Science of the Blockchain‖ CreateSpace Independent Publishing Platform, 2016.
3. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, ―Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction‖, Princeton University Press, 2016.
4. Alex Leverington, ―Ethereum Programming‖, Packt Publishing, 2017.
5. Antony Lewis ―The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies

and the Technology that Powers Them‖, Mango Publishing 2018.
6. Andreas M. Antonopoulos, ¯Mastering Bitcoin: Programming the Open Block chain‖, O'Reilly Publishing, 2017.
7. Massimo Ragnedda, Giuseppe Destefanis, ¯Blockchain and Web 3.0: Social, Economic, and Technological Challenges‖, Routledge, 2019.

22CS904	CLOUD AND DATA SECURITY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Learn the basics of cloud security, including the shared responsibility model and identity management.
- Set up a secure cloud infrastructure with features like virtual private clouds and security groups.
- Develop skills for identifying and managing security incidents in the cloud, adhering to best practices.
- Safeguard application data at rest and in transit using encryption.
- Understand the features of Database Security and Security in Data Warehouses.

UNIT I	INTRODUCTION TO SECURITY IN CLOUD	9

Introduction to Security, Security in the Cloud, Security design principles, Shared responsibility model, Activity: Shared Responsibility Model, Identity and Access Management (IAM) fundamentals, Authenticating and Authorizing with IAM, Examples of authorizing with IAM, Additional authentication and access management services, Using Organizations.

UNIT II	SECURING INFRASTRUCTURE	9

Structure of a three-tier web application, virtual private cloud (VPC), Setting up public and private subnets and internet protocols, Security groups, Network access control lists (ACLs), Load balancers, Protecting compute resources- Cloud service models: IaaS, PaaS, SaaS.

UNIT III	INCIDENT RESPONSE AND RISK MANAGEMENT	9

Identifying an incident, Services that support the discovery and recognition phase, AWS Config and AWS Lambda, Services that support the resolution and recovery phase, Best practices for handling an incident.

UNIT IV	SECURING CLOUD: DATA SECURITY	9

Overview of Data Security in Cloud Computing- Common Risks with Cloud Data Security- Data Encryption: Applications and Limits- Cloud Data Security: Sensitive Data Categorization- Authentication and Identity- Data Categorization and the Use of Data Labels- Cloud Data Storage.

UNIT V	DATABASE SECURITY	9

Database Security: Recent Advances in Access Control, Access Control Models for XML, Database Issues in Trust Management and Trust Negotiation, Security in Data Warehouses and OLAP Systems.

TOTAL:45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand security principles in cloud computing.
CO2: Implement infrastructure security measures in cloud environments.
CO3: Demonstrate incident response and risk management techniques in cloud systems.

CO4: Apply data security practices to protect cloud-based information.
CO5: Evaluate database security mechanisms for cloud-based data management.
CO6: Integrate ethical considerations into cloud security practices.

TEXTBOOKS:

1. Tim Mather, Subra Kumaraswamy, Shahed Latif, ¯Cloud Security and Privacy, An Enterprise Perspective on Risks and Compliance‖, Oreilly Media 2009.
2. Vic (J.R.) Winkler, ¯Securing the Cloud, Cloud Computer Security Techniques and Tactics‖, Syngress, April 2011.

REFERENCES:

1. Michael Gertz, Sushil Jajodia,‖ Handbook on Database security: Applications and Trends‖,Springer, 2010.
2. John R. Vacca, ¯Cloud Computing Security‖, CRC Press, 2016.
3. Giulio D'Agostino, ¯Data Security in Cloud Computing, Volume I‖, Momentum Press, 2019.

22CS905	DIGITAL AND MOBILE FORENSICS	L	T	P	C
		3	0	0	3

OBJECTIVES:
The Course will enable learners to:

- Learn how to acquire digital forensic evidence.
- Learn how to investigate different digital artifacts and write reports
- Understand network forensics processes and procedures
- Understand mobile forensics processes and procedures.
- Able to analyze SIM cards and analyze mobile file systems.

UNIT I	ACQUIRING DIGITAL FORENSICS EVIDENCE	9

Types of Computer-Based Investigations - The Forensic Analysis Process- Acquisition of Evidence - Computer Systems- Case Study: Use The Sleuth Kit and Autopsy to retrieve data from file disk

UNIT II	DIGITAL FORENSICS INVESTIGATION& REPORTING	9

Computer Investigation Process-Windows Artifact Analysis-RAM Memory Forensic Analysis-Email Forensics-Investigation Techniques-Internet Artifacts- Case Study: Use DumpIt to acquire memory

UNIT III	NETWORKING FORENSICS	9

Characteristics in the network world-Identifying threats to the enterprise-Data breach surveys-Defining network forensics-Differentiating between computer forensics and network forensics-Digital footprints-Collecting network traffic using tcp dump-Collecting network traffic using Wireshark-Collecting network logs-Acquiring memory using FTK Imager-Tapping into network traffic-Packet sniffing and analysis using Wireshark-Packet sniffing and analysis using Network Miner

UNIT IV	MOBILE FORENSICS FUNDAMENTALS	9

Mobile Devices vs. Computer Devices in the World of Forensics-Living in the Cloud: The Place to Hide and Store Mobile Data-Preparing, Protecting, and Seizing Digital Device Evidence

UNIT V	ANALYSING MOBILE INTERNALS	9

Analyzing SIM Cards - Advanced Android Analysis - Advanced iOS Analysis-Case Study: Use Andriller or equivalent to extract data from Android

	TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand how to acquire digital forensic evidence.
CO2: Understand how to investigate different digital artifacts and write reports
CO3: Understand network forensics processes and procedures.
CO4: Understand mobile forensics processes and procedures.
CO5: Analyze SIM cards and analyze mobile file systems.
CO6: Understand Digital forensic techniques for comprehensive documentation and reporting.

TEXTBOOKS:

1. William Oettinger, ¯Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence‖, Packt Publishing, 1st Edition, 2020
2. Samir Datt, ¯Learning Network Forensics‖, Packt Publishing, 1st Edition, 2016

REFERENCES:

1. Lee Reiber, ¯Mobile Forensic Investigations: A Guide to Evidence Collection, Analysis, and Presentation‖, McGraw Hill, 2nd Edition, 2018.
2. Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty, ¯Practical Mobile Forensics‖, Packt Publishing, 3rd Edition, 2018
3. Gerard Johansen, ¯Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response‖, Packt Publishing, 3rd Edition, 2022

22CS906	VULNERABILITY ANALYSIS AND PENETRATION TESTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To learn the tools that can be used to perform information gathering.
- To identify operating systems, server applications to widen the attack surface and perform vulnerability assessment activity and exploitation phase.
- To learn how vulnerability assessment can be carried out by means of automatic tools or manual investigation.
- To learn the web application attacks starting from information gathering to exploitation phases.
- To learn how to metasploit and meterpreter are used to automate the attacks and penetration testing techniques.

UNIT I	INTRODUCTION	9

Introduction- Vulnerability Assessment- Understanding the Risks Posed by Vulnerabilities Detecting Vulnerabilities via Security Technologies- Categories of Penetration Testing - Types of Penetration Test – Structure of Penetration Testing Reports - Information Gathering Techniques - Active, Passive and Sources of Information Gathering - Approaches and Tools - Traceroutes, Neotrace, Whatweb, Netcraft, Xcode Exploit Scanner and NSlookup. Host discovery - Scanning for open ports and services - Types of Port.

UNIT II	NETWORK VULNERABILITY ASSESSMENT	9

Project Scoping-Assessing Vulnerability assessment timeline-NVAT-Prioritizing risks and threats Assessment Methodology-Top down and Bottom up Examination-Case study with

assessment report- Case Study: Web Based Email Attacks.		
UNIT III	**MOBILE APPLICATION SECURITY**	**9**
Types of Mobile Application Key challenges in Mobile Application and its impact Need for mobile application penetration testing Mobile application penetration testing methodology Android and ios Vulnerabilities - OWASP mobile security risk - Exploiting WM - BlackBerry Vulnerabilities - Vulnerability Landscape for Symbian - Exploit Prevention - Handheld Exploitation		
UNIT IV	**WIRELESS NETWORK VULNERABILITY ANALYSIS**	**9**
WLAN and its inherent insecurities Bypassing WLAN Authentication uncovering hidden SSIDs MAC Filters Bypassing open and shard authentication - Attacking the client caffe latte attack Deauthenticating the client cracking WEP with the hirte attack AP-less WPA cracking - Advanced WLAN Attacks Wireless eavesdropping using MITM session hijacking over wireless - WLAN Penetration Test Methodology		
UNIT V	**PENETRATION TESTING**	**9**
Introduction to Kali and Backtrack-Linux tools – Attack Machine- Phases of penetration test-reconnaissance extracting information from DNS-scanning-pings and ping sweeps-port scanning- NMap-Vulnerability scanning.		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand vulnerability assessment principles and methods.
CO2: Analyze network vulnerabilities and prioritize risks.
CO3: Evaluate mobile application security challenges and methodologies.
CO4: Assess wireless network vulnerabilities and conduct penetration testing.
CO5: Apply penetration testing methodologies using appropriate tools.
CO6: Demonstrate ethical considerations in penetration testing practices.

TEXTBOOKS:

1. Rafay Baloch, Ethical Hacking and Penetration Testing Guide, CRC Press, 2015.
2. Dr. Patrick Engebretson, The Basics of Hacking and Penetration Testing Ethical Hacking and Penetration Testing made easy, Syngress publications, Elsevier, 2013.

REFERENCES:

1. Steve Manzuik, Andre Gold, Chris Gatford, ¯Network Security Assessment from Vulnerability to Patch‖, Syngress Publishing Incorporation, 2007.
2. Mastering Modern Web Penetration Testing By Prakhar Prasad, October 2016.
3. Kali Linux 2: Windows Penetration Testing, By Wolf Halton, Bo Weaver, June 2016.

22CS927	ENGINEERING SECURE SOFTWARE SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Know the importance and need for software security.
- Know about various attacks.
- Learn about secure software design.
- Understand risk management in secure software development.
- Know the working of tools related to software security.

UNIT I	NEED OF SOFTWARE SECURITY AND LOW-LEVEL ATTACKS	9
Software Assurance and Software Security – Threats to software security – Sources of software insecurity – Benefits of Detecting Software Security – Properties of Secure Software – Memory Based Attacks: Low-Level Attacks Against Heap and Stack – Defense Against Memory-Based Attacks		
UNIT II	**SECURE SOFTWARE DESIGN**	**9**
Requirements Engineering for secure software- SQUARE process Model- Tools- SQUARE Final Results- Requirement Elicitation and Prioritization- The Critical Role of Architecture and Design- Issues and Challenges - Software Characterization - Threat Analysis - Architectural Vulnerability Assessment.		
UNIT III	**SECURITY RISK MANAGEMENT**	**9**
Risk Management Life Cycle – Risk Profiling – Risk Exposure Factors – Risk Evaluation and Mitigation – Risk Assessment Techniques – Threat and Vulnerability Management.		
UNIT IV	**SECURITY TESTING**	**9**
Software Security Testing- Contrasting Software Testing and Software Security Testing- Functional Testing- Risk-Based Testing-Secure Software Development Life Cycle- Unit Testing, Testing Libraries and Executable Files, Integration Testing, System Testing.		
UNIT V	**SECURE PROJECT MANAGEMENT**	**9**
Governance and security – Adopting an enterprise software security framework – Security and project management – Maturity of Practice-Case Study: Implement the SQL Injection attack and Buffer Overflow attack..		
	TOTAL: 45 PERIODS	

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Analyze low-level memory attacks and implement corresponding defenses.
CO2: Implement requirements engineering and architectural vulnerability assessment.
CO3: Evaluate and mitigate risks throughout the software development lifecycle.
CO4: Implement various testing techniques to ensure software security
CO5: Apply enterprise security frameworks in project governance.
CO6: Analyze case studies to understand real-world security threats in project management.

TEXTBOOKS:

1. Julia H. Allen, Software Security Engineering, Pearson Education, 2009.
2. Evan Wheeler, Security Risk Management: Building an Information Security Risk Management Program from the Ground Up, First edition, Syngress Publishing, 2011.

REFERENCES:

1. Rajib Mall,‖ Fundamentals Of Software Engineering‖, 5th Edition, PHI Learning, 2018.
2. Jon Erickson,‖Hacking:The Art of Exploitation‖,2nd Edition, No Starch Press, 2008.
3. Mike Shema,‖Hacking Web Apps: Detecting and Preventing Web Application Security Problems‖, First Edition, Syngress Publishing,2012.
4. Bryan Sullivan and Vincent Liu,‖Web Application Security, A Beginner‘s Guide‖,Kindle Edition, McGraw Hill,2012.
5. Lee Allen,‖Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide(Open Source:Community Experience Distilled)‖, Kindle Edition, Packt Publishing,2012.
6. Chris Wysopal, Lucas Nelson, Dino Dai Zovi, and Elfriede Dustin, ¯The Art of Software Security Testing: Identifying Software Security Flaws (Symantec Press)‖, Addison-Wesley Professional, 2006.

22CS928	NETWORK DESIGN AND PROGRAMMING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basic networking principles.
- Explore various networking devices and protocols required for network design and management.
- Gain knowledge in logical and physical designs for scalable LAN and WAN networks
- Study two novel networking technologies: SDN and DTN.
- Learn network programming in UNIX C.

UNIT I	NETWORKING PRINCIPLES	9

Advanced multiplexing – Code Division Multiplexing, DWDM and OFDM – Shared media networks – Collision detection and collision avoidance, Hidden and Exposed Terminals – Switched networks – Datagrams, Virtual circuits, Cell switching and Label switching – Wireless Networks – Infrastructure based, ad hoc and hybrid – End to end semantics – Connectionless, Connection oriented, Wireless Scenarios –Applications, Quality of Service – End to end level and network level solutions.

UNIT II	PHYSICAL NETWORK DESIGN	9

LAN cabling topologies – Ethernet Switches – High speed and Gigabit and 10Gbps – Building cabling topologies and Campus cabling topologies – Routers, Firewalls and L3 switches –Remote Access Technologies and Devices – Modems and DSLs – SLIP and PPP - WAN Design and Enterprise Networks – Core networks, distribution networks and access networks

UNIT III	LOGICAL DESIGN AND MANAGEMENT	9

IPv4 and IPv6 Dynamic Addressing –Hierarchical routing – VLSMand CIDR – Transition from IPv4 to IPv6 – NAT and DHCP – Static and Dynamic routes – RIP, OSPF and BGP – VPN –RMON and SNMP

UNIT IV	INNOVATIVE NETWORKS	9

Software Defined Networks – Evolution of switches and control planes – Centralized and distributed data and control planes – OpenFlow and SDN Controllers – Network Function Virtualization – Needs of the Data Centres – SDN solutions for data centres - Delay Tolerant Networks – Overlay architecture – Bundle Protocol – Opportunistic routing and Epidemic routing

UNIT V	NETWORK PROGRAMMING IN UNIX C	9

Socket address structures – Byte ordering and byte manipulation functions – Elementary TCP sockets – socket, connect, bind, listen, accept and close functions – TCP client and server – Elementary UDP sockets –recvfrom and sendto functions , connect function with UDP – Raw sockets – Client-server design alternatives – Iterative and Concurrent servers.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand advanced multiplexing methods like DWDM and OFDM.
CO2: Evaluate network protocols for efficient data transmission.
CO3: Develop logical and physical designs for scalable LAN and WAN networks.
CO4: Apply strategies for transitioning from IPv4 to IPv6.
CO5: Investigate Software Defined Networks and Delay Tolerant Networks.
CO6: Gain proficiency in network programming using socket APIs in C.

TEXTBOOKS:

1. Larry Peterson and Bruce Davie, ¯Computer Networks: A Systems Approach‖, 5th edition, Morgan Kauffman, 2011
2. ParitoshPuri, M.P.Singh,‖A survey paper on routing in delay tolerant networks‖, International Conference on Information and Computer Networks (ISCON), 2013.

REFERENCES:

1. Paul Goransson, Chuck Black, ¯Software Defined Networks: A Comprehensive Approach‖, Morgan Kauffman, 2016.
2. W.Richard Stevens, Bill Fenner and Andrew M Rudoff, ¯Unix Network Programming: The Sockets Networking API: Volume 1 ¯, 3rd Edition, Addison Wesley, 2003.
3. Ying Dar Lin, Ren-Hung Hwang and Fred Baker, ¯Computer Networks: An Open Source Approach‖, McGraw Hill, 2011.

22CS929	FAULT TOLERANT COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Create understanding of the fundamental concepts of fault tolerance systems
- Learn basic techniques for achieving fault tolerance in hardware
- Have in-depth understanding in software fault tolerance systems
- Gain knowledge in design & testing of fault tolerance systems
- Develop skills in modelling and evaluating fault tolerant architectures in Real time systems

UNIT I	INTRODUCTION	9

Faults, Errors and Failures - Levels of Fault tolerance - Dependability measures - Dependability evaluation - Fault Tolerant techniques - Hardware redundancy - Information redundancy - Software redundancy- Time redundancy -Software Testing.

UNIT II	HARDWARE FAULT TOLERANCE	9

The Rate of Hardware Failures - Failure Rate, Reliability, and Mean Time to Failure - Canonical and Resilient Structures - Poisson Processes - Markov Models Fault-Tolerance Processor-Level Techniques - Byzantine failures.

UNIT III	SOFTWARE FAULT TOLERANCE	9

Single-Version Fault Tolerance – N Version programming - Recovery Block Approach - Exception-Handling - Software Reliability Models - Check pointing - Optimal Checkpointing - Checkpointing in Distributed Systems, Shared-Memory Systems and Real-Time Systems.

UNIT IV	DESIGN DIVERSITY & TESTING	9

Fault tolerant Control and coordination algorithms design – F-T system abstractions and functions- Pitfalls- Practical application- Modeling and analysing F-T Distributed systems - Software fault insertion testing- Fault manager- Categorization of Software faults, Errors, and failures- SIFT methodology and Test plans

UNIT V	FAULT TOLERANCE IN REAL TIME SYSTEMS	9

Time- Space tradeoff - Fault tolerant scheduling algorithms - Fault tolerant ATM switches - Fault tolerant Routing and sparing Techniques - Yield and reliability enhancement for VLSI/WSI array processors. Case studies: Non-stop systems, Stratus systems, Cassini command and data sub system, IBM G5, Itanium

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the need for fault tolerance systems.
CO2: Evaluate hardware fault tolerance techniques and their reliability metrics.
CO3: Apply software redundancy and fault tolerance methods in program design.
CO4: Develop fault-tolerant algorithms and architectures for dependable systems.
CO5: Design and implement fault injection testing methodologies for software reliability.
CO6: Implement fault-tolerant algorithms for real-time applications and analyze their tradeoffs.

TEXTBOOKS:

1. E.Dubrova, ―Fault-Tolerant Design‖, Springer, 2013.
2. I. Korenand, M.Krishna, ‖Fault Tolerant Systems‖, Morgan Kaufmann, 2nd Edition, November 2020.

REFERENCES:

1. Kjetil Norvag, ―An Introduction to fault tolerant systems‖, IDI Technical report, July 2000.
2. Olga Goloubeva , Maurizio Rebaudengo , Matteo Sonza Reorda , Massimo Violante, ―Software-Implemented Hardware Fault Tolerance‖, Springer, 2006.

22CS930	ENTERPRISE CYBER SECURITY	L	T	P	C
		3		0	3

OBJECTIVES:

The Course will enable learners to:

- Learn the fundamentals of cryptography.
- Learn the key management techniques and authentication approaches.
- Explore the network and transport layer security techniques.
- Understand the application layer security standards.
- Learn the real time security practices.

UNIT I	INTRODUCTION TO CYBERSECURITY	9

Cyber Security – Need of Cybersecurity in Organizations – CIA Triad- Confidentiality, Integrity, Availability; Reason for Cyber Crime –Need for Cyber Security – History of Cyber Crime; Cybercriminals – Classification of Cybercrimes- A Global Perspective on Cyber Crimes; Cyber Laws – The Indian IT Act – Cybercrime and Punishment.

UNIT II	NETWORK SECURITY BASICS	9

Network Security Concepts- Basics of Networks- Common Types of Network Attacks- Introduction to Firewalls- Types of Firewalls- IDS/IPS- Virtual Private Networks (VPN‘s)- Secure configuration and management of network devices. Case Study: Install Kali Linux on Virtual box.

UNIT III	SECURE COMMUNICATION PROTOCOLS	9

Encryption Principles- Cryptography, Cryptanalysis, Feistel Cipher Structure. Block Encryption algorithms: DES, triple DES, and AES. Transport-Level Security: Secure Sockets Layer (SSL), Transport Layer Security TLS). Electronic Mail Security- Pretty Good Privacy (PGP), S/MIME. Securing wireless networks: WPA, WPA2, WPA3.

UNIT IV	INTRUSION DETECTION AND PREVENTION SYSTEMS	9

IDPS- Need of Intrusion Detection Systems in Cyber Security- Types of IDPS: Network-based and Host-based. Configuring and Managing IDPS for threat detection using Honeypots. Case Study: Setup a honey pot and monitor the honey pot on network.

UNIT V	WEB APPLICATION SECURITY	9

Introduction to Web Application Vulnerabilities – Cross Site Scripting (XSS) – SQL injection- Denial of Service (DoS)- Web Application Testing - Types of Penetration Tests- OWASP and OWASP Top.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understanding the core concepts and importance of cybersecurity in organizational settings.

CO2: Acquire the knowledge common network attacks and deploy appropriate security measures.

CO3: Implement encryption and secure communication protocols for data integrity and confidentiality.

CO4: Deploy and manage Intrusion Detection and Prevention Systems for threat detection.

CO5: Identify and mitigate common web application vulnerabilities.

CO6: Conduct penetration tests to evaluate the security posture of web applications.

TEXTBOOKS:

1. Anand Shinde, ―Introduction to Cyber Security Guide to the World of Cyber Security‖, Notion Press, 2021.
2. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education, 2018.

REFERENCES:

1. William Stallings, "Cryptography and Network Security - Principles and Practice", Seventh Edition, Pearson Education, 2017.
2. Ravi Das and Greg Johnson, ―Testing and Securing Web Applications‖, 2021.
3. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, O'Reilly Media, Inc, 2020.

