



ANNA UNIVERSITY, CHENNAI
UNDERGRADUATE CURRICULUM (UNIVERSITY DEPARTMENTS)

Campus: College of Engineering Guindy & Madras Institute of Technology

Department: Computer Science and Engineering & Computer Technology

Programme: B.E. Computer Science and Engineering

Regulations: 2023 (Revised 2024), with effect from the AY 2024 – 25 to all the students of UG Programme.

OVERVIEW OF CREDITS

Sem	PCC	PEC	ESC	HSMC	ETC	OEC	SDC	UC	SLC	Total
I			11	7			2	1		21
II	3		4	11			3	1		22
III	13		5	4			1	3		26
IV	17			4			2	0		23
V	13	3				3	2	3	1	25
VI	13	6				3	2	3		27
VII		9			7		1			17
VIII							8			8
Total	59	18	20	26	7	6	21	11	1	169
% of Category	34.91	10.65	11.83	15.38	4.14	3.55	12.42	6.51	0.59	

CATEGORY OF COURSES

PCC – Professional Core Course

PEC – Professional Elective Course

ETC – Emerging Technology Course

OEC – Open Elective Course

SLC – Self Learning Course

ESC – Engineering Science Course

HSMC – Humanities Science and Management Course

SDC – Skill Development Course

UC – University Course

**For Honours & Minor Degree, please refer the Regulations 2023 (Revised 2024).*

SEMESTER – III							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS	CATE GORY
				L-T-P	TCP*		
1	MA23C05	Probability and Statistics	T	3-1-0	4	4	HSMC
2	CS23301	Software Engineering	T	3-0-0	3	3	PCC
3	CS23302	Data Structures	LIT	3-0-4	7	5	PCC
4	CS23303	Digital System Design	LIT	3-0-4	7	5	ESC
5	CS23304	Java Programming	LIT	3-0-4	7	5	PCC
6	CS23S01	Skill Development Course – (Level II)	L	0-0-2	2	1	SDC
7	CS23U01	Standards – Computer Science & Engg.	T	1-0-0	1	1	UC
8	UC23U01	Universal Human Values	LIT	1-0-2	3	2	UC
TOTAL CREDITS						26	

SEMESTER – IV							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS	CATEG ORY
				L-T-P	TCP*		
1		Linear Algebra and Numerical Methods	T	3-1-0	4	4	HSMC
2		Database Management Systems	LIT	3-0-4	7	5	PCC
3		Computer Architecture	LIT	3-0-2	5	4	PCC
4		Full Stack Technologies	LIT	3-0-4	7	5	PCC
5		Design and Analysis of Algorithms	T	3-0-0	3	3	PCC
6		Skill Development Course - Level II	L	0-0-2	2	1	SDC
7		Audit Course II	T	2-0-0	2	0	UC
8		Industry Oriented Course I	L	0-0-2	2	1	SDC
TOTAL CREDITS						23	

SEMESTER – V (PREFERENCE FOR FOREIGN EXCHANGE)							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1		Operating Systems	LIT	3-0-4	7	5	PCC
2		Networks and Data Communication	LIT	3-0-4	7	5	PCC
3		Theory of Computation	T	3-0-0	3	3	PCC
4		Professional Elective – I	-	-	-	3	PEC
5		Open Elective – I	T	3-0-0	3	3	OEC
6		Self Learning Course	-	-	-	1	SLC
7		Industry Oriented Course -II/ Summer Internship - I**	IPW	0-0-2	2	1	SDC
8		Entrepreneurship Development Course	T	3-0-0	3	3	UC
9		Skill Development Course - Level III	L	0-0-2	2	1	SDC
TOTAL CREDITS						25	
COURSES FOR HONOURS DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Capstone – Ideation	CDP	0-0-8	8	4	SDC
2		Skill Development Course – (Design Skills)	LIT	1-0-2	3	2	SDC
TOTAL CREDITS						6	
(OR)							
1.		Honours Elective – I	-	-	-	3	PEC
2.		Honours Elective – II	-	-	-	3	PEC
COURSES FOR MINOR DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Minor Elective – I	-	-	-	-	-
2.		Minor Elective – II	-	-	-	-	-