22IT910	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework.
- Cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL.
- Explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations.
- Implement and manage Spring AOP applications using annotation-based configurations for method interception and post-execution operations.
- Build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.

UNIT I	INTRODUCTION TO REST API	6+6

RESTful APIs – overview about data exchange between client and server - separating concerns between handling HTTP requests and executing business logic - retrieving server resources via HTTP requests - injection of property values - self-contained application - serialization and deserialization – JSON properties -
managing data access.

List of Exercises/Experiments:

1. Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data

exchange between client and server.
2. Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles.
3. Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file.
4. Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control

UNIT II	ADVANCED DATA MANAGEMENT WITH JAVA AND MYSQL	6+6

Build production-grade applications – MYSQL - mapping Java classes to relational database - repository interface - data access operations – retrieving data from the database –mapping of request body to entity - retrieve an entity - capture data from API requests - building complex queries using keywords.

List of Exercises/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.

UNIT III	ADVANCED JPA QUERIES AND ANNOTATIONS	6+6

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercises/Experiments:

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations.
3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns.
4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.

UNIT IV	JPA ASSOCIATIONS AND MAPPING	6+6

JPA Mapping of One-to-One Associations - fetching entities using queries – Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA.

List of Exercises/Experiments:

1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities.
2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities.
3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to retrieve employee details efficiently

UNIT V	SPRING BOOT ESSENTIALS: API SECURITY, LOGGING, AOP, AND	6+6

	BUILD MANAGEMENT	

SwaggerUI with Spring Boot, OpenUI with Spring Boot, Logging with Spring Boot, Changing Log Level, Logging Request and Response- Managing Spring Boot Logging Configuration - Aspect-Oriented Programming (AOP) Concepts – Method Parameter Handling - Post- Execution Operations - Returning Data Handling - Comprehensive Advice Handling. API security using JWT, Gradle for build management, Sonar Lint for coding standards and guidelines.

List of Exercises/Experiments:

1. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction.
2. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively.
3. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Create simple applications using RESTful APIs and effectively manage HTTP methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6: Apply security measures to REST APIs using Spring Security and JWT to protect sensitive data and ensure secure communication between clients and servers.

TEXTBOOKS:

3. Raja CSP Raman, Ludovic Dewailly, ―Building RESTful Web Services with Spring 5‖, Packt Publishing, 2018.
4. Leonard Richardson, Sam Ruby ―RESTful Web Services‖ O'Reilly Media, 2008.
5. Ludovic Dewailly, ―Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework‖, Packt Publishing, 2015
6. Raja CSP Raman, Ludovic Dewailly, ―Building RESTful Web Services with Spring 5 – Second Edition
7. Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0‖, Packt Publishing, 2018

REFERENCES:

4. Ranga Karanam, ―Master Java Web Services and REST API with Spring Boot‖, Packt Publishing, 2018.
5. Balaji Varanasi, Sudha Belida, ―Spring REST‖, Apress, 2015.
6. Greg L. Turnquist, ―Learning Spring Boot 2.0‖, Packt Publishing, 2021
7. Sourabh Sharma, ―Modern API Development with Spring and Spring Boot‖, Packt Publishing, 2021

LIST OF EQUIPMENTS/SOFTWARE:

Java Persistence API, Spring Boot

CLOUD COMPUTING

22CS907	CLOUD FOUNDATIONS (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

- ✓ To describe the different ways a user can interact with Cloud.
- ✓ To discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.
- ✓ To confer the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.
- ✓ To demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.
- ✓ To determine a variety of managed big data services in the cloud.

UNIT I	INTRODUCTION TO CLOUD	6+6

Cloud Computing - Cloud Versus Traditional Architecture - IaaS, PaaS, and SaaS - Cloud Architecture - The GCP Console - Understanding projects - Billing in GCP - Install and configure Cloud SDK - Use Cloud Shell - APIs - Cloud Console Mobile App.

List of Exercise/Experiments:

1. Install and configure cloud SDK.
2. Connect to computing resources hosted on Cloud via Cloud Shell.

UNIT II	COMPUTE AND STORAGE	6+6

Compute options in the cloud - Exploring IaaS with Compute Engine - Configuring elastic apps with autoscaling - Exploring PaaS - Event driven programs - Containerizing and orchestrating apps - Storage options in the cloud - Structured and unstructured storage in the cloud - Unstructured storage using Cloud Storage - SQL managed services - NoSQL managed services.

List of Exercise/Experiments:

1. Create virtual machine instances of various machine types using the Cloud Console and the command line. Connect an NGINX web server to your virtual machine.
2. Create a small App Engine application that displays a short message.
3. Create, deploy, and test a cloud function using the Cloud Shell command line.
4. Deploy a containerized application.
5. Create a storage bucket, upload objects to it, create folders and subfolders in it, and make objects publicly accessible using the Cloud command line.

UNIT III	APIs AND SECURITY IN THE CLOUD	6+6

The purpose of APIs – API Services - Managed message services - Introduction to security in the cloud - The shared security model - Encryption options - Authentication and authorization with Cloud IAM - Identify Best Practices for Authorization using Cloud IAM.

List of Exercise/Experiments:

1. Deploy a sample API with any of the API service.
2. Publish messages with managed message service using the Python client library.
3. Create two users. Assign a role to a second user and remove assigned roles associated with Cloud IAM. Explore how granting and revoking permissions works from Cloud Project Owner and Viewer roles.

UNIT IV	NETWORKING, AUTOMATION AND MANGAEMENT TOOLS	6+6

Introduction to networking in the cloud - Defining a Virtual Private Cloud - Public and private IP address basics - Cloud network architecture - Routes and firewall rules in the cloud - Multiple VPC networks - Building hybrid clouds using VPNs - Different options for load balancing - Introduction to Infrastructure as Code - Terraform - Monitoring and management tools.

List of Exercise/Experiments:

1. Create several VPC networks and VM instances and test connectivity across networks.
2. Create two nginx web servers and control external HTTP access to the web servers using tagged firewall rules.
3. Configure a HTTP Load Balancer with global backends. Stress test the Load Balancer and denylist the stress test IP.
4. Create two managed instance groups in the same region. Then, configure and test an Internal Load Balancer with the instances groups as the backends.
5. Monitor a Compute Engine virtual machine (VM) instance with Cloud Monitoring by creating uptime check, alerting policy, dashboard and chart.

UNIT V	BIG DATA AND MACHINE LEARNING SERVICES	6+6

Introduction to big data managed services in the cloud - Leverage big data operations - Build Extract, Transform, and Load pipelines - Enterprise Data Warehouse Services - Introduction to machine learning in the cloud - Building bespoke machine learning models with AI Platform - Pre-trained machine learning APIs.

List of Exercise/Experiments:

1. Create a cluster, run a simple Apache Spark job in the cluster, then modify the number of workers in the cluster.
2. Create a streaming pipeline using one of the cloud service.
3. Set up your Python development environment, get the relevant SDK for Python, and run an example pipeline using the Cloud Console.
4. Use cloud-based data preparation tool to manipulate a dataset. Import datasets, correct mismatched data, transform data, and join data.
5. Utyilize a cloud-based data processing and analysis tool for data exploration and use a machine learning platform to train and deploy a custom TensorFlow Regressor model for predicting customer lifetime value.

TOTAL: 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Describe the different ways a user can interact with Cloud.

CO2: Discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.

CO3: Discuss the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.

CO4: Demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.

CO5: Discover a variety of managed big data services in the cloud.

CO6: Use Cloud services to build applications.

REFERENCES:

1. https://cloud.google.com/docs
2. https://www.cloudskillsboost.google/paths/36
3. https://nptel.ac.in/courses/106105223
4. Anthony J. Sequeira, "AWS Certified Cloud Practitioner (CLF-C01) Cert Guide", First Edition, Pearson Education, 2020.
5. AWS Documentation (amazon.com)
6. AWS Skill Builder
7. AWS Academy Cloud Foundations Course - https://www.awsacademy.com/vforcesite/LMS_Login

LIST OF EQUIPMENTS:

GCP / CloudSkillBoost Platform/AWS Console /AWS Academy Learner Lab.

22CS908	CLOUD ARCHITECTING (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners:

- To make architectural decisions based on AWS architectural principles and best practices.
- To describe the features and benefits of Amazon EC2 instances, and compare and contrast managed and unmanaged database services.
- To create a secure and scalable AWS network environment with VPC, and configure IAM for improved security and efficiency.
- To use AWS services to make infrastructure scalable, reliable, and highly available.
- To use AWS managed services to enable greater flexibility and resiliency in an infrastructure.

UNIT I	INTRODUCING CLOUD ARCHITECTING AND STORAGE LAYER	6+6

Cloud architecting - The AWS Well-Architected Framework - AWS global infrastructure - Amazon S3 - Amazon S3 Versioning - Storing data in Amazon S3 - Moving data to and from Amazon S3 - Amazon S3 Transfer Acceleration - Choosing Regions for your

architecture.

List of Exercise/Experiments:

1. Creating a Static Website for the Café.
2. Configure an S3 bucket to automatically encrypt all uploaded objects.
3. Set up a cross-region replication configuration for an S3 bucket.

UNIT II	COMPUTE LAYER AND DATABASE LAYER	6+6

Adding compute with Amazon EC2 - Choosing an Amazon Machine Image (AMI) to launch an Amazon EC2 instance - Selecting an Amazon EC2 instance type - Using user data to configure an EC2 instance - Adding storage to an Amazon EC2 instance - Amazon EC2 pricing options - Amazon EC2 considerations - Database layer considerations - Amazon Relational Database Service (Amazon RDS) - Amazon DynamoDB - Database security controls - Migrating data into AWS databases.

List of Exercise/Experiments:

1. Creating a Dynamic Website for the Café.
2. Creating an Amazon RDS database.
3. Migrating a Database to Amazon RDS.
4. Create a web application that stores data in a managed database using EC2 instances and Amazon RDS.

UNIT III	CREATING AND CONNECTING NETWORKS	6+6

Creating an AWS networking environment - Connecting your AWS networking environment to the internet - Securing your AWS networking environment - Connecting your remote network with AWS Site-to-Site VPN - Connecting your remote network with AWS Direct Connect - Connecting virtual private clouds (VPCs) in AWS with VPC peering - Scaling your VPC network with AWS Transit Gateway - AWS Transit Gateway - Connecting your VPC to supported AWS services. Securing User and Application Access: Account users and AWS Identity and Access Management (IAM) - Organizing users - Federating users - Multiple accounts.

List of Exercise/Experiments:

1. Creating a Virtual Private Cloud.
2. Creating a VPC Networking Environment for the Café.
3. Creating a VPC Peering Connection.
4. Configure a VPC with subnets, an internet gateway, route tables, and a security group, and connect an on-premises network to the VPC.

UNIT IV	RESILIENT CLOUD ARCHITECTURE	6+6

Scaling your compute resources - Scaling your databases - Designing an environment that‘s highly available – Monitoring - Reasons to automate - Automating your infrastructure - Automating deployments - AWS Elastic Beanstalk - Overview of caching - Edge caching - Caching web sessions - Caching databases.

List of Exercise/Experiments:

1. Controlling Account Access by Using IAM.
2. Creating Scaling Policies for Amazon EC2 Auto Scaling.

3. Creating a Highly Available Web Application.
4. Creating a Scalable and Highly Available Environment for the Café.
5. Streaming Dynamic Content Using Amazon CloudFront.

UNIT V	BUILDING DECOUPLED ARCHITECTURES, MICROSERVICES AND SERVERLESS ARCHITECTURE	6+6

Decoupling your architecture - Decoupling with Amazon Simple Queue Service (Amazon SQS) - Decoupling with Amazon Simple Notification Service (Amazon SNS) - Sending messages between cloud applications and on-premises with Amazon MQ. Introducing microservices - Building microservice applications with AWS container services - Introducing serverless architectures - Building serverless architectures with AWS Lambda - Extending serverless architectures with Amazon API Gateway - Orchestrating microservices with AWS Step Functions - Disaster planning strategies - Disaster recover patterns.

List of Exercise/Experiments:

1. Breaking a Monolithic Node.js Application into Microservices.
2. Implementing a Serverless Architecture on AWS.
3. Implementing a Serverless Architecture for the Café.
4. Creating an AWS Lambda Function and explore using AWS Lambda with Amazon S3.

TOTAL: 60 PERIODS

OUTCOMES:

Upon Completion of this course, the students will be able to:

CO1: Make architectural decisions based on AWS architectural principles and best practices

CO2: Describe the features and benefits of Amazon EC2 instances, and compare and contrast managed and unmanaged database services.

CO3: Create a secure and scalable AWS network environment with VPC, and configure IAM for improved security and efficiency.

CO4: Use AWS services to make infrastructure scalable, reliable, and highly available

CO5: Use AWS managed services to enable greater flexibility and resiliency in an infrastructure

CO6: Develop disaster recovery strategies for AWS environments.

REFERENCES:

1. AWS Certified Solutions Architect Official Study Guide by Joe Baron, Hisham Baz, Tim Bixler
2. Architecting the Cloud by Michael Kavis.
3. AWS Documentation (amazon.com)
4. AWS Skill Builder
5. AWS Academy Cloud Architecting Course - https://www.awsacademy.com/vforcesite/LMS_Login

LIST OF EQUIPMENTS:

AWS Console Account/AWS Academy Learner Lab.

22CS909	VIRTUALIZATION	L	T		C
		3	0		3

OBJECTIVES:

The Course will enable learners to:

- ✓ Explain the fundamental concepts of virtualization
- ✓ Analyze the role of hypervisors in hardware virtualization
- ✓ Apply the understanding of CPU, memory (MMU), and I/O virtualization techniques
- ✓ Assess security considerations of virtualized environments
- ✓ Discuss strategies for protecting VMs and data centers

UNIT I	INTRODUCTION	9

Virtualization - Virtual Machines - Hypervisors - Type-1 and Type-2 Hypervisors - Multiplexing and Emulation - Approaches to Virtualization and Paravirtualization - Benefits of Using Virtual Machines. Working with Virtual Machines.

UNIT II	HARDWARE VIRTUALIZATION	9

The Popek/Goldberg Theorem - Virtualization without Architectural Support: Full Virtualization - Paravirtualization - Designs Options for Type-1 Hypervisors. Hypervisors: Describing a Hypervisor - Role of Hypervisor - VMWare ESX - Citrix Hypervisor - Microsoft Hyper-V.

UNIT III	TYPES OF VIRTUALIZATIONS	10

CPU Virtualization with VT-x: Design requirements - The VT-x Architecture - KVM. MMU Virtualization: Extended Paging - Virtualizing Memory in KVM. I/O Virtualization: Benefits of I/O Interposition - Physical I/O - Virtual I/O Without Hardware Support- Virtual I/O with Hardware Support. Virtualization Support in ARM Processors.

UNIT IV	VIRTUALIZATION SECURITY	9

Fundamentals of Virtualization Security: Virtualization Architecture - Threats to a Virtualized Environment. Securing Hypervisors: Hypervisor Configuration and Security. Designing Virtual Networks for Security: Comparing Virtual and Physical Networks - Virtual Network Security Considerations - Configuring Virtual Switches for Security.

UNIT V	VIRTUALIZATION AND AVAILABILITY	8

Availability - Protecting a Virtual Machine - Protecting Multiple Virtual Machines - Protecting Datacenters - Deploying Applications in a Virtual Environment - Recent Trends in Virtualization.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of virtualization and its benefits.

CO2: Assess the significance of hypervisors in hardware virtualization, examining their roles and implications for system efficiency and performance

CO3: Utilize knowledge of virtualization technologies to solve practical problems and implement effective solutions

CO4: Analyze security threats and design secure virtual networks

CO5: Discuss strategies to improve availability in virtual environment and for protecting VMs

and data centers

CO6: Use virtualization technology effectively to optimize system performance and resource usage in real-world settings

TEXTBOOKS:

1. Edouard Bugnion, Jason Nieh, Dan Tsafrir, ¯Hardware and Software Support for Virtualization‖, Morgan & Claypool Publishers, 2017.
2. Matthew Portnoy, ¯Virtualization Essentials‖, Third Edition, Sybex - John Wiley & Sons, 2023.

REFERENCES:

1. Dave Shackleford, ¯Virtualization Security: Protecting Virtualized Environments‖, Sybex - John Wiley & Sons, 2012.
2. Nelson Ruest, Danielle Ruest, Virtualization, A beginners guide, 2009, McGrawHill.
3. Nadeau,Tim Cerng, Je Buller, Chuck Enstall, Richard Ruiz, Mastering Microsoft Virtualization, Wiley Publication, 2010.
4. William Von Hagen, Professional Xen Virtualization, Wiley Publication, 2008.

22CS910	DEVOPS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Bridge the gap between development and operations for faster, more reliable software releases.
- Automate software delivery with CI/CD pipelines.
- Package and deploy apps efficiently using Docker containers.
- Automate infrastructure with Infrastructure as Code (IaC).
- Monitor and troubleshoot applications in production.

UNIT I	INTRODUCTION TO DEVOPS	9

Software Development Methodologies - Operations Methodologies - Systems Methodologies - Development, Release, and Deployment Concepts - Infrastructure Concepts. What is DevOps? - DevOps importance and benefits -DevOps principles and practices - 7 C's of DevOps lifecycle for business agility - DevOps and continuous testing. How to choose right DevOps tools? - Challenges with DevOps implementation.

UNIT II	VERSION CONTROL WITH GIT	9

Introduction to Git version control system - Git commands for basic operations (clone, commit, push, pull) - Branching and merging strategies - Collaboration using Git workflows.

UNIT III	CONTINUOUS INTEGRATION AND DELIVERY (CI/CD)	9

Introduction to CI/CD pipelines - Benefits of CI/CD for faster deployments - Setting up a CI/CD pipeline with Jenkins - Automating builds, tests, and deployments.

UNIT IV	CONTAINERIZATION WITH DOCKER	9

Introduction to containerization and its benefits - Understanding Docker concepts: images, containers, registries - Building and managing Docker containers - Docker Compose for multi-container applications - Introduction to container orchestration with Docker Swarm or Kubernetes.

UNIT V	INFRASTRUCTURE AS CODE (IAC) AND MONITORING	9

Introduction to Infrastructure as Code (IaC) - Benefits of using IaC for repeatable infrastructure provisioning - Learning IaC with Terraform - Setting up infrastructure configurations with Terraform - Introduction to monitoring and logging tools for applications - Alerting and troubleshooting techniques.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the core principles and philosophies of DevOps.
CO2: Implement version control systems for code management and collaboration.
CO3: Automate software delivery pipelines using CI/CD tools.
CO4: Utilize containerization technologies for packaging and deploying applications.
CO5: Configure infrastructure as code (IaC) for repeatable deployments.
CO6: Monitor and maintain applications in a production environment.

TEXT BOOKS:

1. Deepak Gaikwad, Viral Thakkar, "DevOps Tools: from Practitioner's Point of View", Wiley, 2019.
2. Jennifer Davis, Ryn Daniels, "Effective DevOps", O'Reilly Media, 2016.

REFERENCES:

1. Gene Kim, Jez Humble, Patrick Debois, "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", IT Revolution Press, 2016.
2. Jez Humble, Gene Kim, "Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation", Addison-Wesley, 2010.
3. Yevgeniy Brikman, "Terraform: Up & Running: Writing Infrastructure as Code", O'Reilly Media, 2019.
4. Joseph Muli, "Beginning DevOps with Docker", Packt Publishing, 2018.

22CS911	DATA ENGINEERING IN CLOUD	L	T	P	C
		3	0		3

OBJECTIVES:

The Course will enable learners to:

- ✓ Grasp the fundamentals of data engineering, emphasizing cloud-based data access.
- ✓ Construct robust and secure data pipelines using Cloud services.
- ✓ Select and implement appropriate data storage solutions while prioritizing pipeline security.
- ✓ Utilize cloud tools for handling extensive data for machine learning purposes.
- ✓ Efficiently analyze, visualize, and automate data pipelines to streamline operations.

UNIT I	INTRODUCTION	8

Introduction to data Engineering - The Data Engineering Life Cycle - Data Engineering and Data Science - Data-Driven Organizations: Data-driven decisions - The data pipeline - The role of the data engineer in data-driven organizations - Modern data strategies - The Elements of Data: The five Vs of data – volume, velocity, variety, veracity, and value. Demo: Accessing and Analyzing Data by Using Amazon S3.

UNIT II	SECURE AND SCALABLE DATA PIPELINES	10

The evolution of data architectures - Modern data architecture on AWS - Modern data architecture pipeline: Ingestion and storage - Processing and consumption - Streaming

analytics pipeline - Security of analytics workloads - Scaling - Creating a scalable infrastructure and components. ETL and ELT comparison - Data wrangling.		
UNIT III	**STORING AND ORGANIZING DATA**	**9**
Comparing batch and stream ingestion - Batch ingestion processing - Purpose-built ingestion tools - AWS Glue for batch ingestion processing - Kinesis for stream processing - Scaling considerations for batch processing and stream processing - Storage in the modern data architecture - Data lake storage - Data warehouse storage - Purpose-built databases - Storage in support of the pipeline - Securing storage.		
UNIT IV	**PROCESSING BIG DATA AND DATA FOR ML**	**10**
Big data processing concepts - Apache Hadoop - Apache Spark - Amazon EMR - Managing your Amazon EMR clusters - Apache Hudi - The ML lifecycle - Collecting data - Applying labels to training data with known targets - Preprocessing data – Feature engineering - Developing a model - Deploying a model - ML infrastructure on AWS - SageMaker - Amazon CodeWhisperer - AI/ML services on AWS.		
UNIT V	**DATA ANALYSIS AND VISUALIZATION**	**8**
Analyzing and Visualizing Data: Considering factors that influence tool selection - Comparing AWS tools and services - Selecting tools for a gaming analytics use case. Automating the Pipeline: Automating infrastructure deployment - CI/CD - Automating with Step Functions		
TOTAL: 45 PERIODS		

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand data engineering, pipelines & access data in the cloud.

CO2: Build secure & scalable data pipelines using AWS services.

CO3: Choose the right data storage & secure your data pipelines.

CO4: Process big data for machine learning with cloud tools.

CO5: Analyze & visualize data and automate data pipelines.

CO6: Apply best practices in data governance, compliance, and ethics throughout the data engineering process, ensuring responsible handling and usage of data.

TEXT BOOKS:

1. Martin Kleppman, ¯Data Engineering: Building Reliable Scalable Data Systems‖, O'Reilly Media, 2017.
2. Wes McKinney, ¯Python for Data Analysis‖, 2nd Edition, O'Reilly Media, 2017.

REFERENCES:

5. Martin Kleppman, ¯Designing Data-Intensive Applications‖, O'Reilly Media, 2017.
6. AWS Documentation (amazon.com)
7. AWS Skill Builder
8. AWS Academy Data Engineering Course - https://www.awsacademy.com/vforcesite/LMS_Login

22CS912	CLOUD SECURITY FOUNDATIONS	L		P	C
		3		0	3

OBJECTIVES:

The Course will enable learners to:

- ✓ Learn the basics of cloud security, including the shared responsibility model and identity management.
- ✓ Set up a secure cloud infrastructure with features like virtual private clouds and security groups.
- ✓ Safeguard application data at rest and in transit using encryption and Amazon S3 protection features.
- ✓ Learn to capture and analyze log data using AWS services like CloudTrail and CloudWatch.
- ✓ Develop skills for identifying and managing security incidents in the cloud, adhering to best practices.

UNIT I	SECURITY IN CLOUD	9

Introduction to Security, Security in the Cloud, Security design principles, Shared responsibility model, Activity: Shared Responsibility Model, Identity and Access Management (IAM) fundamentals, Authenticating and Authorizing with IAM, Examples of authorizing with IAM, Additional authentication and access management services, Using Organizations.

UNIT II	SECURING INFRASTRUCTURE	9

Structure of a three-tier web application, virtual private cloud (VPC), Setting up public and private subnets and internet protocols, Security groups, Network access control lists (ACLs), Load balancers, Protecting compute resources.

UNIT III	PROTECTING APPLICATION DATA	9

Basics on Data Protection, Protect data at rest, Amazon S3 protection features, Protection through encryption, Protect data in transit, protect data in Amazon S3, additional data protection services.

UNIT IV	LOGGING AND MONITORING	9

Importance of logging and monitoring, Capture and collect, Reading a Log File, AWS services with built-in logs, Monitor and report, CloudTrail and Amazon CloudWatch, methods for logging and monitoring, additional AWS services for logging and monitoring, AWS Security Hub.

UNIT V	RESPONDING AND MANAGING AN INCIDENT	9

Identifying an incident, Services that support the discovery and recognition phase, AWS Config and AWS Lambda, Services that support the resolution and recovery phase, Best practices for handling an incident.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Learn essential cloud security principles and identity management fundamentals.

CO2: Design and implement secure cloud infrastructure components for effective resource protection.

CO3: Explore methods for securing application data stored in the cloud, including encryption techniques and data protection features.

CO4: Gain proficiency in implementing logging and monitoring practices to detect and respond to security events efficiently in cloud environments.

CO5: Acquire skills to identify and manage security incidents in the cloud, utilizing appropriate tools and techniques for incident detection, analysis, and resolution.

CO6: Proficiently ensure the protection of cloud environments by applying strong security measures across all aspects, ensuring resilience and compliance.

TEXT BOOKS:

3. Tim Mather, Subra Kumaraswamy, Shahed Latif, ¯Cloud Security and Privacy, An Enterprise Perspective on Risks and Compliance‖, Oreilly Media 2009.
4. Vic (J.R.) Winkler, ¯Securing the Cloud, Cloud Computer Security Techniques and Tactics‖, Syngress, April 2011.

REFERENCES:

13. Rajkumar Buyya, James Broberg, Andrzej, ¯Cloud Computing: Principles and Paradigms‖, Wiley India Publications 2011.
14. Arshdeep Bahga and Vijay Madisetti, ¯Cloud Computing –A Hands on Approach‖, Universities Press (India) Pvt Ltd. 2014.

22CS931	SOFTWARE DEFINED NETWORKS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- ✓ Understand the need for SDN and its data plane operations
- ✓ Understand the functions of control plane
- ✓ Comprehend the migration of networking functions to SDN environment
- ✓ Explore various techniques of network function virtualization
- ✓ Summarize the concepts behind network virtualization

UNIT I	SDN: INTRODUCTION	9

Evolving Network Requirements – The SDN Approach – SDN architecture - SDN Data Plane, Control plane and Application Plane.

UNIT II	SDN DATA PLANE AND CONTROL PLANE	9

Data Plane functions and protocols - OpenFLow Protocol - Flow Table - Control Plane Functions - Southbound Interface, Northbound Interface – SDN Controllers - Ryu, OpenDaylight, ONOS - Distributed Controllers.

UNIT III	SDN APPLICATIONS	9

SDN Application Plane Architecture – Network Services Abstraction Layer – Traffic Engineering – Measurement and Monitoring – Security – Data Center Networking.

UNIT IV	NETWORK FUNCTION VIRTUALIZATION	9

Network Virtualization - Virtual LANs – OpenFlow VLAN Support - NFV Concepts – Benefits and Requirements – Reference Architecture.

UNIT V	NFV FUNCTIONALITY	9

NFV Infrastructure – Virtualized Network Functions – NFV Management and Orchestration – NFV Use cases – SDN and NFV. Case Study: Implement SDN for Network optimization.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Describe the motivation behind SDN
CO2: Identify the functions of the data plane and control plane
CO3: Design and develop network applications using SDN
CO4: Orchestrate network services using NFV
CO5: Explain various use cases of SDN and NFV
CO6: Integrate SDN and NFV principles to effectively tackle diverse challenges in network virtualization.

TEXTBOOKS:

1. William Stallings, ¯Foundations of Modern Networking: SDN, NFV, QoE, IoT and Cloud‖, Pearson Education, 1st Edition, 2022.

REFERENCES:

1. Ken Gray, Thomas D. Nadeau, ¯Network Function Virtualization‖, Morgan Kauffman, 2016.
2. Thomas D Nadeau, Ken Gray, ¯SDN: Software Defined Networks‖, O'Reilly Media, 2013.
3. Fei Hu, ¯Network Innovation through OpenFlow and SDN: Principles and Design‖, 1st Edition, CRC Press, 2014.
4. Paul Goransson, Chuck Black Timothy Culver, ¯Software Defined Networks: A Comprehensive Approach‖, 2nd Edition, Morgan Kaufmann Press, 2016.
5. Oswald Coker, Siamak Azodolmolky, ¯Software-Defined Networking with OpenFlow‖, 2nd Edition, O'Reilly Media, 2017.

22CS932	STORAGE TECHNOLOGIES	L	T	P	C
		3	0		3

OBJECTIVES:

The Course will enable learners to:

- ✓ Characterize the functionalities of logical and physical components of storage
- ✓ Describe various storage networking technologies
- ✓ Identify different storage virtualization technologies
- ✓ Discuss the different backup and recovery strategies
- ✓ Understand common storage management activities and solutions

UNIT I	STORAGE SYSTEMS	9

Introduction to Information Storage: Digital data and its types, Information storage, Key characteristics of data center and Evolution of computing platforms. Information Lifecycle Management. Third Platform Technologies: Cloud computing and its essential characteristics, Cloud services and cloud deployment models, Big data analytics, Social networking and mobile computing, Characteristics of third platform infrastructure and Imperatives for third platform transformation. Data Center Environment: Building blocks of a data center, Compute systems and compute virtualization and Software-defined data center.

UNIT II	INTELLIGENT STORAGE SYSTEMS AND RAID	8

Components of an intelligent storage system, Components, addressing, and performance of hard disk drives and solid-state drives, RAID, Types of intelligent storage systems, Scale-up and scale-out storage Architecture. Block-Based Storage System, File-Based Storage

System, Object-Based and Unified Storage.		
UNIT III	**STORAGE NETWORKING TECHNOLOGIES AND VIRTUALIZATION**	**10**
Fibre Channel SAN: Software-defined networking, FC SAN components and architecture, FC SAN topologies, link aggregation, and zoning, Virtualization in FC SAN environment. Internet Protocol SAN: iSCSI protocol, network components, and connectivity, Link aggregation, switch aggregation, and VLAN, FCIP protocol, connectivity, and configuration. Fibre Channel over Ethernet SAN: Components of FCoE SAN, FCoE SAN connectivity, Converged Enhanced Ethernet, FCoE architecture.		
UNIT IV	**BACKUP, ARCHIVE AND REPLICATION**	**10**
Introduction to Business Continuity, Backup architecture, Backup targets and methods, Data deduplication, Cloud-based and mobile device backup, Data archive, Uses of replication and its characteristics, Compute based, storage-based, and network-based replication, Data migration, Disaster Recovery as a Service (DRaaS)**.**		
UNIT V	**SECURING STORAGE INFRASTRUCTURE**	**8**
Information security goals, Storage security domains, Threats to a storage infrastructure, Security controls to protect a storage infrastructure, Governance, risk, and compliance, Storage infrastructure management functions, Storage infrastructure management processes.		
TOTAL: 45 PERIODS		

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Demonstrate the fundamentals of information storage management and various models of Cloud infrastructure services and deployment

CO2: Illustrate the usage of advanced intelligent storage systems and RAID

CO3: Interpret various storage networking architectures - SAN, including storage subsystems and virtualization

CO4: Examine the different role in providing disaster recovery and remote replication technologies

CO5: Infer the security needs and security measures to be employed in information storage management

CO6: Synthesize and apply comprehensive strategies for optimizing information storage infrastructure resilience and security.

TEXTBOOKS:

1. EMC Corporation, Information Storage and Management, Wiley, India.
2. Jon Tate, Pall Beck, Hector Hugo Ibarra, Shanmuganathan Kumaravel and Libor Miklas, Introduction to Storage Area Networks, Ninth Edition, IBM - Redbooks, December 2017.

REFERENCES:

1. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein, Storage Networks Explained, Second Edition, Wiley, 2009.

22CS933	MACHINE LEARNING FOR NLP IN CLOUD	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Illustrate how to apply the ML pipeline to NLP.
- Implement text extraction to obtain data from web pages.
- Build a solution that uses AWS services to transcribe and translate text from multimedia.
- Build a solution using a combination of algorithms and Amazon Machine Learning (Amazon ML) services.
- Identify use cases to use generative AI and LLMs.
- Use LLMs with AWS generative AI services.

UNIT I	Introduction to NLP	8

NLP – Business Problems Solved by NLP – NLP Roles - NLP and ML – Common NLP tasks – Apply ML to NLP problem - Evolution of NLP architectures.

UNIT II	Processing Text for NLP	10

Text processing overview - Getting text - Extracting Text from Webpages and Images - Text preprocessing - Vectorizing text - Encoding and Vectorizing Text - Advanced processing - Storing and visualizing unstructured data – Implement Sentiment Analysis - Identifying the steps for text processing - Examining the algorithms for sentiment analysis.

UNIT III	Information Extraction	9

Information extraction overview - Types of information extraction - Implementing information extraction – Working with Entities - Topic Modeling - Identifying the approach - Implementing Topic Modeling with Amazon Comprehend, Neural Topic Model (NTM).

UNIT IV	Translating Languages	9

Working with language issues - Detecting and translating languages - Transcribing and vocalizing text with AWS services - Implementing a Multilingual Solution.

UNIT V	Generative AI	9

Generative AI - Amazon Bedrock Overview - Introducing foundations models and large language models - Transformer architecture - LLMs configuration parameters - Introducing prompt engineering - Use LLMs to Perform NLP Tasks - Adapting LLMs - Application Integration.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Apply the ML pipeline to NLP.
CO2: Implement text extraction to obtain data from webpages.
CO3: Build a solution that uses AWS services to transcribe and translate text from multimedia.
CO4: Build a solution using a combination of algorithms and Amazon Machine Learning (Amazon ML) services.
CO5: Identify use cases to use generative AI and LLMs.
CO6: Use LLMs with AWS generative AI services.

TEXT BOOKS:

1. Mona M, Premkumar Rangarajan, Natural Language Processing with AWS AI Services, Packt Publications, 2021

REFERENCES:

1. Saket S R Mengle, Maximo Gurmendez, Mastering Machine Learning on AWS: Advanced machine learning in Python using SageMaker, Apache Spark, and TensorFlow, Packt Publications, 2019.
2. AWS Documentation (amazon.com)

3. AWS Skill Builder
4. AWS Academy Machine Learning for Natural Language Processing Course - https://www.awsacademy.com/vforcesite/LMS_Login

22CS934	CLOUD SERVICES MANAGEMENT	L	T		C
		3	0		3

OBJECTIVES:

The Course will enable learners to:

- ✓ Introduce Cloud Service Management terminology, definition & concepts
- ✓ Compare and contrast cloud service management with traditional IT service management
- ✓ Identify strategies to reduce risk and eliminate issues associated with adoption of cloud services
- ✓ Select appropriate structures for designing, deploying and running cloud-based services in a business environment
- ✓ Illustrate the benefits and drive the adoption of cloud-based services to solve real world problems

UNIT I	CLOUD SERVICE MANAGEMENT FUNDAMENTALS	9

Cloud Ecosystem, The Essential Characteristics, Basics of Information Technology Service Management and Cloud Service Management, Service Perspectives, Cloud Service Models, Cloud Service Deployment Models.

UNIT II	CLOUD SERVICES STRATEGY	9

Cloud Strategy Fundamentals, Cloud Strategy Management Framework, Cloud Policy, Key Driver for Adoption, Risk Management, IT Capacity and Utilization, Demand and Capacity matching, Demand Queueing, Change Management, Cloud Service Architecture.

UNIT III	CLOUD SERVICE MANAGEMENT	9

Cloud Service Reference Model, Cloud Service LifeCycle, Basics of Cloud Service Design, Dealing with Legacy Systems and Services, Benchmarking of Cloud Services, Cloud Service Capacity Planning, Cloud Service Deployment and Migration, Cloud Marketplace, Cloud Service Operations Management.

UNIT IV	CLOUD SERVICE ECONOMICS	9

Pricing models for Cloud Services, Freemium, Pay Per Reservation, Pay per User, Subscription based Charging, Procurement of Cloud-based Services, Capex vs Opex Shift, Cloud service Charging, Cloud Cost Models.

UNIT V	CLOUD SERVICE GOVERNANCE & VALUE	9

IT Governance Definition, Cloud Governance Definition, Cloud Governance Framework, Cloud Governance Structure, Cloud Governance Considerations, Cloud Service Model Risk Matrix, Understanding Value of Cloud Services, Measuring the value of Cloud Services, Balanced Scorecard, Total Cost of Ownership.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Exhibit cloud-design skills to build and automate business solutions using cloud technologies.
CO2: Possess Strong theoretical foundation leading to excellence and excitement towards adoption of cloud-based services
CO3: Solve the real world problems using Cloud services and technologies
CO4: Develop and deploy services on the cloud and set up a cloud environment
CO5: Explain security challenges in the cloud environment
CO6: Demonstrate proficiency in integrating cloud technologies and services to address diverse business challenges effectively.

TEXT BOOKS:

1. Enamul Haque, "Cloud Service Management and Governance: Smart Service Management in Cloud Era", Enel Publications, 2023.
2. Thomas Erl, Ricardo Puttini, Zaigham Mohammad, "Cloud Computing: Concepts, Technology & Architecture", Prentice Hall, 2013.

REFERENCES:

1. Thomas Erl, Robert Cope, Amin Naserpour, "Cloud Computing Design Patterns", Prentice Hall, 2015.
2. Praveen Ayyappa, "Economics of Cloud Computing", LAP Lambert Academic Publishing, 2020.
2. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "Mastering Cloud Computing Foundations and Applications Programming", Elsevier, 2013.

22IT910	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:
The Course will enable learners to:

- Provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework.
- Cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL.
- Explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations.
- Implement and manage Spring AOP applications using annotation-based configurations for method interception and post-execution operations.
- Build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.

UNIT I	INTRODUCTION TO REST API	6+6

RESTful APIs – overview about data exchange between client and server - separating concerns between handling HTTP requests and executing business logic - retrieving server resources via HTTP requests - injection of property values - self-contained application - serialization and deserialization – JSON properties -
managing data access.

List of Exercises/Experiments:

1. Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data exchange between client and server.
2. Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles.
3. Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file.
4. Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control

UNIT II	ADVANCED DATA MANAGEMENT WITH JAVA AND MYSQL	6+6

Build production-grade applications – MYSQL - mapping Java classes to relational database - repository interface - data access operations – retrieving data from the database –mapping of request body to entity - retrieve an entity - capture data from API requests - building complex queries using keywords.

List of Exercises/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.

UNIT III	ADVANCED JPA QUERIES AND ANNOTATIONS	6+6

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercises/Experiments:

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations.
3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns.
4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.

UNIT IV	JPA ASSOCIATIONS AND MAPPING	6+6

JPA Mapping of One-to-One Associations - fetching entities using queries – Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA.

List of Exercises/Experiments:

1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities.
2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities.
3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to retrieve employee details efficiently

UNIT V	SPRING BOOT ESSENTIALS: API SECURITY, LOGGING, AOP, AND BUILD MANAGEMENT	6+6

SwaggerUI with Spring Boot, OpenUI with Spring Boot, Logging with Spring Boot, Changing Log Level, Logging Request and Response- Managing Spring Boot Logging Configuration - Aspect-Oriented Programming (AOP) Concepts – Method Parameter Handling - Post- Execution Operations - Returning Data Handling - Comprehensive Advice Handling. API security using JWT, Gradle for build management, Sonar Lint for coding standards and guidelines.

List of Exercises/Experiments:

1. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction.
2. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively.
3. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Create simple applications using RESTful APIs and effectively manage HTTP methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6: Apply security measures to REST APIs using Spring Security and JWT to protect sensitive data and ensure secure communication between clients and servers

TEXTBOOKS:

1. Raja CSP Raman, Ludovic Dewailly, ¯Building RESTful Web Services with Spring 5‖, Packt Publishing, 2018.
2. Leonard Richardson, Sam Ruby ¯RESTful Web Services‖ O'Reilly Media, 2008.
3. Ludovic Dewailly, ¯Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework‖, Packt Publishing, 2015
4. Raja CSP Raman, Ludovic Dewailly, ¯Building RESTful Web Services with Spring 5 – Second Edition
5. Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0‖, Packt Publishing, 2018

REFERENCES:

1. Ranga Karanam, ¯Master Java Web Services and REST API with Spring Boot‖, Packt Publishing, 2018.
2. Balaji Varanasi, Sudha Belida, ¯Spring REST‖, Apress, 2015.
3. Greg L. Turnquist, ¯Learning Spring Boot 2.0‖, Packt Publishing, 2021
4. Sourabh Sharma, ¯Modern API Development with Spring and Spring Boot‖, Packt Publishing, 2021

LIST OF EQUIPMENTS/SOFTWARE:

Java Persistence API, Spring Boot

FULL STACK TECHNOLOGY

22CS913	UI/UX DESIGN (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners:

- Explain the principles of User Interface (UI) in order to do design with intention.
- Define the User eXperience (UX) and the psychology behind user decision making.
- Discuss about UX process and user Psychology.
- Apply technology for designing web applications with multimedia effects.
- Create a wireframe and prototype.

UNIT I	INTRODUCTION TO UI	6+6

Introduction to UI - Designing Behaviour: Designing with Intention - Conditioning and Addiction - Timing Matters - Gamification - Social/Viral Structure–Trust - Hidden versus Visible. Basic Visual Design Principles: Visual Weight - Contrast - Depth and Size – Color- Layout: Page Framework - Footers - Navigation -Images, and Headlines - Forms - Input Types - Labels and Instructions - Primary and Secondary Buttons - Adaptive and Responsive Design - Touch versus Mouse.

List of Exercise/Experiments:

1. Design UI for a Game website.
2. Design one-page UI for a website.

UNIT II	USER OBSERVATION AND EXPERIENCE	6+6

User Research - Subjective Research - Objective Research - Sample size - Three Basic Types of Questions. Observe a user: Watch How They Choose - Interviews - Surveys - Card Sorting - Creating User Profiles - Bad profile - Useful profile.

List of Exercise/Experiments:

1. Design UI for a mobile.
2. Explore the Look and Feel of the new Project developed in Ex1.

UNIT III	INTRODUCTION TO UX	6+6

Introduction about UX - Five Main Ingredients of UX - Three ¯Whats‖ of user Perspective - Pyramid of UX Impact - UX Is a Process - UX - Not an Event or Task. Behaviour Basics: Psychology versus Culture - User Psychology - Experience - Conscious vs Subconscious Experience - Emotions - Gain and Loss – Motivations.

List of Exercise/Experiments:

1. Design a mascot for an imaginary brand.
2. Create a Sample Pattern Library for a product (Mood board, Fonts, Colors based on UI principles).

UNIT IV	WEB INTERFACE DESIGN	6+6

Designing Web Interfaces – Drag and Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow - Using Motion for UX - Design Pattern: Z-Pattern - F-Pattern - Visual Hierarchy - Lookup patterns – Feedback patterns.

List of Exercise/Experiments:

1. Design a mock-up website for a service sector company.
2. Create a brainstorming feature for proposed product.

UNIT V	WIREFRAMING, PROTOTYPING AND TESTING	6+6

Sketching Principles - Sketching Red Routes - Responsive Design – Wireframing - Creating Wire flows - Building a Prototype - Building High-Fidelity Mock-ups - Designing Efficiently with Tools - Interaction Patterns - Conducting Usability Tests - Other Evaluative User Research Methods - Synthesizing Test Findings - Prototype Iteration

List of Exercise/Experiments:

1. Sketch, design with popular tool and build a prototype and perform usability testing and identify improvements.
2. Design a mobile mock-up website for an online store.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the principles of User Interface (UI) Design in order to design with intention

CO2: Learn the effective User eXperience (UX) and the psychology behind user decision making.

CO3: Understand the importance of UX process and user Psychology.

CO4: Elucidate the implications for designing web application with multimedia effects.

CO5: Create Wireframe and Prototype.