SEMESTER – VI (PREFERENCE FOR FOREIGN EXCHANGE)							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1		Cryptography and System Security	LIT	3-0-2	5	4	PCC
2		Compiler Design	LIT	3-0-2	5	4	PCC
3		Machine Learning	LIT	3-0-4	7	5	PCC
4		Professional Elective – II	-	-	-	3	PEC
5		Professional Elective – III	-	-	-	3	PEC
6		Open Elective – II	T	3-0-0	3	3	OEC
7		Sustainability Course	T	3-0-0	3	3	UC
PRACTICALS							
8		Creative and Innovative Project	L	0-0-4	4	2	SDC
TOTAL CREDITS						27	
COURSES FOR HONOURS DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Self Learning Course – II	-	-	-	1	SLC
2		Skill Development Course	L	0-0-4	4	2	SDC
3		Capstone Project Phase I (Proof of Concept, Implementation & Validation)	CDP	0-0-6	6	3	SDC
TOTAL CREDITS						6	
(OR)							
1.		Honours Elective – III	-	-	-	3	PEC
2.		Honours Elective – IV	-	-	-	3	PEC
COURSES FOR MINOR DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Minor Elective – III	-	-	-	-	-
2.		Minor Elective – IV	-	-	-	-	-

SEMESTER – VII							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1		Embedded Systems and Internet of Things	LIT	3-0-2	5	4	ETC
2		Artificial Intelligence	T	3-0-0	3	3	ETC
3		Professional Elective – IV	-	-	-	3	PEC
4		Professional Elective – V	-	-	-	3	PEC
5		Professional Elective – VI	-	-	-	3	PEC
6		Industry Oriented Course III / Summer internship II***	IPW	0-0-2	2	1	SDC
TOTAL CREDITS						17	
COURSES FOR HONOURS DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Capstone Project Phase II (Product Development –Publication / Patent Submission)	CDP	0-0-12	12	6	SDC
TOTAL CREDITS						6	
(OR)							
1.		Honours Elective – V	-	-	-	3	PEC
2.		Honours Elective – VI	-	-	-	3	PEC
COURSES FOR MINOR DEGREE							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS	CATEGORY
				L-T-P	TCP*		
1.		Minor Elective – V	-	-	-	-	-
2.		Minor Elective – VI	-	-	-	-	-

SEMESTER – VIII							
S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS	CATEG ORY
				L-T-P	TCP*		
1.		Project Work / Semester Long Internship	PW	0-0-16	16	8	SDC
TOTAL CREDITS						8	

PROFESSIONAL ELECTIVE COURSES: VERTICALS						
VERTICAL I	VERTICAL II	VERTICAL III	VERTICAL IV	VERTICAL V	VERTICAL VI	VERTICAL VII
DATA SCIENCE	FULL STACK DEVELOPMENT	CLOUD COMPUTING AND DATA CENTER TECHNOLOGIES	CYBER SECURITY AND DATA PRIVACY	CREATIVE MEDIA	EMERGING TECHNOLOGIES	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
Exploratory data Analysis	Web Technologies	Cloud Computing	Ethical Hacking	Distributed Systems	Augmented Reality/ Virtual Reality	Knowledge Engineering
Recommender Systems	App Development	Virtualization	Digital and Mobile Forensics	Multimedia and Animation	Robotic Process Automation	Soft Computing
Data Warehousing	Cloud Services Management	Information Visualization	Social Network Security	Video Creation and Editing	Health Care Analytics	Deep Learning
Data Mining	UI and UX Design	Unix Internals	Modern Cryptography	Game Theory	Large Language Models	Text and Speech Analysis
Business Analytics	Software Testing and Automation	Storage Technologies	Engineering Secure Software Systems	Digital marketing	Quantum Computing	Optimization Techniques
Image and Video Analytics	Web Application Security	Software Defined Networks	Cyber Security	Visual Effects	Cryptocurrency and Block chain Technologies	Social Network Analysis
Natural Language Processing	DevOps	Stream Processing	Network Security	Game Development	Metaverse	Cognitive Science
Big Data Analytics	Programming Paradigms	Security and Privacy in Cloud	Information Security	Graph Theory	3D Printing and Design	Responsible AI
Bio Informatics	Software Project Management	GPU Computing	Mobile Networks	Image Processing	Autonomous Vehicle	Generative AI

VERTICAL I: DATA SCIENCE

SL. NO.	COURSE CODE	COURSE TITLE	CATE GORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1.		Exploratory Data Analysis	PEC	2	0	2	