TEXT BOOKS:

1. Joel Marsh, ¯UX for Beginners‖, O'Reilly Media, Inc., 1st Edition 2015.
2. Xia Jiajia, ¯UI UX Design‖, O'Reilly, Artpower International, 2016.
3. Jenifer Tidwell, Charles Brewer, Aynne Valencia, ¯Designing Interface‖ 3rd Edition, O'Reilly 2020

REFERENCES:

1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, ¯Designing Interface‖ 3rd Edition , O'Reilly 2020.
2. Steve Schoger, Adam Wathan ¯Refactoring UI‖, 2018.
3. https://www.uxai.design/#:~:text=for%20designers,for%20AI%20products%20and%20services.

SOFTWARE REQUIREMENTS:

Javascript, Applets, Equivalent Frontend tools, MySQL, Figma or equivalent.

22CS914	MERN FULL STACK DEVELOPMENT	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Design applications using Node .JS
- Create architecture involving Express and graphQL
- Develop applications using mongoDB
- Apply the concepts of React Components and State
- Build web applications using React Router, Forms and Bootstrap

UNIT I	INTRODUCTION TO MERN and NODE JS	6+6

Introduction - MERN Components - Node JS: Introduction to Node JS, Setting up Node.js,

Node.js Modules - HTTP Servers and Clients - Request Handling - Database connectivity - Data Storage and Retrieval - Dynamic Client/Server Interaction with Socket.IO

List of Exercise/Experiments:

1. Create your own modules and return Current date and time.
2. Create the HTTP server using createServer() method that listens to server ports and gives a response back to the client.

UNIT II	EXPRESS	6+6

Express - Routing - Request Matching - Route parameters - Route Lookup - Handler Function - Request Object - Response Object - Middleware - REST API - GraphQL - About API - List API - List API Integration - Custom Scalar Types - Create API Integration - Query Variables - Input Validations - Displaying Errors

List of Exercise/Experiments:

1. Create an application using Express.js to print Hello world on the Homepage.
2. Build a Simple Node.js/Express server that handles GET and POST request and returns data in JSON format.

UNIT III	MongoDB	6+6

MongoDB Basics - CRUD Operations - NODE.js driver - Schema Initialization - Reading from MongoDB - Writing to MongoDB - UI Server - Multiple Environments - Proxy-based Architecture - ESLint - ESLint for Front End - React PropTypes - Back End Modules - Front End Modules and Webpack - Transform and bundle - Libraries Bundle - Module Replacement - Debugging - Defineplugin - Product Optimization.

List of Exercise/Experiments:

1. Build an application to perform Basic CRUD operation in MongoDB using Node/Express.
2. Building the MongoDB database for the My To-do List app.

UNIT IV	REACT COMPONENTS AND STATE	6+6

React Components- Issue Tracker - React Classes - Composing Components - Passing Data - Dynamic Composition - React State - Hooks - Event handling - Stateless Components - Designing Components.

List of Exercise/Experiments:

1. Write a program to create a simple calculator Application using React JS
2. Build a simple React application that displays the list of items and allows the user to add new items to the list.

UNIT V	REACT ROUTER, FORMS AND BOOTSTRAP	6+6

React Router - Simple Routing - Route Parameters - Query Parameters - Links - Programmatic Navigation - Nested Routes - React Forms - Controlled Components - Specialized Input Components - Update API - Delete API - React Bootstrap - Buttons -Navigation - Panels - Tables - Forms - Grid - Inline Forms - Horizontal Forms - Validation Alerts - Toasts – Modals.

List of Exercise/Experiments:

1. Create a Simple Login form using React JS.
2. Build an application for E-Commerce platform.
3. Build a full-stack MERN app that allows the user to register, login, and create a list of items that are stored in a MongoDB database. The app should also display the list of items using React components.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop applications using Node .js
CO2: Handle queries using GraphQL with Express server
CO3: Create applications based on mongoDB CRUD operations
CO4: Learn about dynamic composition and event handling
CO5: Implement React Forms and Bootstrap

TEXT BOOKS:

1. Vasan Subramanian, Pro MERN Stack - Full stack web app development, 2nd Edition, Apress, 2019 (Unit 2 to 5)
2. David Herron , Node.js Web Development - Fourth Edition, Packt Publishing, 2018. (Unit 1)

REFERENCES:

1. Adam Freeman, Essential TypeScript, Apress, 2019.
2. Shama Hoque, Full-Stack React Projects, 2nd edition, Apress,2022
3. Karl Seguin, ¯The Little Mongo DB Book‖, https://github.com/karlseguin/the-little-mongodb-book.
4. https://aws.amazon.com/education/awseducate/
5. http://packaging.ubuntu.com/html/packaging-new-software.html
6. https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm

LIST OF EQUIPMENTS:

Node, Express, MongoDB, React

22CS915	MOBILE ARCHITECTURE AND APPLICATION DEVELOPMENT (Lab Integrated)	L		P	C
		2		2	3

OBJECTIVES:

The Course will enable learners to:

- Understand Android SDK
- Understand and develop Android applications
- Learn the concepts of iOS application development
- Develop Hybrid platform application
- Develop cross-platform applications with event handling

UNIT I	MOBILE ARCHITECTURE AND BASICS IN APP DEVELOPMENT	6+6

Mobile Device Operating Systems – Special Constraints & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Mobile App Architecture: Android - iOS - Hybrid - Cross Platform - Factors of Mobile App Architecture - Windows Phone — MCommerce — Structure — Pros & Cons — Mobile Payment System – Security Issues.

List of Exercise/Experiments

1. Develop an application that uses GUI components, Font, Layout Managers and event listeners.
2. Develop a native application that uses GPS location information.

UNIT II	ANDROID APPLICATION	6+6

Introduction to Android: The Android Platform - Android SDK, Eclipse Installation, Android

Installation, Building First Android application - Life cycle of android application - Android User Interface Design Essentials - Android APIs - Deploying android applications.

List of Exercise/Experiments

1. Create an android application that converts the user input text to voice.
2. Develop a Mobile application for simple and day to day needs.

UNIT III	IOS APPLICATION	6+6

Introduction - Performance metrics - Core optimizations - concurrent programming - Application life cycle - User Interface - Network -Security - Testing - Tools - Instrumentation and Analytics

List of Exercise/Experiments

1. Implement an iOS app allowing users to type or paste text and have it spoken aloud using Apple's speech synthesis.
2. Develop an ios application for simple and day to day needs.

UNIT IV	HYBRID APP ENVIRONMENT	6+6

Hybrid Web App, Benefits of Hybrid App, Criteria for creating Native App, Tools for creating Hybrid App, Cons of Hybrid App, Popular Hybrid App Development Frameworks, Ionic, Apache Cordova

List of Exercise/Experiments

1. Design an android application using Cordova for a user login screen with username, password, reset button and a submit button. Also, include header image and a label. Use layout managers.
2. Design and develop an android application using Apache Cordova to find and display the current location of the user.

UNIT V	CROSS-PLATFORM APP DEVELOPMENT	6+6

What is Cross-platform App, Benefits of Cross-platform App, Criteria for creating Cross-platform App, Tools for creating Cross-platform App, Cons of Cross-platform App, Popular Cross-platform App Development Frameworks, Flutter, Xamarin, React-Native, Basics of React Native, Native Components, JSX, State, Props

List of Exercise/Experiments

1. Design and develop a cross platform application for day to day task (to-do) management.
2. Build a cross platform application for a simple expense manager which allows entering expenses and income on each day and displays category wise weekly income and expense.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Identify various concepts of mobile architecture.

CO2: Create, test and debug Android application by setting up Android development

CO3: Develop iOS application with performance metrics

CO4: Develop hybrid applications with basic event handling.

CO5: Implement cross platform applications with basic GUI and event handling.

CO6: Deploy applications to the Android Market place for distribution.

TEXT BOOKS:

1. Gaurav Vaish, ¯High Performance iOS Apps‖, O"Reilly Media, 1st ed. 2016.

2. Lauren Darcey and Shane Conder, ―Android Wireless Application Development‖, Pearson Education, 2nd ed.2012.
3. Prasanth Kumar Pattnaik, Rajib Mall,‖ Fundamentals of Mobile Computing‖, PHI Learning Pvt. Ltd, New Delhi-2012.

REFERENCES:

1. Building Cross-Platform Mobile and Web Apps for Engineers and Scientists: An Active Learning Approach, Cengage Learning, 2016.
2. Mahesh Panhale, Beginning Hybrid Mobile Application Development, Apress,2016.
3. Android Application development James C. Sheusi Cengage learning 2017.
4. Erik Hellman, ―Android Programming – Pushing the Limits‖, 1st Edition, Wiley India Pvt Ltd, 2014.

LIST OF SOFTWARES:

1. Software required: Android Studio, macOS.
2. Software Tools required: Android 11 SDK, Xcode.

22CS916	MICROSERVICE ARCHITECTURE	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Learn the Concepts of Micro Service Architecture
- Understand the inter process communication in a microservice architecture
- Design business logic in a microservice architecture
- Develop business logic with event sourcing
- Implement queries in a microservice architecture

UNIT I	INTRODUCTION TO MICROSERVICE ARCHITECTURE	9

Introduction - Microservice - Microservice Architecture - Comparing Microservice Architecture with SOA - Benefits and Drawbacks of Microservice Architecture - Microservice Architecture Pattern Language - Beyond microservices -Process and Organization - Microservice Architecture Style - Application Microservice Architecture.

UNIT II	INTERPROCESS COMMUNICATION AND TRANSACTION MANAGEMENT	9

Overview of Inter process Communication - Synchronous Remote Procedure Invocation Pattern - Asynchronous Messaging Pattern - Asynchronous Messaging to Improve Availability - Transaction Management in a Microservice Architecture - Coordinating sagas - Handling the Lock of Isolation - Order Service and Order Saga.

UNIT III	DESIGN AND DEVELOPMENT OF BUSINESS LOGIC	9

Business logic organization patterns - Designing a domain model using the DDD aggregate pattern - Publishing domain events - Kitchen Service business logic - Order Service business logic - Developing business logic with event sourcing - Implementing an event store - Using sagas and event sourcing together.

UNIT IV	QUERY IMPLEMENTATION, EXTERNAL API PATTERNS AND MICROSERVICE TESTING	9

Querying using the API composition pattern - Using the CQRS pattern - Designing CQRS views - Implementing a CQRS view with AWS DynamoDB - External API design issues - The API gateway pattern - Implementing an API gateway - Testing strategies for microservice architectures - Writing unit tests for a service - Writing integration tests - Developing component tests - Writing end-to-end tests.

UNIT V	MICROSERVICES DEVELOPMENT AND DEPLOYMENT	9

Developing secure services - Designing configurable services - Designing observable services - Developing services using the Microservice chassis pattern - Deploying Microservices - Language-specific packaging format pattern - Service as a virtual machine pattern - Service as a container pattern - Serverless deployment - Refactoring to microservices.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Develop microservice applications.
CO2: Deploy microservices in different platforms
CO3: Create integration and component testing for microservices
CO4: Design External API patterns for microservices
CO5: Develop inter process communication mechanisms in microservice architecture
CO6: Create transaction management in microservice

TEXT BOOKS:

1. Chris Richardson, ―Microservices Pattern with Examples in JAVA‖, Manning Publications, 1st edition, 2019.
2. Sam Newman, ―Building Microservices: Designing Fine-Grained Systems‖, 2nd edition, O'Reilly Publication, 2021.

REFERENCES:

1. Irakli Nadareishvili, Ronnie Mitra, Matt McLarty & Mike Amundsen, ―Microservice Architecture ALIGNING PRINCIPLES, PRACTICES, AND CULTURE,‖ 1st edition, O'Reilly Publication, 2016.
2. Eberhard Wolff, ―Microservices Flexible Software Architectures,‖ 1st edition, Addison-Wesley, 2016.
3. John Kernal, IIlary Huaylupo Sanchez, ―Micro Service in Action‖, 2nd edition, Manning Publication,2018.
4. Susan J Fowler, ―Production-Ready Microservices‖, O'Reilly Publication, 2016.

22CS917	WEB APPLICATION SECURITY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the fundamentals of web application security
- Focus on wide aspects of secure development and deployment of web applications
- Learn how to build secure APIs
- Learn the basics of vulnerability assessment and penetration testing
- Get an insight about Hacking techniques and Tools

UNIT I	FUNDAMENTALS OF WEB APPLICATION SECURITY	9

The history of Software Security-Recognizing Web Application Security Threats, Web

Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation.		
UNIT II	**SECURE DEVELOPMENT AND DEPLOYMENT**	**9**
Web Applications Security - Security Testing, Security Incident Response Planning, The Microsoft Security Development Lifecycle (SDL), OWASP Comprehensive Lightweight Application Security Process (CLASP), The Software Assurance Maturity Model (SAMM). Case Study: Use the OWASP ZAP tool to identify vulnerabilities.		
UNIT III	**SECURE API DEVELOPMENT**	**9**
API Security- Session Cookies, Token Based Authentication, Securing Natter APIs: Addressing threats with Security Controls, Rate Limiting for Availability, Encryption, Audit logging, Securing service-to-service APIs: API Keys, OAuth2, Securing Microservice APIs: Service Mesh, Locking Down Network Connections, Securing Incoming Requests.		
UNIT IV	**VULNERABILITY ASSESSMENT AND PENETRATION TESTING**	**9**
Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database-based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.		
UNIT V	**HACKING TECHNIQUES AND TOOLS**	**9**
Social Engineering, Injection, Cross-Site Scripting (XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc. Case Study: Attack the website using Social Engineering method.		
		45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understanding the basic concepts of web application security and the need for it.

CO2: Develop a thorough understanding of the process for securely designing, implementing, testing, and deploying web applications.

CO3: Acquire the skill to design and develop Secure Web Applications that use Secure APIs.

CO4: Develop the ability to grasp the importance of conducting vulnerability assessments and penetration testing.

CO5: Acquire the skill to think like a hacker and to use hackers tool sets.

CO6: Develop the ability to implement and manage web application security measures effectively, ensuring ongoing protection against evolving threats.

TEXT BOOKS

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.
2. Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies.
3. Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA.

REFERENCES

1.Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing,

Inc.
2.Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.
3.Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.
4.Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.
5.Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

22CS903	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand block chain system's fundamental components, how they fit together and examine a decentralization using block chain.
- To explain how Crypto currency works.
- To explain the components of Ethereum and Programming Languages for Ethere
- To study the basics of Web3 and Hyper ledger.
- To give an insight of alternative block chains and its emerging trends.

UNIT I	INTRODUCTION TO BLOCKCHAIN	9

History of Blockchain – Types of Blockchain – Consensus – Decentralization using Blockchain – Blockchain and Full Ecosystem Decentralization – Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography - public and private keys - Elliptic curve cryptography - Discrete logarithm problem in ECC.

UNIT II	INTRODUCTION TO CRYPTOCURRENCY	9

Bitcoin – Digital Keys and Addresses – Transactions – Mining – Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin. Case study - Web3j.

UNIT III	ETHEREUM	9

The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code - Blocks and Blockchain - Fee Schedule - Supporting Protocols - Solidity Language.

UNIT IV	WEB3 AND HYPERLEDGER	9

Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger – Case study - Corda.

UNIT V	ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS	9

Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research - Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the technology components of Blockchain and how it works behind the scenes.
CO2: Understand the Bitcoin and its limitations by comparing with other alternative coins.
CO3: Develop deep understanding of the Ethereum model, its consensus model, code execution.
CO4: Understand the architectural components of a Hyperledger and its development framework.
CO5: Explore the alternative blockchains and its emerging trends.
CO6: Understand blockchain technology, including Bitcoin, Ethereum, and alternative blockchains.

TEXT BOOKS:

1. Imran Bashir, ―Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained‖, Second Edition, Packt Publishing, 2018.
2. Arshdeep Bahga, Vijay Madisetti, ―Blockchain Applications: A Hands-On Approach‖, VPT, 2017.

REFERENCES:

3. Andreas Antonopoulos, Satoshi Nakamoto, ―Mastering Bitcoin‖, O‘Reilly Publishing, 2014.
4. Roger Wattenhofer, ―The Science of the Blockchain‖ CreateSpace Independent Publishing Platform, 2016.
3. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, ―Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction‖, Princeton University Press, 2016.
4. Alex Leverington, ―Ethereum Programming‖, Packt Publishing, 2017.
5. Antony Lewis ―The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them‖, Mango Publishing 2018.
6. Andreas M. Antonopoulos, ―Mastering Bitcoin: Programming the Open Block chain‖, O‘Reilly Publishing, 2017.
7. Massimo Ragnedda, Giuseppe Destefanis, ―Blockchain and Web 3.0: Social, Economic, and Technological Challenges‖, Routledge, 2019.

22CS918	SOFTWARE TESTING AND AUTOMATION	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Introduce the basics and necessity of software testing.
- Provide various testing techniques along with concepts of software bugsand its impact.
- Develop and validate a test plan.
- Build a testing team required.
- Understand the need for and challenges in test automation and to developtesting scripts.

UNIT I	FOUNDATIONS OF SOFTWARE TESTING	9

Why do we test Software? Stages of Testing, Software Testing Principles, Software Testing Life Cycle, Program Correctness and Verification, Reliability versus Safety, Failures, Errors and Faults (Defects), Case Study: Test plan for testing an e-commerce web/mobile application (Amazon).

UNIT II	TEST PLANNING	9

The Goal of Test Planning, High Level Expectations, Intergroup Responsibilities, Test Phases,

Test Strategy, Resource Requirements, Tester Assignments, Test Schedule, Test Cases, Bug Reporting, Metrics and Statistics.

UNIT III	TEST DESIGN AND EXECUTION	9

Test Objective and Requirement Identification, Test Design Factors, Modeling a Test Design Process, Model and Results, Boundary Value Testing, Equivalence Class Testing, Path Testing, Data Flow Testing, Test Design Metrics and Effectiveness, Test Procedures. Case Study: Execute the test cases on the client-server or desktop application to identify any defect.

UNIT IV	ADVANCED TESTING CONCEPTS	9

Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-Over Testing, Recovery Testing, Configuration Testing, Compatibility Testing, Usability Testing, Testing the Documentation, Security testing, Testing in the Agile Environment, Testing Web and Mobile Applications.

UNIT V	TEST AUTOMATION AND TOOLS	9

Software Test Automation – Skill Needed for Automation – Scope of Automation – Design and Architecture for Automation – Requirements for a Test Tool – Challenges in Automation – Test Metrics and Measurements – Project, Progress and Productivity Metrics – Maintenance of Documents During Testing. Case Study: Install and learn popular software testing tools like Selenium, WinRunner, LoadRunner, Performance Tester etc.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basic concepts of software testing and the need for software testing.

CO2: Design Test planning and different activities involved in test planning.

CO3: Design effective test cases that can uncover critical defects in the application.

CO4: Carry out advanced types of testing.

CO5: Automate the software testing using Selenium and TestNG.

CO6: Analyze and evaluate the results of testing to ensure software quality, and implement continuous improvement processes based on testing outcomes.

TEXT BOOKS:

1. Yogesh Singh, ¯Software Testing‖, Cambridge University Press, 2012

2. Dorothy Graham, Mark Fewster, ¯Experiences of Test Automation: Case Studies of Software Test Automation‖, Pearson Education, 2012.

3. Unmesh Gundecha, Satya Avasarala, "Selenium WebDriver 3 Practical Guide" - Second Edition 2018.

4. Paul C. Jorgensen, ¯Software Testing‖, A Craftsman Approach, CRC Press, Fourth Edition, 2013.

REFERENCES:

1. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing, 3rd Edition, 2012, John Wiley& Sons, Inc.

2. Ron Patton, Software testing, 2nd Edition, 2006, Sams Publishing.

3. Mark Fewster, Software Test Automation, 1999, Addison-Wesley.

4. Ali Mili, Fairouz Chier, ¯Software Testing: Concepts and Operations‖, Wiley, 2015.

22CS910	DEVOPS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Bridge the gap between development and operations for faster, more reliable software releases.
- Automate software delivery with CI/CD pipelines.
- Package and deploy apps efficiently using Docker containers.
- Automate infrastructure with Infrastructure as Code (IaC).
- Monitor and troubleshoot applications in production.

UNIT I	INTRODUCTION TO DEVOPS	9

Software Development Methodologies - Operations Methodologies - Systems Methodologies - Development, Release, and Deployment Concepts - Infrastructure Concepts. What is DevOps? - DevOps importance and benefits -DevOps principles and practices - 7 C's of DevOps lifecycle for business agility - DevOps and continuous testing. How to choose right DevOps tools? - Challenges with DevOps implementation.

UNIT II	VERSION CONTROL WITH GIT	9

Introduction to Git version control system - Git commands for basic operations (clone, commit, push, pull) - Branching and merging strategies - Collaboration using Git workflows.

UNIT III	CONTINUOUS INTEGRATION AND DELIVERY (CI/CD)	9

Introduction to CI/CD pipelines - Benefits of CI/CD for faster deployments - Setting up a CI/CD pipeline with Jenkins - Automating builds, tests, and deployments.

UNIT IV	CONTAINERIZATION WITH DOCKER	9

Introduction to containerization and its benefits - Understanding Docker concepts: images, containers, registries - Building and managing Docker containers - Docker Compose for multi-container applications - Introduction to container orchestration with Docker Swarm or Kubernetes.

UNIT V	INFRASTRUCTURE AS CODE (IAC) AND MONITORING	9

Introduction to Infrastructure as Code (IaC) - Benefits of using IaC for repeatable infrastructure provisioning - Learning IaC with Terraform - Setting up infrastructure configurations with Terraform - Introduction to monitoring and logging tools for applications - Alerting and troubleshooting techniques.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the core principles and philosophies of DevOps.
CO2: Implement version control systems for code management and collaboration.
CO3: Automate software delivery pipelines using CI/CD tools.
CO4: Utilize containerization technologies for packaging and deploying applications.
CO5: Configure infrastructure as code (IaC) for repeatable deployments.
CO6: Monitor and maintain applications in a production environment.

TEXT BOOKS:

3. Deepak Gaikwad, Viral Thakkar, "DevOps Tools: from Practitioner's Point of View", Wiley, 2019.
4. Jennifer Davis, Ryn Daniels, "Effective DevOps", O'Reilly Media, 2016.

REFERENCES:
5. Gene Kim, Jez Humble, Patrick Debois, "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", IT Revolution Press, 2016.
6. Jez Humble, Gene Kim, "Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation", Addison-Wesley, 2010.
7. Yevgeniy Brikman, "Terraform: Up & Running: Writing Infrastructure as Code", O'Reilly Media, 2019.
8. Joseph Muli, "Beginning DevOps with Docker", Packt Publishing, 2018.

22CS919	USABILITY DESIGN OF SOFTWARE APPLICATIONS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To create a learning system through which management students can enhance their innovation.
- To acquaint themselves with the special challenges of starting new ventures.
- To use IPR as an effective tool to protect their innovations and intangible assets from exploitation.
- To develop a learning system that enables the creative thinking skills.

UNIT I	INTRODUCTION TO USER CENTERED DESIGN AND ASPECT	9

Basics of User Centered Design- Product Appreciation Assignment – Evaluating the product from user centered design aspects such as functionality, ease of use, ergonomics, and aesthetics.
Case Study: Identify a website or an App to redesign, with justification.

UNIT II	HEURISTIC EVALUATION AND PROJECT DESIGN LIFECYCLE	9

10 Heuristic Principles, Examples Heuristic Evaluation: Group Assignment initiation (Website and App) Evaluation for key tasks of the app or website for heuristic principles, severity, recommendations. Case Study: Analyze the mobile app or website throughout the design life cycle.

UNIT III	PROJECT DESIGN LIFECYCLE	9

Redesign project through the design lifecycle – Discovery - Define – Design - Implement (Design Prototype)-Usability Testing. Case Study: Identify personas and scenarios for the app or website

UNIT IV	UX RESEARCH	9

Understanding users, their goals, context of use, and environment of use. Research Techniques: Contextual Enquiry, User Interviews, Competitive Analysis for UX. Scenarios and Persona Technique -Overview of Design Thinking Technique - Discovery andbrainstorming.

UNIT V	DEVELOPMENT AND PROTOTYPING	9

Concept Development - Task flow detailing for the Project – Prototyping Techniques - Paper, Electronic, and Prototyping Tools. Case Study: Prototype development with Iterations and justification.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Sensitize the students to the fundamentals of User Centred Design and User

Experience theirrelevance and contribution to businesses.

CO2: Familiarize them to the facets of User Experience (UX) Design, particularly as applied to the digitalartefacts.

CO3: Appreciation of user research, solution conceptualization and validation as interwoven activities in thedesign and development lifecycle.

CO4: Acquire the ability to constructively engage with the Design professionals they would work with in thefuture.

CO5: Analyse and identify the methods to offer a better UI experience for the applications
Gain expertise in redesigning an existing Application or website for better user experience.

CO6: Develop the capability to integrate user feedback and usability testing results into iterative design processes, ensuring continuous improvement and alignment with user needs.

TEXT BOOKS:

1. Jennifer Preece, Helen Sharp, Yvonne Rogers, ―Interaction Design: Beyond Human-Computer Interaction‖, 2015, 4th Edition, Wiley publications.
2. Bruce Ferwerda, Mark Graus, Marko Tkalčič, Panagiotis Germanakos, A Human-centered Perspective of Intelligent Personalized Environments and Systems,2024, Springer.
3. Steve Krug, ‖Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability‖2013, 3rd Edition, New Riders.

REFERENCES:

1. Alan Cooper and Robert Riemann, ―About Face The Essentials of Interaction Design‖, 2014, 4th Edition, Wiley Publications.
2. Elizabeth Goodman, Mike Kuniavsky, Andrea Moed , ― Observing the User Experience – A Practitioner's Guide to User Research‖ , 2012, Second Edition, Morgan Kaufmann Publications.
3. David Benyon,‖Designing Interactive Systems- A Comprehensive Guide to UX and Interaction Design‖,2013,Pearson Publications.

22CS920	GENERATIVE AI FUNDAMENTALS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basic concepts of Generative AI.
- Build Generative AI systems to generate images.
- Understand the concept used in Generative AI Models.
- Use various Generative AI models.
- Compare and use the various Large Language Models.
- Understand the basics of Prompt Engineering.

UNIT I	INTRODUCTION	9
Generative Models – Image transformation – Challenges - Deep Neural Networks – Perceptron – back propagation – CNN – RNN – Optimizer.		
UNIT II	**IMAGE GENERATION**	**9**
Creating encodings of images – variational objective – Inverse Autoregressive flow – Importing CIFAR – Creating the network from TensorFlow 2.		
UNIT III	**GENERATIVE ADVERSARIAL NETWORKS**	**9**

Generative Adversarial Networks – Vanilla GAN – Improved GANs – Progressive GAN – Challenges – Paired style transfer – Unpaired style transfer – Deepfakes – Modes of operation – key feature set – High level flow – Replacement – Re-enactment.

UNIT IV	LARGE LANGUAGE MODELS	9

Overview of LLMs - Transformers – GPT – Types of LLMs – Key concepts – other Transformers – T5 – Generative Pre-Training Models – Multi-modal Models – DALL.E 2

UNIT V	PROMPT ENGINEERING	9

Basics – In-Context Learning – In-Context Prompting – Techniques – Image Prompting – Prompt Hijacking – Challenges.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Elaborate the basic concepts of Generative AI.
CO2: Build Generative AI systems to generate images.
CO3: Apply the concepts used in Generative AI Models.
CO4: Use various Generative AI models.
CO5: Compare and use the various Large Language Models.
CO6: Analyze the basics of Prompt Engineering.

TEXT BOOKS:

1. Ben Auffarth, Generative AI with LangChain, Packt Publishing, 2023.
2. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.

REFERENCES:

1. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023.
2. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024.
3. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.

22IT910	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework.
- Cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL.
- Explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations.
- Implement and manage Spring AOP applications using annotation-based configurations for method interception and post-execution operations.
- Build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.

UNIT I	INTRODUCTION TO REST API	6+6

RESTful APIs – overview about data exchange between client and server - separating concerns between handling HTTP requests and executing business logic - retrieving server resources via HTTP requests - injection of property values - self-contained application - serialization and deserialization – JSON properties -

managing data access.

List of Exercises/Experiments:

1. Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data exchange between client and server.
2. Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles.
3. Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file.
4. Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control

UNIT II	ADVANCED DATA MANAGEMENT WITH JAVA AND MYSQL	6+6

Build production-grade applications – MYSQL - mapping Java classes to relational database - repository interface - data access operations – retrieving data from the database –mapping of request body to entity - retrieve an entity - capture data from API requests - building complex queries using keywords.

List of Exercises/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.

UNIT III	ADVANCED JPA QUERIES AND ANNOTATIONS	6+6

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercises/Experiments:

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations.
3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns.
4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.

UNIT IV	JPA ASSOCIATIONS AND MAPPING	6+6

JPA Mapping of One-to-One Associations - fetching entities using queries – Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA.

List of Exercises/Experiments:

1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities.
2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities.
3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to

retrieve employee details efficiently

UNIT V	SPRING BOOT ESSENTIALS: API SECURITY, LOGGING, AOP, AND BUILD MANAGEMENT	6+6

SwaggerUI with Spring Boot, OpenUI with Spring Boot, Logging with Spring Boot, Changing Log Level, Logging Request and Response- Managing Spring Boot Logging Configuration - Aspect-Oriented Programming (AOP) Concepts – Method Parameter Handling - Post- Execution Operations - Returning Data Handling - Comprehensive Advice Handling. API security using JWT, Gradle for build management, Sonar Lint for coding standards and guidelines.

List of Exercises/Experiments:

1. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction.
2. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively.
3. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Create simple applications using RESTful APIs and effectively manage HTTP methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6:

TEXTBOOKS:

1. Raja CSP Raman, Ludovic Dewailly, ¯Building RESTful Web Services with Spring 5‖, Packt Publishing, 2018.
2. Leonard Richardson, Sam Ruby ¯RESTful Web Services‖ O'Reilly Media, 2008.
3. Ludovic Dewailly, ¯Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework‖, Packt Publishing, 2015
4. Raja CSP Raman, Ludovic Dewailly, ¯Building RESTful Web Services with Spring 5 – Second Edition
5. Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0‖, Packt Publishing, 2018

REFERENCES:

1. Ranga Karanam, ¯Master Java Web Services and REST API with Spring Boot‖, Packt Publishing, 2018.
2. Balaji Varanasi, Sudha Belida, ¯Spring REST‖, Apress, 2015.
3. Greg L. Turnquist, ¯Learning Spring Boot 2.0‖, Packt Publishing, 2021
4. Sourabh Sharma, ¯Modern API Development with Spring and Spring Boot‖, Packt Publishing, 2021

LIST OF EQUIPMENTS/SOFTWARE:

Java Persistence API, Spring Boot

DATA SCIENCE

22AM901	DATA SCIENCE USING PYTHON (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To learn the fundamentals of Data Science.
- To experiment and implement python libraries for data science Learn the tools and packages in Python for Data Science.
- To apply and implement basic classification algorithms
- To apply clustering and outlier detection approaches.
- To present and interpret data using visualization libraries in Python

UNIT I	INTRODUCTION	6+6

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – data preparation - Exploratory Data analysis – build the model – presenting findings and building applications - Data Mining - Data Warehousing – Basic statistical descriptions of Data.

List of Exercise/Experiments:

1. Download, install and explore the features of R/Python for data analytics
    - Installing Anaconda
    - Basic Operations in Jupiter Notebook
    - Basic Data Handling

UNIT II	PYTHON LIBRARIES FOR DATA SCIENCE	6+6

Introduction to Numpy - Multidimensional Ndarrays - Indexing - Properties - Constants - Data Visualization: Ndarray Creation – Matplotlib - Introduction to Pandas – Series – Dataframes – Visualizing the Data in Dataframes - Pandas Objects – Data Indexing and Selection – Handling missing data – Hierarchical indexing – Combining datasets – Aggregation and Grouping – Joins- Pivot Tables - String operations – Working with time series – High performance Pandas.

List of Exercise/Experiments:

1. Working with Numpy arrays - Creation of numpy array using the tuple, Determine the size, shape and dimension of the array, Manipulation with array Attributes, Creation of Sub array, Perform the reshaping of the array along the row vector and column vector, Create Two arrays and perform the concatenation among the arrays.
2. Working with Pandas data frames - Series, DataFrame , and Index, Implement the Data Selection Operations, Data indexing operations like: loc, iloc, and ix, operations of handling the missing data like None, Nan, Manipulate on the operation of Null Vaues (is null(), not null(), dropna(), fillna()).
3. Perform the Statistics operation for the data (the sum, product, median, minimum and maximum, quantiles, arg min, arg max etc.).
4. Use any data set compute the mean ,standard deviation, Percentile.

UNIT III	CLASSIFICATION	6+6

Basic Concepts – Decision Tree Induction – Bayes Classification Methods – Rule-Based Classification – Model Evaluation and Selection.
Bayesian Belief Networks – Classification by Backpropagation – Support Vector Machines –

Associative Classification – K-Nearest-Neighbor Classifiers – Fuzzy Set Approaches - Multiclass Classification - Semi-Supervised Classification.

List of Exercise/Experiments:

1. Apply Decision Tree algorithms on any data set.
2. Apply SVM on any data set
3. Implement K-Nearest-Neighbor Classifiers

UNIT IV	**CLUSTERING AND OUTLIER DETECTION**	**6+6**

Cluster Analysis - Partitioning Methods – Evaluation of Clusters – Probabilistic Model-Based Clustering – Outliers and Outlier Analysis – Outlier Detection Methods – Statistical Approaches – Clustering and Classification-Based Approaches.

List of Exercise/Experiments:

1. Apply K-means algorithms for any data set.
2. Perform Outlier Analysis on any data set.

UNIT V	**DATA VISUALIZATION**	**6+6**

Importing Matplotlib – Simple line plots – Simple scatter plots – visualizing errors – density and contour plots – Histograms – legends – colors – subplots – text and annotation – customization – three dimensional plotting - Geographic Data with Basemap - Visualization with Seaborn.

List of Exercise/Experiments:

1. Basic plots using Matplotlib.
2. Implementation of Scatter Plot.
3. Construction of Histogram, bar plot, Subplots, Line Plots.
4. Implement the three dimensional potting.
5. Visualize a dataset with Seaborn.

TOTAL:30+30 = 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Explain the fundamentals of data science
CO2: Experiment python libraries for data science
CO3: Apply and implement basic classification algorithms
CO4: Implement clustering and outlier detection approaches
CO5: Present and interpret data using visualization tools in Python

TEXT BOOKS:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, ¯Introducing Data Science‖, Manning Publications, 2016. (Unit 1)
2. Ashwin Pajankar, Aditya Joshi, Hands-on Machine Learning with Python: Implement Neural Network Solutions with Scikit-learn and PyTorch, Apress, 2022.
3. Jake VanderPlas, ¯Python Data Science Handbook – Essential tools for working with data‖, O'Reilly, 2017.

REFERENCES:

1. Roger D. Peng, R Programming for Data Science, Lulu.com, 2016
2. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012.
3. Samir Madhavan, Mastering Python for Data Science, Packt Publishing, 2015
4. Laura Igual, Santi Seguí, "Introduction to Data Science: A Python Approach to Concepts,
5. Techniques and Applications", 1st Edition, Springer, 2017
6. Peter Bruce, Andrew Bruce, "Practical Statistics for Data Scientists: 50 Essential
7. Concepts", 3rd Edition, O'Reilly, 2017
8. Hector Guerrero, ―Excel Data Analysis: Modelling and Simulation‖, Springer International Publishing, 2nd Edition, 2019
9. NPTEL Courses:
    a. Data Science for Engineers - https://onlinecourses.nptel.ac.in/noc23_cs17/preview
    b. Python for Data Science - https://onlinecourses.nptel.ac.in/noc23_cs21/preview

22CS935	DATA EXPLORATION AND VISUALIZATION	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Outline an overview of exploratory data analysis and phases involved in data analytics
- Acquire an in-depth knowledge in EDA techniques
- Experiment the data visualization
- Describe the methods of time series analysis
- Explain the basics of tree and hierarchical representation of big data

UNIT I	EXPLORATORY DATA ANALYSIS	9

EDA fundamentals – Understanding data science – Significance of EDA – Making sense of data – Comparing EDA with classical and Bayesian analysis – Software tools for EDA

UNIT II	EDA TECHNIQUES	9

Visual Aids For EDA- Data transformation techniques-merging database, reshaping and pivoting, Transformation techniques -Descriptive Statistics-types of kurtosis, quartiles, Grouping Datasets-data aggregation, group wise transformation.

UNIT III	VISUALIZING DATA	9

The Seven Stages of Visualizing Data, Processing-load and displaying data – functions, sketching and scripting, Mapping-Location, Data, two sided data ranges, smooth interpolation of values over time

UNIT IV	TIME SERIES ANALYSIS	9

Overview of time series analysis-showing data as an area, drawing tabs, handling mouse input, Connections And Correlations – Preprocessing-introducing regular expression, sophisticated sorting, Scatterplot Maps-deployment issues

UNIT V	TREES, HIERARCHIES, AND RECURSION	9

Treemaps - treemap library, directory structure, maintaining context, file item, folder item, Networks and Graphs-approaching network problems-advanced graph example, Acquiring data, Parsing data

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Explain the overview of exploratory data analysis and phases involved in data analytics
CO2: Explore in-depth knowledge in EDA techniques
CO3: Apply the visualization techniques in data
CO4: Describe the methods of time series analysis
CO5: Represent the data in tree and hierarchical formats
CO6: Apply tools is visualization to represent data.

TEXT BOOKS:

1. Suresh Kumar Mukhiya and Usman Ahmed, ―Hands-on Exploratory Data Analysis with Python‖, Packt publishing , March 2020.
2. Ben Fry, ―Visualizing Data‖, O'reilly publications, 2007.

REFERENCES:

1. Danyel Fisher & Miriah Meyer, ―Making Data Visual: A Practical Guide To Using Visualization For Insight‖, O'reilly publications, 2018.
2. Claus O. Wilke, ‖Fundamentals of Data Visualization‖, O'reilly publications, 2019.
3. EMC Education Services, ―Data Science and Big data analytics: Discovering, Analyzing, Visualizing and Presenting Data‖, Wiley Publishers, 2015.
4. Tamara Munzner, ―Visualization Analysis and Design‖, A K Peters/CRC Press; 1st edition, 2014.
5. Matthew O. Ward, Georges Grinstein, Daniel Keim, ―Interactive Data Visualization: Foundations, Techniques, and Applications‖, 2nd Edition, CRC press, 2015.

22AM904	TEXT AND SPEECH ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To introduce the tools and techniques for performing text and speech analytics in diverse contexts.
- To understand the tools and technologies involved in developing text and speech applications.
- To demonstrate the use of computing for building applications in text and speech processing.
- To use information Retrieval Techniques to build and evaluate text processing systems.
- To apply advanced speech recognition methodologies in practical applications.

UNIT I	TEXT PROCESSING	9

Speech and Language Processing - Regular Expression - Text normalization – Edit Distance - Lemmatization – Stemming – N-gram Language Models - Vector Semantics and Embeddings.

UNIT II	TEXT CLASSIFICATION	9

Text Classification Tasks – Language Model – Neural Language Models – RNNs as Language Models – Transformers and Large Language Models.

UNIT III	QUESTION ANSWERING AND DIALOGUE SYSTEMS	9

Information Retrieval – Dense Vectors – Neural IR for Question Answering – Evaluating Retrieval-based Question Answering – Frame-based Dialogue Systems – Dialogue Acts and Dialogue State – Chatbots – Dialogue System Design.

UNIT IV	TEXT TO SPEECH SYNTHESIS	9

Automatic Speech Recognition Task – Feature Extraction for ASR: Log Mel Spectrum – Speech

Recognition Architecture – CTC - ASR Evaluation: Word Error Rate – TTS – Speech Tasks.

UNIT V	SPEECH RECOGNITION	9

LPC for speech recognition - Hidden Markov Model (HMM) - Training procedure for HMM-subword unit model based on HMM - Language models for large vocabulary speech recognition - Overall recognition system based on subword units - Context dependent subword units-Semantic post processor for speech recognition.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Apply the fundamental techniques in text processing for various NLP tasks.
CO2: Implement advanced language models and improve text classification accuracy.
CO3: Designing text processing systems using state-of-the-art techniques.
CO4: Design, implement, and evaluate ASR and TTS systems.
CO5: Apply advanced speech recognition methodologies in practical applications.
CO6: Use information Retrieval Techniques to build and evaluate text processing systems.

TEXT BOOKS:

1. Jurafsky, D. and J. H. Martin, Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Pearson Publication, Third Edition, 2022.
2. Lawrence Rabiner, Biing-Hwang Juang and B.Yegnanarayana, ¯Fundamentals of Speech Recognition‖, Pearson Education, 2009.

REFERENCES:

1. John Atkinson-Abutridy, Text Analytics: An Introduction to the Science and Applications of Unstructured Information Analysis, CRC Press, 2022.
2. Jim Schwoebel, NeuroLex, Introduction to Voice Computing in Python, 2018
3. Lawrence R. Rabiner, Ronald W. Schafe, Theory and Applications of Digital Speech Processing, First Edition, Pearson, 2010.
4. Srinivasa-Desikan, Bhargav. Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras. Packt Publishing Ltd, 2018.

22AM905	IMAGE AND VIDEO ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the basics of image processing techniques for computer vision and video analysis.
- To illustrate the techniques used for image pre-processing.
- To discuss the various image Segmentation techniques.
- To understand the various Object recognition mechanisms.
- To elaborate on the motion analysis techniques for video analytics.

UNIT I	INTRODUCTION	9

Computer Vision – Image representation and image analysis tasks - Image representations – digitization – properties – color images – Data structures for Image Analysis - Levels of image data representation - Traditional and Hierarchical image data structures.

UNIT II	IMAGE PRE-PROCESSING	9

Pixel brightness transformations - Geometric transformations - Local pre-processing - Image

smoothing - Edge detectors - Zero-crossings of the second derivative - Scale in image processing - Canny edge detection - Parametric edge models - Edges in multi-spectral images - Local pre-processing in the frequency domain - Line detection by local pre-processing operators - Detection of corners (interest points) - Detection of maximally stable extremal regions - Image restoration.

UNIT III	SEGMENTATION	9

Thresholding - Edge-based segmentation - Region-based segmentation – Matching - Evaluationissues in segmentation - Mean shift segmentation - Active contour models.

UNIT IV	OBJECT RECOGNITION	9

Knowledge representation - Statistical pattern recognition - Neural nets - Syntactic pattern recognition - Recognition as graph matching - Optimization techniques in recognition - Fuzzy systems - Boosting in pattern recognition - Random forests - Image understanding control strategies.

UNIT V	MOTION ANALYSIS	9

Differential motion analysis methods - Optical flow - Analysis based on correspondence of interestpoints - Detection of specific motion patterns - Video tracking - Motion models to aid tracking.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of image processing techniques for computer vision and video analysis.
CO2: Illustrate the techniques used for image pre-processing.
CO3: Analyze the various image Segmentation techniques.
CO4: Understand the various Object recognition mechanisms.
CO5: Elaborate on the motion analysis techniques for video analytics.
CO6: Apply image processing techniques in real-world applications.

TEXT BOOKS:

1. Milan Sonka, Vaclav Hlavac, Roger Boyle, ―Image Processing, Analysis, and MachineVision‖, 4nd edition, Thomson Learning, 2013.

REFERENCES:

1. Richard Szeliski, ―Computer Vision: Algorithms and Applications‖, Springer Verlag LondonLimited, 2011.
2. Caifeng Shan, Fatih Porikli, Tao Xiang, Shaogang Gong, ―Video Analytics for BusinessIntelligence‖, Springer, 2012.
3. D. A. Forsyth, J. Ponce, ―Computer Vision: A Modern Approach‖, Pearson Education, 2003.
4. E. R. Davies, ―Computer & Machine Vision‖, Fourth Edition, Academic Press,2012.

22AM906	STREAM PROCESSING AND ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To outline the framework for real time stream processing.
- To learn various algorithms for data streaming.
- To identify frequent item sets by mining from data streams.
- To introduce approaches to evaluate stream learning algorithms.
- To use tools for distributed data flow management.
- To design solutions to stream processing problems.

UNIT I	INTRODUCTION TO DATA STREAMS	9
Data Stream Models – Bounds of Random variables – Poisson Process – Maintaining Simple Statistics from Data Streams – Sliding Window and computing statistics over sliding windows – Data Synopsis – Sampling – Histograms – Wavelets – DFT - Change Detection: Tracking Drifting Concepts - Monitoring the Learning Process.		
UNIT II	**STREAMING ALGORITHMS**	**9**
Clustering Examples: Basic Concepts - Partitioning Clustering – Hierarchical Clustering - Micro Clustering – Grid Clustering - Clustering Variables - The Very Fast Decision Tree Algorithm (VFDT) - The Base Algorithm, Analysis of the VFDT Algorithm, Extensions to the Basic Algorithm: Processing Continuous Attributes, Functional Tree Leaves, Concept Drift.		
UNIT III	**FREQUENT PATTERN MINING**	**9**
Introduction – Heavy Hitters - Mining Frequent Itemsets from Data Streams - Landmark Windows - Mining Recent Frequent Itemsets - Frequent Itemsets at Multiple Time Granularities - Sequence Pattern Mining - Reservoir Sampling for Sequential Pattern Mining over data stream.		
UNIT IV	**EVALUATING STREAMING ALGORITHMS**	**9**
Learning from Data Streams - Evaluation Issues - Design of Evaluation Experiments - Evaluation Metrics - Comparative Assessment - Evaluation Methodology in Non-Stationary Environments.		
UNIT V	**DATA FLOW MANAGEMENT**	**9**
Distributed Data Flows – Apache Kafka – Apace Flume - Processing Streaming Data – Storing Streaming Data – Delivering Streaming Metrics.		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Outline the framework for real time stream processing.
CO2: Elaborate various algorithms for data streaming.
CO3: Illustrate frequent item sets by mining from data streams.
CO4: Apply the metrics and procedures to evaluate a model.
CO5: Use tools for distributed data flow management.
CO6: Develop solutions for real-world problems using streaming data.

TEXT BOOKS:

1. Joao Gama, ―Knowledge Discovery from Data Streams‖, CRC Press, 2010.
2. Byron Ellis, Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data, First Edition, WILEY Big Data Series, 2014.

REFERENCES:

1. Andrew Psaltis, Streaming Data: Paul Lewis, First Edition, Manning Publication, 2017.
2. Bugra Gedik, Deepak S. Turaga, Henrique C. M. Andrade, Fundamentals of Stream Processing: Application Design, Systems, and Analytics, Cambridge University Press, 2014.
3. Charu C. Aggarwal, "Data Streams: Models and Algorithms", Kluwer Academic Publishers, 2007.
4. David Luckham, "The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems", Addison Wesley, 2002.

22CS938	COGNITIVE SCIENCE AND ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand cognitive computing.
- To know about design principles and NLP for Cognitive systems.
- To distinguish between Big Data and Cognitive computing.
- To discuss implications of cognitive computing in business.
- To develop applications of cognitive computing.

UNIT I	FOUNDATIONS OF COGNITIVE SCIENCE	9
Foundation of Cognitive Computing: cognitive computing as a new generation- the uses of cognitive systems- system cognitive- gaining insights from data- Artificial Intelligence as the foundation of cognitive computing- understanding cognition.		
UNIT II	**DESIGN PRINCIPLES FOR COGNITIVE SYSTEMS AND NLP IN COGNITIVE SYSTEMS**	**9**
Components of a cognitive system- building the corpus- bringing data into cognitive system- machine learning- hypotheses generation and scoring- presentation and visualization services. Natural Language Processing in support of a Cognitive System: Role of NLP in a cognitive system- semantic web- Applying Natural language technologies to Business problems.		
UNIT III	**BIG DATA Vs COGNITIVE COMPUTING**	**9**
Relationship between Big Data and Cognitive Computing: Dealing with human-generated data- defining big data- architectural foundation- analytical data warehouses- Hadoop- data in motion and streaming data- integration of big data with traditional data.		
UNIT IV	**THE BUSINESS IMPLICATIONS OF COGNITIVE COMPUTING**	**9**
Preparing for change- advantages of new disruptive models- knowledge meaning to business- difference with a cognitive systems approach- meshing data together differently- using business knowledge to plan for the future- answering business questions in new ways- building business specific solutions- making cognitive computing a reality- cognitive application changing the market- IBM Watson as a cognitive system.		
UNIT V	**APPLICATIONS OF COGNITIVE COMPUTING**	**9**
Build a cognitive health care application - Build a cognitive application on Smarter cities - Applicate Cognitive Computing principle in building a Government related application.		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Elaborate the concepts of cognitive science and computing.
CO2: Design and Implementation of Cognitive Systems.
CO3: Apply NLP in cognitive systems.
CO4: Integrate Big Data and Cognitive computing.
CO5: Discuss implications of cognitive computing in business.
CO6: Develop various applications of cognitive computing.

TEXT BOOKS:

1. Judith H Hurwitz, Marcia Kaufman, Adrian Bowles, ¯Cognitive computing and Big Data Analytics‖ , Wiley, 2015.

REFERENCES:

1. Vijay Raghvan, Venu Govindaraju, C.R. Rao, ¯Cognitive Computing: Theory and Applications", Elsevier publications, North Holland Publication, 1st Edition, 2016.
2. Mallick, Pradeep Kumar, Borah, Samarjeet, "Emerging Trends and Applications in Cognitive Computing‖, IGI Global Publishers, 2019.

ARTIFICIAL INTELLIGENCE

22AM921	SOFT COMPUTING (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Learn the basic concepts of Soft Computing.
- Understand artificial neural networks.
- Explain fuzzy systems.
- Explain Genetic Algorithms.
- Discuss the various Hybrid algorithms and various Swarm Intelligence algorithms.

UNIT I	INTRODUCTION	6+6

Neural Networks - Application Scope of Neural Networks - Fuzzy Logic - Genetic Algorithm - Hybrid Systems - Soft Computing - Artificial Neural Network - Evolution of Neural Networks - Basic Models of ANN – Weights – Bias – Threshold – Learning Rate – Momentum Factor – Vigilance Parameter- McCulloch-Pitts Neuron - Linear Separability - Hebb Network.

List of Exercise/Experiments

1. Write a program to implement Hebb‘s rule.
2. Implement McCulloh-Pitts model using Simple Neural Network.

UNIT II	ARTIFICIAL NEURAL NETWORKS	6+6

Perceptron Networks - Adaptive Linear Neuron - Multiple Adaptive Linear Neurons - Back-Propagation Network - Radial Basis Function Network - Pattern Association – Auto associative and Hetero associative Memory Networks - Bidirectional Associative Memory (BAM) - Hopfield Networks - Fixed Weight Competitive Nets - Kohonen Self-Organizing Feature Maps.

List of Exercise/Experiments

1. Implement Kohonen self-Organizing feature maps
2. Write a program for solving linearly separable problem using Perceptron Model

UNIT III	FUZZY SYSTEMS	6+6

Fuzzy Logic - Classical Sets (Crisp Sets) - Fuzzy Sets – Fuzzy Relation - Features of the Membership Functions - Fuzzification - Methods of Membership Value Assignments - Defuzzification - Lambda-Cuts for Fuzzy Sets (Alpha-Cuts) - Lambda-Cuts for Fuzzy Relations - Defuzzification Methods – Fuzzy Reasoning – Fuzzy Inference Systems.

List of Exercise/Experiments

1. Implement Union, Intersection, Complement and Difference operations on fuzzy sets. Also create fuzzy relation by Cartesian product of any two fuzzy sets and perform max-min composition on any two fuzzy relations.
2. Implementation of fuzzy relations (Max-Min Composition)

UNIT IV	GENETIC ALGORITHMS	6+6

Biological Background - Traditional Optimization and Search Techniques- Genetic Algorithm and Search Space- - Simple GA - General Genetic Algorithm - Operators - Stopping Condition - Constraints - Problem Solving - The Schema Theorem- Classification - Holland Classifier Systems- Genetic Programming - Advantages and Limitations- Applications.

List of Exercise/Experiments

1. Implement travelling salesperson problem (tsp) using genetic algorithms.

2. Implement two classes city and fitness using genetic algorithm.

UNIT V	HYBRID SOFT COMPUTING AND SWARM INTELLIGENCE ALGORITHMS	6+6

Neuro-Fuzzy Hybrid Systems - Genetic Neuro-Hybrid Systems - Genetic Fuzzy Hybrid and Fuzzy Genetic Hybrid Systems - Simplified Fuzzy ARTMAP – Swarm Intelligence Algorithms - Ant Colony Optimization – Artificial Bee Colony – Particle Swarm Optimization – Firefly Algorithm.

List of Exercise/Experiments

1. To design and implement fuzzy logic for a washing machine system.
2. Case study on hybrid system. To study the designing of Neuro-Fuzzy Systems

Mini Project:

1. Apply Swarm Intelligence Algorithms for any one of the following applications:
    a. Disease diagnosis
    b. Image Processing
    c. Business Intelligence
    d. Cyber Security etc.
2. Case study on Hybrid Systems.
3. To study the designing of Neuro Fuzzy systems.
4. To design and implement fuzzy logic for a washing machine system.

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basic concepts of Soft Computing
CO2: Artificial neural networks and its applications.
CO3: Fuzzy logic and its applications.
CO4: Solving problems using Genetic algorithms.
CO5: Applications of Soft computing to solve problems in varieties of application domains.

TEXT BOOKS:

1. S. N. Sivanandam , S. N. Deepa, "Principles of Soft Computing", Wiley India Pvt. Ltd., 2nd Edition, 2019.
2. Adam Slovik, ¯Swarm Intelligence Algorithms: Modification and Applications‖, Taylor & Francis, First Edition, 2020.

REFERENCES:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.
2. Kwang H. Lee, First course on Fuzzy Theory and Applications‖, Springer, 2005.
3. N.P. Padhy, S. P. Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.
4. S. Rajasekaran, G. A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications ", PHI Learning Pvt. Ltd., 2017.
5. NPTEL Courses:
    a. Introduction To Soft Computing - https://onlinecourses.nptel.ac.in/noc23_cs40/preview

22CS936	NEURAL NETWORKS AND DEEP LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basics of deep neural networks.
- Implement deep learning models.
- Elaborate CNN and RNN architectures of deep neural networks.
- Familiarize autoencoders in neural networks.
- Learn about the deep generative models.
- Apply Deep Learning to solve real-world problems.

UNIT I	NEURAL NETWORKS	9

Introduction – Data Representation -Tensor Operations – Gradient-based Optimization – Architecture – Keras – Shallow Neural Networks.

UNIT II	DEEP NETWORKS	9

Deep feedforward networks - Learning XOR - Gradient based learning - Hidden Units – Architecture Design – Back Propagation – Regularization – Parameter Norm Penalties – Constrained Optimization - Under-Constrained Problems - Dataset Augmentation - Noise Robustness - Semi-Supervised Learning - Multi-Task Learning - Early Stopping - Parameter Tying and Sharing - Bagging and Other Ensemble methods - Dropout - Adversarial Training.

UNIT III	CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS	9

Convolution Operation - Pooling - Infinitely Strong prior - Variants - Structured Output - Data Types - Efficient Convolutional Algorithms - Random or Unsupervised features - Neuroscientific Basis - Computational Graphs - RNN - Bidirectional RNN - Encoder-Decoder - Sequence to Sequence RNN - Deep Recurrent Networks - Recursive Neural Networks - Long Term Dependencies - Leaky Units - Strategies for multiple time scales - LSTM and Gated RNNs - Optimization for Long Term Dependencies.

UNIT IV	AUTOENCODERS	9

Autoencoders: Undercomplete autoencoders - Regularized autoencoders – Power, Layer Size and Depth - Stochastic encoders and decoders – Denoising Autoencoders - Learning with autoencoders – contractive Autoencoders – Applications of autoencoders.

UNIT V	DEEP GENERATIVE MODELS	9

Boltzmann Machine – Restricted Boltzmann Machine – Deep Belief Networks – Deep Boltzmann Machines - Boltzmann Machines for Real-Valued Data – Convolutional Boltzmann Machines - Boltzmann Machine for Structured or Sequential Outputs – Directed Generative Nets – Evaluating Generative Models.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Demonstrate the basics of deep neural networks to solve real world problems.
CO2: Implement deep learning models.
CO3: Elaborate CNN and RNN architectures of deep neural networks.
CO4: Use autoencoders in neural networks.
CO5: Illustrate the various deep generative models.
CO6: Apply deep generative models to solve real world problems.

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, ¯Deep Learning‖, MIT Press, 2016.

2. Francois Chollet, ¯Deep Learning with Python‖, Manning Publications, 2018.

REFERENCES:

1. Simon J.D. Prince, ¯Understanding Deep Learning‖, MIT Press, 2023.
2. Charu C. Aggarwal, ¯Neural Networks and Deep Learning: A Textbook‖, Springer International Publishing, 2018.
3. Yoav Goldberg, ¯Neural Network Methods for Natural Language Processing‖, Synthesis Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
4. Francois Chollet, ¯Deep Learning with Python‖, Manning Publications Co, 2018.
5. Josh Patterson, Adam Gibson, ¯Deep Learning: A Practitioner's Approach‖, O'Reilly Media, 2017.
6. Navin Kumar Manaswi, ¯Deep Learning with Applications Using Python‖, Apress, 2018.
7. Richard O. Duda, Peter E. Hart, David G. Stork, ¯Pattern Classification‖, John Wiley & Sons Inc., 2007.

22CS937	REINFORCEMENT AND ENSEMBLE LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Outline the concepts of Reinforcement Learning
- Solve problems using Monte Carlo Decision Process and Dynamic Programming.
- Implement problems using temporal difference learning.
- Apply functional approximation in reinforcement learning.

UNIT I	INTRODUCITON	9

Introduction - Elements of RL, History of RL- Limitation and Scope - Examples – Multi-arm Bandits – k-armed Bandit Problem – Action-Value Methods – Incremental Implementation – Nonstationary Problem – Optimistic Initial Values – Upper Confidence Bound Action Selection – Gradient Bandit Algorithms – Contextual Bandits.

UNIT II	TABULAR SOLUTION METHODS	9

Finite Markov Decision Processes – Dynamic Programming – Monte Caro Methods – Temporal Difference Learning.

UNIT III	FUNCTION APPROXIMATION METHODS	9

On-Policy Prediction with Approximation: Value-function Approximation – The Prediction Objective – Stochastic-gradient and Semi-gradient Methods - Linear Methods – Feature Construction for Linear Methods - Eligibility Traces: The λ-return – TD(λ).

UNIT IV	CORE ENSEMBLE METHODS	9

Boosting - Boosting procedure – AdaBoost Algorithm – Examples and Issues - Bagging - Algorithm – Examples and Issues – Random tree Ensembles -Combination Methods - Averaging – Voting – Combining by learning – Other Combination methods – Relevant methods.

UNIT V	ADVANCED ENSEMBLE METHODS	9

Ensemble Pruning - Categories – Ordering based – Clustering based – Optimization based Clustering Ensembles - Categories – Similarity based – Graph based – Relabeling based – Transformation based.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Analyze the basics and history of reinforcement learning using examples like Multi-arm Bandits

CO2: Use tabular methods for solving Markov Decision Processes.

CO3: Evaluate function approximation methods for on-policy prediction.

CO4: Design and compare ensemble methods like boosting and bagging

CO5: Analyze and combine advanced ensemble methods for pruning and clustering

CO6: Apply ethical principles and communicate effectively in presenting learning methods

TEXT BOOKS:

1. Sutton R. S. and Barto A. G., "Reinforcement Learning: An Introduction", MIT Press, SecondEdition, 2020.
2. Zhi-Hua Zhou. Ensemble Methods Foundations and Algorithms , First Edition, Chapman & Hall/CRC Machine Learning & Pattern Recognition, 2012.

REFERENCES:

6. Kevin Murphy, ¯Machine Learning - A Probabilistic Perspective‖ , MIT press, 2012.
7. Christopher Bishop, ¯Pattern Recognition and Machine Learning‖, Springer, 2006.
8. Phil Winder, ¯Reinforcement Learning: Industrial Applications of Intelligent Agents¯. Oreilly, 2021.

22AM922	APPLIED AI and ML	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand and apply statistical methods to analyze and interpret data.
- Analyze and cluster genomic data using appropriate algorithms.
- Implement linear regression models to predict outcomes.
- Evaluate and improve model performance in binary classification tasks.
- Implement and train neural networks for various tasks.

UNIT I	FOUNDATION OF DATA SCIENCE	9

Python for Data Science- NumPy & Pandas - Data Cleaning and Preparation- Statistics for Data Science- Types of Data- Levels of Measurement-Descriptive Statistics-Probability theory -Inferential Statistics-Advanced Visualization Techniques.

Case Study: Cardio Good Fitness Data Analysis

Projects:1. Food Hub Analysis
2. FIFO World Cup Analysis
3. Mobile Internet Usage Analysis

UNIT II	MAKING SENSE OF UNSTRUCTURED DATA	9

Introduction to Supervised & Unsupervised Learning- Handling Imbalanced Datasets-K-Means Clustering algorithm, Dimensionality Reduction techniques (PCA, t-SNE)-Visualizing High Dimensional Data-Comparsion of t-SNE with PCA-Combining PCA with t-SNE.

Case Study: Genomic Data Clustering

Project: Fantasy Sports Clustering Analysis

UNIT III	REGRESSION AND PREDICTION	9

Introduction to Linear Regression-OLS Method-Cost function and Optimization-Gradient Descent Algorithm-Multiple Linear Regression-Elastic Net, Model Evaluation Techniques in solving Real World Regression Problems.

Case Studies: 1. Hospital LOS Prediction
2.Big Mart Sales Prediction
Project: Super Kart Sales Prediction

UNIT IV	CLASSIFICATION AND HYPOTHESIS TESTING	9

Concepts of Classification algorithms- Model Performance- Application of Binary Classification- Multi class classification-Multi label classification-Challenges in solving real world classification problems.

Case Studies: 1.HR Employee Attrition Prediction
2. KC Roasters Coffee Quality Prediction
Projects: 1. Travel Package Purchase Prediction
2. Potential Customers Prediction

UNIT V	DEEP LEARNING	9

Implementation of Neural Networks-Data Quality & Quantity-Data Augmentation- Hyper parameter tuning-Computational Challenges -Transformer Networks-Transfer learning - solving real world Neural Network based Problems.

Case Study: 1. Audio MNLST Digit Recognition,
2.Street View Housing Number Digit Recognition
Project: Food Image Classification

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Apply statistical techniques to interpret data and make data-driven decisions.
CO2: Utilize dimensionality reduction techniques such as PCA and t-SNE to simplify complex datasets.
CO3: Apply regression techniques to real-world problems.
CO4: Perform hypothesis testing to validate assumptions and make inferences from data.
CO5: Apply deep learning techniques to solve practical problems.
CO6: Implement the concepts of AI and ML to solve various applications.

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, Machine Learning, Pearson, 2019.
2. Ethem Alpaydin, Introduction to Machine Learning, Adaptive Computation and Machine Learning Series, Third Edition, MIT Press, 2014.
3. Deep Learning: A Practitioner's Approach, Josh Patterson, Adam Gibson, O'Reilly Media, 2017.
4. Deep Learning, Ian Goodfellow, Yoshua Bengio Aaron Courville, MIT Press, 2017.
5. Neural Networks and Deep Learning, Michael Nielsen, Determination Press, 2015.

REFERENCES:

1. Anuradha Srinivasaraghavan, Vincy Joseph, Machine Learning, First Edition, Wiley, 2019.
2. Peter Harrington, ―Machine Learning in Action‖, Manning Publications, 2012.
3. Stephen Marsland, ―Machine Learning – An Algorithmic Perspective‖, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
4. Tom M Mitchell, Machine Learning, First Edition, McGraw Hill Education, 2013.
5. Christoph Molnar, ―Interpretable Machine Learning - A Guide for Making Black Box Models Explainable‖, Creative Commons License, 2020.
6. Deep Learning with TensorFlow: Explore neural networks with Python, Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, Packt Publisher, 2017.
7. Deep Learning with Keras, Antonio Gulli, SujitPal , Packt Publishers, 2017.
8. Deep Learning with Python", Francois Chollet, Manning Publications, 2017
9. https://olympus.mygreatlearning.com/courses

22AM923	RECOMMENDER SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the foundations of the recommender system.
- To learn about collaborative filtering.
- To discuss content-based recommendation systems.
- To elaborate on the evaluation paradigms for a recommendation system.
- To make students design and implement a recommender system.

UNIT I	INTRODUCTION TO RECOMMENDER SYSTEMS	9

Introduction - Basic Models of Recommender Systems - Domain-Specific Challenges in Recommender Systems - Cold-Start Problem – Attack-Resistant Recommender Systems – Group – Multi-Criteria – Active-Learning – Privacy - Application Domains.

UNIT II	COLLABORATIVE FILTERING	9

Neighborhood-Based Collaborative Filtering - Key Properties - Predicting Ratings – Clustering -Dimensionality Reduction - A Regression Modeling - Graph Models – Model-based Collaborative Filtering - Decision and Regression Trees - Rule-Based Collaborative Filtering - Naive Bayes Collaborative Filtering – Latent Factor Models.

UNIT III	CONTENT-BASED RECOMMENDATION	9

Basic Components of Content-Based Systems - Preprocessing and Feature Extraction - Learning User Profiles and Filtering - Content-Based Versus Collaborative Recommendations - Using Content-Based Models for Collaborative Filtering.

UNIT IV	DESIGN EVALUATION	9

Evaluating Paradigms – General Goals of Evaluation Design-Design Issues in Offline Recommender Evaluation-Accuracy Metrics in Offline Evaluation-Limitations of Evaluation Measures.

UNIT V	TYPES OF RECOMMENDATION SYSTEMS	9

Content-based Recommender Systems – Basic Components – Constraint-based Recommender Systems – Context-sensitive Recommender Systems – Social and Trust-Centric Recommender Systems.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Elaborate the foundations of the recommender system.
CO2: Use collaborative filtering to design recommendation systems.
CO3: Discuss content-based recommendation systems.
CO4: Elaborate on the evaluation paradigms for a recommendation system.
CO5: Use appropriate type of recommendation systems to solve real-world problems.
CO6: Design, implement and evaluate a recommendation algorithm.

TEXT BOOKS:

1. Charu C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.
2. Jannach D., Zanker M., FelFering A., Friedrich G., Recommender Systems: An Introduction, Cambridge University Press, First Edition, 2011.

REFERENCES:

1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Mining of massive datasets, 3rd edition, Cambridge University Press, 2020.
2. Ricci, F., Rokach, L. and Shapira, B., Introduction to recommender systems handbook. In Recommender systems handbook, Springer, 2011.
3. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer, First Edition, 2013.

22CS920	GENERATIVE AI FUNDAMENTALS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basic concepts of Generative AI.
- Build Generative AI systems to generate images.
- Understand the concept used in Generative AI Models.
- Use various Generative AI models.
- Compare and use the various Large Language Models.
- Understand the basics of Prompt Engineering.

UNIT I	INTRODUCTION	9

Generative Models – Image transformation – Challenges - Deep Neural Networks – Perceptron – back propagation – CNN – RNN – Optimizer.

UNIT II	IMAGE GENERATION	9

Creating encodings of images – variational objective – Inverse Autoregressive flow – Importing CIFAR – Creating the network from TensorFlow 2.

UNIT III	GENERATIVE ADVERSARIAL NETWORKS	9

Generative Adversarial Networks – Vanilla GAN – Improved GANs – Progressive GAN – Challenges – Paired style transfer – Unpaired style transfer – Deepfakes – Modes of operation – key feature set – High level flow – Replacement – Re-enactment.

UNIT IV	LARGE LANGUAGE MODELS	9

Overview of LLMs - Transformers – GPT – Types of LLMs – Key concepts – other Transformers - T5 – Generative Pre-Training Models – Multi-modal Models – DALL.E 2

UNIT V	PROMPT ENGINEERING	9

Basics – In-Context Learning – In-Context Prompting – Techniques – Image Prompting – Prompt Hijacking – Challenges.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Elaborate the basic concepts of Generative AI.
CO2: Build Generative AI systems to generate images.
CO3: Apply the concepts used in Generative AI Models.
CO4: Use various Generative AI models.
CO5: Compare and use the various Large Language Models.
CO6: Analyze the basics of Prompt Engineering.

TEXT BOOKS:

3. Ben Auffarth, Generative AI with LangChain, Packt Publishing, 2023.
4. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.

REFERENCES:

4. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023.

5. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024.
6. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.

EMERGING TECHNOLOGIES

22CS921	INDUSTRIAL IoT	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basic technologies and protocols used in Industrial IoT.
- Illustrate the models and architectures of IIoT.
- Identify and apply different sensors for various IIoT applications.
- Explain the various protocols used in IIoT.
- Build solutions for real-world problems using IIoT.
- Solve real-world problems using IIoT analytics.

UNIT I	INTERNET OF THINGS (IoT)	9

Introduction – Networking – Cyber Physical Systems – Evolution of IoT – IoT Networking Components - Addressing Strategies – IoT Sensing and Actuation.

UNIT II	INDUSTRIAL IoT	9

Industry 4.0 – IIoT – Industrial Internet Systems – Industrial Sensing – Industrial Processes – Business Models and Reference Architecture.

UNIT III	SENSORS AND ACTUATORS	9

Sensors – Sensor Characteristics – Sensorial Deviations – Sensing Types - Considerations – Actuators – Actuator Types – Actuator Characteristics.

UNIT IV	PROTOCOLS	9

Processing topologies and types – Connectivity Technologies – IEEE 802.15.4 – Zigbee – RFID – LoRa – Wi-Fi – Communication Technologies – Constrained nodes – Networks – Infrastructure Protocols - IPV6 – Discovery Protocols – MQTT – MQTT-SN – SOAP - REST.

UNIT V	IIOT ANALYTICS AND APPLICATIONS	9

IIoT Analytics – Categorization – Use – Challenges – Mapping of analytics with IIRA Architecture – Deployment of Analytics – Health care applications in industries - Inventory Management and Quality Control – Plant Safety and Security.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Elaborate the basic technologies and protocols used in Industrial IoT.
CO2: Illustrate the models and architectures of IIoT.
CO3: Interpret and apply different sensors for various IIoT applications.
CO4: Explain the various protocols used in IIoT.
CO5: Build solutions for real-world problems using IIoT.
CO6: Solve real-world problems using IIoT analytics.

TEXT BOOKS:

1. S. Misra, A. Mukherjee, and A. Roy, Introduction to IoT. Cambridge University Press, 2020.
2. S. Misra, C. Roy, and A. Mukherjee, Introduction to Industrial Internet of Things and Industry 4.0. CRC Press, 2020.

REFERENCES:

1. Daniel Minoli, Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications,1st Edition, Wiley Publications, 2013.
2. Dieter Uckelmann , Mark Harrison, Florian Michahelles, Architecting the Internet of Things, Industry 4.0: The Industrial Internet of Things, Springer-Verlag Berlin Heidelberg, 2011.
3. Arshdeep Bahga, Vijay Madisetti, ¯Internet of Things – A hands-on approach‖, Universities Press, 2015.
4. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton, Jerome Henry, ¯IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things‖, CISCO Press, 2017.
5. https://onlinecourses.nptel.ac.in/noc20_cs69/preview

22AM912	GPU COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the basics of GPU Architectures and CUDA Programming.
- To learn synchronization using CUDA.
- To discuss memories and its impact on performance.
- To understand the various parallel algorithms on GPU.
- To learn the basics of OPENCL.

UNIT I	GPU ARCHITECTURES AND CUDA PROGRAMMING	9

Heterogeneous Parallel Computing – Architecture of a modern GPU – Parallel Programming languages and models – GPU Computing – Introduction to Data Parallelism and CUDA C: Data Parallelism – CUDA Program Structure – A vector additional Kernel – Device Global Memory and Data Transfer – Kernel functions and Threading.

UNIT II	MULTI-DIMENSIONAL DATA & SYNCHRONIZATION	9

CUDA Thread Organization - Mapping Threads to Multi-Dimensional Data – Synchronization and Transparent Scalability – Assigning resources to Blocks – Querying Device Properties – Thread Scheduling and Latency Tolerance.

UNIT III	CUDA MEMORIES & PERFORMANCE	9

CUDA Memories – Memory Access Efficiency – CUDA Device Memory Types – Reducing global Memory Traffic – Performance Considerations - Warps and Thread Execution – Global Memory Bandwidth – Dynamic Partitioning of Execution Resources – Instruction Mix and Thread Granularity.

UNIT IV	ALGORITHMS ON GPU	9

Parallel Patterns: Convolution – Prefix Sum – Sparse Matrix – Vector Multiplication.

UNIT V	OPENCL BASICS	9

Introduction – OpenCL Platform Model – Execution Model – Programming model – Memory Model – OpenCL Runtime.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of GPU Architectures and implement simple CUDA Programs.
CO2: Discuss synchronization using CUDA.
CO3: Elaborate CUDA memories and its impact on performance.

CO4: Design various parallel algorithms on GPU.
CO5: Solve simple problems using parallel algorithms.
CO6: Apply OpenCL to solve programs and improve performance.

TEXT BOOKS:

1. David Kirk and Wen-mei Hwu, Programming Massively Parallel Processors – A hands-on Approach, Morgan Kaufmann, Second Edition, 2013.
2. Benedict Gaster,Lee Howes, David R. Kaeli , ¯Heterogeneous Computing with OpenCL‖, Third Edition, Morgan Kaufman, 2012.

REFERENCES:

1. David Kaeli, Perhaad Mistry, Dana Schaa, Dong Ping Zhang, ¯Heterogeneous Computing with OpenCL 2.0‖, Third Edition, Morgan Kaufman, 2015.
2. John L.Hennessy and David A. Patterson, ¯Computer Architecture - A Quantitative Approach‖, Sixth Edition, Morgan Kaufman, 2017.
3. NPTEL Courses:
   a. GPU Architectures And Programming - https://onlinecourses.nptel.ac.in/noc23_cs61/preview

22CS922	INTRODUCTION TO AUGMENTED AND VIRTUAL REALITY	L	T	P	C
		3	0	0	3

OBJECTIVES:
The Course will enable learners to:

- Get exposure on Augmented Reality.
- Introduce Virtual Reality and input and output devices.
- Acquire knowledge on computing architectures and modelling.
- Explore Virtual Reality programming and human factors.
- Learn various applications of Virtual Reality.

UNIT I	AUGMENTED REALITY	9

Introduction to Augmented Reality-Computer vision for AR-Interaction-Modelling and Annotation-Navigation-Wearable devices.

UNIT II	INTRODUCTION TO VIRTUAL REALITY AND INPUT AND OUTPUT DEVICES	9

Introduction: The three I‘s of Virtual Reality - - Early commercial VR technology - The five classic components of a VR system. Input devices: Three-Dimensional position trackers - tracker performance parameters - ultrasonic trackers - optical trackers - Navigation and manipulation interfaces - gesture interfaces. Output devices: graphics displays - large-volume displays - sound displays.

UNIT III	COMPUTING ARCHITECTURES AND MODELING OF A VR SYSTEM	9

Computing architectures for VR: The rendering pipeline - The graphics rendering pipeline - The haptics rendering pipeline - PC graphics architecture - PC graphics accelerators - Graphics benchmarks - Distributed VR architectures - Multipipeline synchronization - Colocated rendering pipelines. Modeling: geometric modeling - kinematics modeling - physical and behavior modelling

UNIT IV	VR PROGRAMMING AND HUMAN FACTORS	9

Toolkits and scene graphs - World Toolkit - Model geometry and appearance - The WTK scene graph - Sensors and action functions - WTK networking - Java 3D - Model geometry and

appearance - Java 3D scene graph - Sensors and behaviors - Java 3D networking - WTK and Java 3D performance comparison -Human factors in VR: Methodology and terminology - user performance studies - VR health and safety issues - VR and society		
UNIT V	**APPLICATIONS OF VR**	**9**
Medical Application of VR - Virtual anatomy-Triage and diagnostic - Surgery - VR in education - VR and the Arts - Entertainment applications of VR - military VR applications - Army use of VR - VR applications in the Navy - Air force use of VR - Applications of VR in Robotics - Robot programming - Robot teleoperation		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand Augmented Reality.
CO2: Explore different input and output devices used in Virtual Reality system.
CO3: Model the VR system.
CO4: Learn about Google Toolkit‘s and Scene Graph.
CO5: Apply VR in various fields.
CO6: Use modern tools to design Applications.

TEXT BOOKS:

1. Dieter Schmalstieg, Tobias Hollerer, ―Augmented Reality: Principles & Practice‖, Addison Wesley, 2016.
2. Grigore C. Burdea, Philippe Coiffet, ―Virtual reality technology‖, Wiley, Second Edition, 2017.

REFERENCE BOOKS:

1. Sherman, William R & Craig, Alan B, ―Understanding Virtual reality‖, Elsevier India Private Limited, Noida, 2018.
2. Charles Palmer, John Williamson, ―Virtual Reality Blueprints: Create compelling VR experiences for mobile‖, Packt Publisher, 2018.

SOFTWARE REQUIREMENTS:

Unity, Maya/3DS MAX/Blender.

22CS923	**DIGITAL MARKETING**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

OBJECTIVES:

The Course will enable learners to:

- Examine and explore the role and importance of digital marketing in today‘s rapidly changing business environment.
- Focus on how digital marketing can be utilized by organizations and how its effectiveness can be measured.

UNIT I	**INTRODUCTION TO ONLINE MARKET**	**9**
Online Market space- Digital Marketing Strategy- Components - Opportunities for building Brand Website - Planning and Creation - Content Marketing		
UNIT II	**SEARCH ENGINE OPTIMIZATION**	**9**
Search Engine optimization - Keyword Strategy- SEO Strategy - SEO success factors -On-Page Techniques - Off-Page Techniques. Search Engine Marketing- How Search Engine works- SEM components- PPC advertising -Display Advertisement		
UNIT III	**E- MAIL MARKETING**	**9**

E- Mail Marketing - Types of E- Mail Marketing - Email Automation - Lead Generation - Integrating Email with Social Media and Mobile- Measuring and maximizing email campaign effectiveness. Mobile Marketing- Mobile Inventory/channels- Location based; Context based; Coupons and offers, Mobile Apps, Mobile Commerce, SMS Campaigns- Profiling and target in

UNIT IV	SOCIAL MEDIA MARKETING	9

Social Media Marketing - Social Media Channels- Leveraging social media for brand conversations and buzz. Successful /benchmark social media campaigns. Engagement Marketing- Building Customer relationships - Creating Loyalty drivers - Influencer Marketing.

UNIT V	DIGITAL TRANSFORMATION	9

Digital Transformation & Channel Attribution – Analytics - Ad-words, Email, Mobile, Social Media, Web Analytics - Changing your strategy based on analysis- Recent trends in Digital marketing.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Examine and explore the role and importance of digital marketing in today‘s rapidly changing business environment.

CO2: Focuses on how digital marketing can be utilized by organizations and how its effectiveness can be measured.

CO3: Understand the key elements of a digital marketing strategy

CO4: Study how the effectiveness of a digital marketing campaign can be measured

CO5: Demonstrate advanced practical skills in common digital marketing tools such as SEO, SEM, social media and Blogs

CO6: Understand the recent trends in digital marketing.

TEXT BOOKS:

1. Puneet Singh Bhatia, ¯Fundamentals of Digital Marketing‖, Pearson Education; First edition, 2017.
2. Vandana Ahuja, ¯Digital Marketing‖, Oxford University Press, 2015

REFERENCE BOOKS:

1. Philip Kotler,‖ Marketing 4.0: Moving from Traditional to Digital‖, Wiley; 1st edition, 2017
2. Ryan, D., ¯Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation‖, Kogan Page Limited., 2014
3. Barker, Barker, Bormann and Neher, Social Media Marketing: A Strategic Approach, 2E South-Western ,Cengage Learning. 2017.
4. Pulizzi,J, ¯Beginner's Guide to Digital Marketing‖, Mcgraw Hill Education, 2019.

SOFTWARE REQUIREMENTS:

Unity, Maya/3DS MAX/Blender.

22CS924	QUANTUM COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Analyse the behaviour of basic quantum algorithms
- To discuss simple quantum algorithms and information channels in the quantum circuit model

- Apply the Quantum Algorithms in Superdense coding and quantum Teleportation
- Analyse the Algorithms with Superpolynomial Speed-up
- Illustrate a simple quantum error-correcting code

UNIT I	FOUNDATION	9

Overview of traditional computing – Church-Turing thesis – circuit model of computation – reversible computation – quantum physics – quantum physics and computation – Dirac notation and Hilbert Spaces – dual vectors – operators – the spectral theorem – functions of operators – tensor products – Schmidt decomposition theorem

UNIT II	QUBITS AND QUANTUM MODEL OF COMPUTATION	9

State of a quantum system – time evolution of a closed system – composite systems – measurement – mixed states and general quantum operations – quantum circuit model – quantum gates – universal sets of quantum gates – unitary transformations – quantum circuits

UNIT III	QUANTUM ALGORITHMS – I	9

Superdense coding – quantum teleportation – applications of teleportation – probabilistic versus quantum algorithms – phase kick-back – the Deutsch algorithm – the Deutsch-Jozsa algorithm – Simon's algorithm – Quantum phase estimation and quantum Fourier Transform – eigenvalue estimation

UNIT IV	QUANTUM ALGORITHMS – II	9

Order-finding problem – eigenvalue estimation approach to order finding – Shor's algorithm for order finding – finding discrete logarithms – hidden subgroups – Grover's quantum search algorithm – amplitude amplification – quantum amplitude estimation – quantum counting – searching without knowing the success probability 101

UNIT V	QUANTUM COMPUTATIONAL COMPLEXITY AND ERROR CORRECTION	9

Computational complexity – black-box model – lower bounds for searching – general black-box lower bounds – polynomial method – block sensitivity – adversary methods – classical error correction – classical three-bit code – fault tolerance – quantum error correction – three- and nine-qubit quantum codes – fault-tolerant quantum computation

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Analyse the behaviour of basic quantum algorithms
CO2: Discuss simple quantum algorithms and information channels in the quantum circuit model
CO3: Apply the Quantum Algorithms in Superdense coding and quantum Teleportation
CO4: Analyse the Algorithms with Super polynomial Speed-up
CO5: Illustrate a simple quantum error-correcting code
CO6: Elaborate various quantum algorithms.

TEXT BOOKS:

1. P. Kaye, R. Laflamme, and M. Mosca, ¯An introduction to Quantum Computing‖, Oxford University Press, 2007.

REFERENCES:

1. E. Rieffel and W. Polak ¯Quantum Computing A Gentle Introduction‖, The MIT Press Cambridge, 2011.
2. Jack D. Hidary ¯Quantum Computing: An Applied Approach‖ Springer, 2019.
3. V. Sahni, ¯Quantum Computing‖, Tata McGraw-Hill Publishing Company, 2007.
4. Michael A. Nielsen and Issac L. Chuang, ¯Quantum Computation and Quantum Information‖, Tenth Edition, Cambridge University Press, 2010

22AM909	INTELLIGENT ROBOTS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the basics of Intelligent Robots.
- To discuss the Autonomous capabilities and Software architecture.
- To elaborate the Reactive Functionality of intelligent Robots.
- To use the various sensors in building Intelligent Robots.
- To illustrate the Deliberative Functionality of intelligent Robots.

UNIT I	INTRODUCTION	9

Overview- Definition – Components -Three Modalities – Need for Intelligent Robots – History of AI Robotics – Industrial Manipulators – Mobile Robots – Drones – Cognitive Systems.

UNIT II	AUTOMATION AND AUTONOMY	9

Autonomous Capabilities – Bounded Rationality – Automation and Autonomy – Programming Style – Hardware Design – Types of Functional Failures – Autonomous Capabilities. Types of Software Architectures – Operational Architectures – Components of a Telesystem – Human Supervisory Control.

UNIT III	REACTIVE FUNCTIONALITY	9

Behaviours : Agency and Marr‘s Computational Theory – Animal Behaviours – Schema Theory. Perception: Action-Perception cycle – Functions. Behaviour Coordination – Function – Cooperating Methods – Competing Methods – Sequences.

UNIT IV	SENSORS AND SENSING	9

Locomotion: Mechanical, Biomimetic, Legged Locomotion – Action Selection – Sensors and Sensing Model – Choosing – Range Sensing: Stereo – Depth from X – Sonar or Ultrasonics.

UNIT V	DELIBERATIVE FUNCTIONALITY	9

Deliberation – Strips – Navigation – Spatial Memory – Types of Path Planning – Configuration Space – Metric Path Planning – Motion Planning – Localization – Feature based Localization – Iconic Localization – Static vs Dynamic Environments – Simultaneous Localization and Mapping - Terrain Identification and Mapping – Scale and Traversability - Exploration – Mutlirobot Systems and AI – Human-Robot Interaction and areas of AI.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of Intelligent Robots.
CO2: Design and implement Autonomous capabilities in Robotics systems.
CO3: Elaborate the Reactive Functionality of intelligent Robots.
CO4: Use the various sensors in building Intelligent Robots.
CO5: Illustrate the Deliberative Functionality of intelligent Robots.
CO6: Analyse the various applications of AI Robotics.

TEXT BOOKS:

1. Robin R. Murphy, “Introduction to AI Robotics”, MIT Press, Second Edition, 2019.

REFERENCES:

1. Francis X. Govers, "Artificial Intelligence for Robotics: Build Intelligent Robots that Perform Human Tasks Using AI Techniques", Packt Publishing, 2018.
2. Sebastian Thrun, Wolfram Burgard, and Dieter Fox, "Probabilistic Robotics", MIT Press, 2005.
3. Yoon Seok Pyo, Han Cheol Cho, Ryu Woon Jung, and Tae Hoon Lim, "ROS Robot Programming", ROBOTIS Co., Ltd, 2017.

22CS925	GAME DEVELOPMENT	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand game programming fundamentals.
- To learn about the processes, mechanics, issues in game design.
- To gain knowledge of the game design and Artificial intelligence.
- To understand the design and scripting languages of game programming.
- To know about networked games and analyse code for sample games.

UNIT I	INTRODUCTION	9

Evolution of video game programming-The Game Loop-Time and games-Game objects-2D rendering Foundations-Sprites-Scrolling-Tile maps-Vectors -Matrices.

UNIT II	3D GRAPHICS FOR GAMES	9

3D graphics-Basics-Coordinate-spaces-Lighting and Shading-visibility-Input Devices-Event based input system-Mobile Input-Basic sound-3D sound-Digital Signal Processing-Physics-Planes, Rays, and line segments-Collision geometry-Collision detection-Physics base movement-Physics middleware.

UNIT III	GAME DESIGN AND AI	9

Cameras-Types of cameras-Perspective projection-Camera implementation-Camera support algorithm- Real AI versus Game AI-Pathfinding-State based behaviours-Strategy and planning.

UNIT IV	USER INTERFACE AND SCRIPTING LANGUAGES	9

Menu system-HUD elements-Radar-other UI considerations-Scripting languages-Implementing a scripting language-Tokenization-Syntax Analysis-Code Execution or Generation-Data Formats-Case study UI mods in world of warcraft.

UNIT V	NETWORKED GAMES	9

Protocols-Network Topology-Server/Client-Peer-to-Peer-Cheating-Sample game -Side scroller for iOS, Tower defence for PC/Mac-Code Analysis.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the fundamentals of game programming.

CO2: Identify the processes, mechanics, issues in game design,

CO3: Analyse the game design and artificial intelligence.

CO4: Construct a basic game engine using UI and scripting languages.
CO5: Develop code for sample games.
CO6: Understand the 3D game design

TEXT BOOKS:

1. Sanjay Madhav, Game Programming Algorithms and Techniques:A platform -Agnostic Approach-Game Design,1st Edition, Addison-Wesley Professional,2013.
2. Jouni Smed, Harri Hakonen, Algorithms and Networking for Computer Games, 2nd Edition,Wiley Publications,2017.

REFERENCES:

1. Ernest Adams and Andrew Rollings, ¯Fundamentals of Game Design‖, Prentice Hall 3rd Edition,2014.
2. JungHyun Han, ¯3D Graphics for Game Programming‖, Chapman and Hall/CRC, 1st Edition, 2011.

22CS926	PRINCIPLES OF 3D PRINTING AND DESIGN	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To learn the basics of 3D Modelling and viewing.
- To demonstrate the principles of 3D processing technique and application.
- To design and illustrate the working principle of inkjet technology.
- To acquire the knowledge of laser technology and Printing machines.
- To apply the knowledge in manufacturing, healthcare and medical applications.

UNIT I	INTRODUCTION	9
Introduction; Design considerations – Material, Size, Resolution, Process; Modelling and viewing - 3D; Scanning; Model preparation – Digital; Slicing; Software; File formats.		
UNIT II	**PRINCIPLE**	**9**
Processes – Extrusion, Wire, Granular, Lamination, Photo polymerisation; Materials - Paper, Plastics, Metals, Ceramics, Glass, Wood, Fiber, Sand, Biological Tissues, Hydrogels, Graphene; Material Selection - Processes, applications, limitations.		
UNIT III	**INKJET TECHNOLOGY**	**9**
Printer - Working Principle, Positioning System, Print head, Print bed, Frames, Motion control; Print head Considerations – Continuous Inkjet, Thermal Inkjet, Piezoelectric Drop-On-Demand; Material Formulation for jetting; Liquid based fabrication – Continous jet, Mulitjet; Powder based fabrication – Colourjet.		
UNIT IV	**LASER TECHNOLOGY**	**9**
Light Sources – Types, Characteristics; Optics – Deflection, Modulation; Material feeding and flow – Liquid, powder; Printing machines – Types, Working Principle, Build Platform, Print bed Movement, Support structures.		
UNIT V	**INDUSTRIAL APPLICATIONS**	**9**

Product Models, manufacturing – Printed electronics, Biopolymers, Packaging, Healthcare, Food, Medical, Biotechnology, Displays; Future trends;

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Outline and examine the basic concepts of 3D printing technology.
CO2: Develop 3D printing workflow.
CO3: Examine and categorize the concepts and working principles of 3D printing using inkjet technique.
CO4: Understand the inkjet technology and its applications.
CO5: Describe the working principles of 3D printing using laser technique.
CO6: Apply various method for designing and modeling for industrial applications.

TEXT BOOKS:

1. Christopher Barnatt, 3D Printing: The Next Industrial Revolution, CreateSpace Independent Publishing Platform, 2013.
2. Ian M. Hutchings, Graham D. Martin, Inkjet Technology for Digital Fabrication, John Wiley & Sons, 2013.

REFERENCES:

1. Chua, C.K., Leong K.F. and Lim C.S., Rapid prototyping: Principles and applications, second edition, World Scientific Publishers, 2010.
2. Ibrahim Zeid, Mastering CAD CAM Tata McGraw-Hill Publishing Co., 2007.
3. Joan Horvath, Mastering 3D Printing, APress, 2014.

22AM702	COMPUTER VISION	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamental concepts related to Image formation and processing.
- To learn feature detection, matching and detection.
- To become familiar with feature based alignment and motion estimation.
- To develop skills on 3D reconstruction.
- To understand image based rendering and recognition.

UNIT I	INTRODUCTION TO IMAGE FORMATION AND PROCESSING	9

Computer Vision - Geometric primitives and transformations - Photometric image formation - The digital camera - Point operators - Linear filtering - More neighborhood operators - Fourier transforms - Pyramids and wavelets - Geometric transformations - Global optimization.

UNIT II	FEATURE DETECTION, MATCHING AND SEGMENTATION	9

Points and patches - Edges - Lines - Segmentation - Active contours - Split and merge - Mean shift and mode finding - Normalized cuts - Graph cuts and energy-based methods.

UNIT III	FEATURE-BASED ALIGNMENT & MOTION ESTIMATION	9

2D and 3D feature-based alignment - Pose estimation - Geometric intrinsic calibration - Triangulation - Two-frame structure from motion - Factorization - Bundle adjustment - Constrained structure and motion - Translational alignment - Parametric motion - Spline-based motion - Optical flow - Layered motion.		
UNIT IV	**3D RECONSTRUCTION**	**9**
Shape from X - Active range finding - Surface representations - Point-based representations- Volumetric representations - Model-based reconstruction - Recovering texture maps and albedosos		
UNIT V	**IMAGE-BASED RENDERING AND RECOGNITION**	**9**
View interpolation Layered depth images - Light fields and Lumi graphs - Environment mattes - Video-based rendering-Object detection - Face recognition - Instance recognition - Category recognition - Context and scene understanding- Recognition databases and test sets.		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Analyze and apply basic image processing techniques in practical applications.
CO2: Compare the concepts related to feature detection, matching and detection.
CO3: Implement feature-based alignment and motion estimation in real-world applications.
CO4: Create and Apply 3D Reconstruction techniques in diverse applications.
CO5: Perform image-based rendering and recognition.
CO6: Implement efficient solutions to image processing and computer vision problems.

TEXT BOOKS:

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011.
2. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Person Education, Second Edition, 2015

REFERENCES:

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
2. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006
3. E. R. Davies, Computer and Machine Vision, Fourth Edition, Academic Press, 2012.

OPEN ELECTIVE

22CS001	ETHICAL HACKING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand Information Security, Cyber threats, attacks, web security.

- Know about different modes of hacking tools and phases of penetration tests and Methodologies.
- Gain the knowledge of the use and availability of tools to support an ethical hack.
- Gain the knowledge of interpreting the results of a controlled attack.

UNIT I	FUNDAMENTALS OF ETHICAL HACKING	9

Overview of Cyber threats – Data and Network Security Attacks – Threats: MAC spoofing – Access control Network protocol and services-Hacking terms - Ethical Hacking overview –Modes of Ethical Hacking – Ethics and Legality.

UNIT II	HACKING METHODOLOGY RECONNAISSANCE	9

Foot printing: Reconnaissance - Footprinting theory – Penetration test – Phases of Penetration test - Methods of Footprinting – Network Information gathering process – Terminologies of Foot printing –Footprinting through search engine directives – Whois tool –NetCraft – Extract Information from DNS - Foot printing from Email servers – Shodan – Dig – MetaGooFil – Social Engineering.

UNIT III	SCANNING AND ENUMERATION	9

Scanning: Concept of Nmap - - Port scanning with Nmap – Subnet - Scanning IPs with Nmap Pings and Ping sweeps – Port - Three way handshake – NmapSyn scanning – Nmap TCP Scan – Nmap UDP Scan - Bypass of IPS and IDS – Nmap Script Engine Enumeration: Service Fingerprinting – Vulnerability Scanners – Basic Banner Grabbing – Common Network services – SMTP – DNS – RPCBIND Enumeration – SMB – NetBIOS

UNIT IV	SYSTEM AND NETWORK VULNERABILITY	9

Metasploit – Penetration testing with framework Metasploit – Scan services to identify vulnerabilities – Scan FTP services – Scan HTTP services – Exploitation – Post exploitation techniques – Meterpreter – Rootkit – Backdoor – Password hashes – Privilege Escalation - Scanning vulnerable services with Nessus

UNIT V	SOFTWARE VULNERABILITY (OWASP 10)	9

Fundamentals of OWASP Zed Attack Proxy (ZAP) – Web app vulnerability scan - Code Injection Attacks – Broken Authentication – Sensitive Data Exposure – XML External Entities – Broken Access Control – Security misconfiguration – Website pen testing - Cross Site Scripting (XSS) – Insecure Deserialization – Using Components with known vulnerabilities – Insufficient logging and monitoring.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the basics of information security, threats and its attacks

CO2: Understand the fundamentals of ethical hacking with the hacking methodologies

CO3: Analyze the phases of the penetration test with the methods

CO4: Understand the vulnerabilities and use the frameworks to identify vulnerabilities by service scan

CO5: Understand the web security issues with the fundamentals of OWASP

TEXTBOOKS:

1. McClure, S., Scambray, J. and Kurtz, G., 2012. Hacking Exposed Network Security Secrets and Solutions. New York: McGraw-Hill.
2. Engebretson, P., 2013. The Basics Of Hacking And Penetration Testing. Amsterdam: Syngress, an imprint of Elsevier.

REFERENCES:

1. Zaid Sabih, Learn Ethical Hacking from Scratch, 2018, PACKT publishing, ISBN: 978-1-78862-205-9
2. Harsh Bothra, Hacking be a hacker with ethics, Khanna Publishing, 2016, ISBN: 978-03-86173-05-8

LIST OF SOFTWARE:

22CS002	CLOUD FOUNDATIONS	L	T		C
		3	0		3

OBJECTIVES:

- ✓ To describe the different ways a user can interact with Cloud.
- ✓ To discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.
- ✓ To confer the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.
- ✓ To demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.
- ✓ To determine a variety of managed big data services in the cloud.

UNIT I	INTRODUCTION TO CLOUD	9

Cloud Computing - Cloud Versus Traditional Architecture - IaaS, PaaS, and SaaS - Cloud Architecture - The GCP Console - Understanding projects - Billing in GCP - Install and configure Cloud SDK - Use Cloud Shell - APIs - Cloud Console Mobile App.

UNIT II	COMPUTE AND STORAGE	9

Compute options in the cloud - Exploring IaaS with Compute Engine - Configuring elastic apps with autoscaling - Exploring PaaS - Event driven programs - Containerizing and orchestrating apps - Storage options in the cloud - Structured and unstructured storage in the cloud - Unstructured storage using Cloud Storage - SQL managed services - NoSQL managed services

UNIT III	APIs AND SECURITY IN THE CLOUD	9

The purpose of APIs – API Services - Managed message services - Introduction to security in the cloud - The shared security model - Encryption options - Authentication and authorization with Cloud IAM - Identify Best Practices for Authorization using Cloud IAM.

UNIT IV	NETWORKING, AUTOMATION AND MANGAEMENT TOOLS	9

Introduction to networking in the cloud - Defining a Virtual Private Cloud - Public and private IP address basics - Cloud network architecture - Routes and firewall rules in the cloud - Multiple VPC networks - Building hybrid clouds using VPNs - Different options for load balancing - Introduction to Infrastructure as Code - Terraform - Monitoring and management tools.

UNIT V	BIG DATA AND MACHINE LEARNING SERVICES	9

Introduction to big data managed services in the cloud - Leverage big data operations - Build Extract, Transform, and Load pipelines - Enterprise Data Warehouse Services -

Introduction to machine learning in the cloud - Building bespoke machine learning models with AI Platform - Pre-trained machine learning APIs.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Describe the different ways a user can interact with Cloud.

CO2: Discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.

CO3: Discuss the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.

CO4: Demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.

CO5: Discover a variety of managed big data services in the cloud.

REFERENCES:

1. https://cloud.google.com/docs
2. https://www.cloudskillsboost.google/paths/36
3. https://nptel.ac.in/courses/106105223
4. Anthony J. Sequeira, ¯AWS Certified Cloud Practitioner (CLF-C01) Cert Guide‖, First Edition, Pearson Education, 2020.
5. AWS Documentation (amazon.com)
6. AWS Skill Builder
7. AWS Academy Cloud Foundations Course - https://www.awsacademy.com/vforcesite/LMS_Login

LIST OF EQUIPMENTS:

22CS003	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand block chain system‘s fundamental components, how they fit together and examine a decentralization using block chain.
- To explain how Crypto currency works.
- To explain the components of Ethereum and Programming Languages for Ethere
- To study the basics of Web3 and Hyper ledger.
- To give an insight of alternative block chains and its emerging trends.

UNIT I	INTRODUCTION TO BLOCKCHAIN	9

History of Blockchain – Types of Blockchain – Consensus – Decentralization using Blockchain – Blockchain and Full Ecosystem Decentralization – Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography -

public and private keys – Elliptic curve cryptography – Discrete logarithm problem in ECC.		
UNIT II	**INTRODUCTION TO CRYPTOCURRENCY**	**9**
Bitcoin – Digital Keys and Addresses – Transactions – Mining – Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin. Case study - Web3j.		
UNIT III	**ETHEREUM**	**9**
The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code – Blocks and Blockchain – Fee Schedule – Supporting Protocols – Solidity Language.		
UNIT IV	**WEB3 AND HYPERLEDGER**	**9**
Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger – Case study - Corda.		
UNIT V	**ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS**	**9**
Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research - Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.		
		TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the technology components of Blockchain and how it works behind the scenes.

CO2: Understand the Bitcoin and its limitations by comparing with other alternative coins.

CO3: Develop deep understanding of the Ethereum model, its consensus model, code execution.

CO4: Understand the architectural components of a Hyperledger and its development framework.

CO5: Explore the alternative blockchains and its emerging trends.

CO6: Understand blockchain technology, including Bitcoin, Ethereum, and alternative blockchains.

TEXT BOOKS:

1. Imran Bashir, ―Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained‖, Second Edition, Packt Publishing, 2018.
2. Arshdeep Bahga, Vijay Madisetti, ―Blockchain Applications: A Hands-On Approach‖, VPT, 2017.

REFERENCES:

5. Andreas Antonopoulos, Satoshi Nakamoto, ―Mastering Bitcoin‖, O'Reilly Publishing, 2014.
6. Roger Wattenhofer, ―The Science of the Blockchain‖ CreateSpace Independent Publishing Platform, 2016.
3. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, ―Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction‖, Princeton University Press, 2016.
4. Alex Leverington, ―Ethereum Programming‖, Packt Publishing, 2017.
5. Antony Lewis ―The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them‖, Mango Publishing 2018.
6. Andreas M. Antonopoulos, ―Mastering Bitcoin: Programming the Open Block chain‖,

O'Reilly Publishing, 2017.
7. Massimo Ragnedda, Giuseppe Destefanis, ¯Blockchain and Web 3.0: Social, Economic, and Technological Challenges‖, Routledge, 2019.

22CS004	DIGITAL AND MOBILE FORENSICS	L		P	C
		3		0	3

OBJECTIVES:

The Course will enable learners to:

- Learn how to acquire digital forensic evidence.
- Learn how to investigate different digital artifacts and write reports
- Understand network forensics processes and procedures
- Understand mobile forensics processes and procedures.
- Able to analyze SIM cards and analyze mobile file systems.

UNIT I	ACQUIRING DIGITAL FORENSICS EVIDENCE	9

Types of Computer-Based Investigations - The Forensic Analysis Process- Acquisition of Evidence - Computer Systems- Case Study: Use The Sleuth Kit and Autopsy to retrieve data from file disk

UNIT II	DIGITAL FORENSICS INVESTIGATION& REPORTING	9

Computer Investigation Process-Windows Artifact Analysis-RAM Memory Forensic Analysis-Email Forensics-Investigation Techniques-Internet Artifacts- Case Study: Use DumpIt to acquire memory

UNIT III	NETWORKING FORENSICS	9

Characteristics in the network world-Identifying threats to the enterprise-Data breach surveys-Defining network forensics-Differentiating between computer forensics and network forensics-Digital footprints-Collecting network traffic using tcp dump-Collecting network traffic using Wireshark-Collecting network logs-Acquiring memory using FTK Imager-Tapping into network traffic-Packet sniffing and analysis using Wireshark-Packet sniffing and analysis using Network Miner

UNIT IV	MOBILE FORENSICS FUNDAMENTALS	9

Mobile Devices vs. Computer Devices in the World of Forensics-Living in the Cloud: The Place to Hide and Store Mobile Data-Preparing, Protecting, and Seizing Digital Device Evidence

UNIT V	ANALYSING MOBILE INTERNALS	9

Analyzing SIM Cards - Advanced Android Analysis - Advanced iOS Analysis-Case Study: Use Andriller or equivalent to extract data from Android

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand how to acquire digital forensic evidence.
CO2: Understand how to investigate different digital artifacts and write reports
CO3: Understand network forensics processes and procedures.
CO4: Understand mobile forensics processes and procedures.
CO5: Analyze SIM cards and analyze mobile file systems.
CO6: Understand Digital forensic techniques for comprehensive documentation and reporting.

TEXTBOOKS:

3. William Oettinger, ¯Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence‖, Packt Publishing, 1stEdition, 2020
4. Samir Datt, ¯Learning Network Forensics‖, Packt Publishing, 1st Edition, 2016

REFERENCES:

4. Lee Reiber, ¯Mobile Forensic Investigations: A Guide to Evidence Collection, Analysis, and Presentation‖, McGraw Hill, 2ndEdition, 2018.
5. Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty, ¯Practical Mobile Forensics‖, Packt Publishing, 3rdEdition, 2018
6. Gerard Johansen, ¯Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response‖, Packt Publishing, 3rdEdition, 2022

22CS005	UI/UX DESIGN	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners:

- Explain the principles of User Interface (UI) in order to do design with intention.
- Define the User eXperience (UX) and the psychology behind user decision making.
- Discuss about UX process and user Psychology.
- Apply technology for designing web applications with multimedia effects.
- Create a wireframe and prototype.

UNIT I	INTRODUCTION TO UI	9

Introduction to UI - Designing Behaviour: Designing with Intention - Conditioning and Addiction - Timing Matters - Gamification - Social/Viral Structure-Trust - Hidden versus Visible. Basic Visual Design Principles: Visual Weight - Contrast - Depth and Size – Color-Layout: Page Framework - Footers - Navigation -Images, and Headlines - Forms - Input Types - Labels and Instructions - Primary and Secondary Buttons - Adaptive and Responsive Design - Touch versus Mouse.

List of Exercise/Experiments:

1. Design UI for a Game website.
2. Design one-page UI for a website.

UNIT II	USER OBSERVATION AND EXPERIENCE	9

User Research - Subjective Research - Objective Research - Sample size - Three Basic Types of Questions. Observe a user: Watch How They Choose - Interviews - Surveys - Card Sorting - Creating User Profiles - Bad profile - Useful profile.

List of Exercise/Experiments:

1. Design UI for a mobile.
2. Explore the Look and Feel of the new Project developed in Ex1.

UNIT III	INTRODUCTION TO UX	9

Introduction about UX - Five Main Ingredients of UX - Three ¯Whats‖ of user Perspective - Pyramid of UX Impact - UX Is a Process - UX - Not an Event or Task. Behaviour Basics: Psychology versus Culture - User Psychology - Experience - Conscious vs Subconscious Experience - Emotions - Gain and Loss – Motivations.

List of Exercise/Experiments:

1. Design a mascot for an imaginary brand.

2. Create a Sample Pattern Library for a product (Mood board, Fonts, Colors based on UI principles).

UNIT IV	WEB INTERFACE DESIGN	9

Designing Web Interfaces – Drag and Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow – Using Motion for UX - Design Pattern: Z-Pattern - F-Pattern - Visual Hierarchy - Lookup patterns – Feedback patterns.

List of Exercise/Experiments:

1. Design a mock-up website for a service sector company.
2. Create a brainstorming feature for proposed product.

UNIT V	WIREFRAMING, PROTOTYPING AND TESTING	9

Sketching Principles - Sketching Red Routes - Responsive Design – Wireframing - Creating Wire flows - Building a Prototype - Building High-Fidelity Mock-ups - Designing Efficiently with Tools - Interaction Patterns - Conducting Usability Tests - Other Evaluative User Research Methods - Synthesizing Test Findings - Prototype Iteration

List of Exercise/Experiments:

1. Sketch, design with popular tool and build a prototype and perform usability testing and identify improvements.
2. Design a mobile mock-up website for an online store.

TOTAL: 30 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the principles of User Interface (UI) Design in order to design with intention

CO2: Learn the effective User eXperience (UX) and the psychology behind user decision making.

CO3: Understand the importance of UX process and user Psychology.

CO4: Elucidate the implications for designing web application with multimedia effects.

CO5: Create Wireframe and Prototype.

TEXT BOOKS:

1. Joel Marsh, ¯UX for Beginners‖, O'Reilly Media, Inc., 1st Edition 2015.
2. Xia Jiajia, ¯UI UX Design‖, O'Reilly, Artpower International, 2016.
3. Jenifer Tidwell, Charles Brewer, Aynne Valencia, ¯Designing Interface‖ 3rd Edition, O'Reilly 2020

REFERENCES:

1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, ¯Designing Interface‖ 3rd Edition , O'Reilly 2020.
2. Steve Schoger, Adam Wathan ¯Refactoring UI‖, 2018.
3. https://www.uxai.design/#:~:text=for%20designers,for%20AI%20products%20and%20services.

SOFTWARE REQUIREMENTS:

22CS006	INTRODUCTION TO COMPUTER NETWORKS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To study the fundamental concepts of computer networks and physical layer.
- To apply the knowledge of various protocols and techniques used in the data link layer.
- To implement the services of network layer and network layer protocols.
- To illustrate different protocols used in the transport layer.
- To build applications using the application layer protocols.

UNIT I	INTRODUCTION AND PHYSICAL LAYER	9

Data Communications – Network Types – Protocol Layering – Network Models (OSI, TCP/IP) Networking Devices: Hubs, Bridges, Switches – Performance Metrics – Transmission media - Guided media -Unguided media- Switching-Circuit Switching - Packet Switching.

UNIT II	DATA LINK LAYER	9

Introduction – Link-Layer Addressing- Error Detection and Correction - Wired LANs: Ethernet - Wireless LANs – Introduction – IEEE 802.11, Bluetooth

UNIT III	NETWORK LAYER	9

Network Layer Services – IPV4 Addresses – Forwarding of IP Packets - Network Layer Protocols: IP, ICMP v4 – Unicast Routing Algorithms – Protocols – Multicasting Basics – IPV6 Addressing – IPV6 Protocol.

UNIT IV	TRANSPORT LAYER	9

Introduction – Transport Layer Protocols – Services – Port Numbers – User Datagram Protocol –Transmission Control Protocol – SCTP.

UNIT V	APPLICATION LAYER	9

Application layer-WWW and HTTP – FTP – Email –Telnet –SSH – DNS – SNMP

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to:

CO1: Understand the fundamental concepts of computer networks.

CO2: Apply the various routing protocols to solve real-world problems.

CO3: Understand the layered architecture.

CO4: Apply the simulation tools to implement various protocols used in the various layers.

CO5: Analyze the various application layer protocols.

CO6: Apply the mathematical knowledge to do performance analysis of various routing protocols.

TEXT BOOK:

1. Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill Education, 5th Ed., 2017.

REFERENCES:

1. Computer Networking- A Top Down Approach, James F. Kurose, University of Massachusetts and Amherst Keith Ross, 8th Edition, 2021.
2. Computer Networks, Andrew S. Tanenbaum, Sixth Edition, Pearson, 2021.
3. Data Communications and Computer Networks, P.C. Gupta, Prentice-Hall of India, 2006.
4. Computer Networks: A Systems Approach, L. L. Peterson and B. S. Davie, Morgan Kaufmann, 3rd ed., 2003.

LIST OF EQUIPMENTS: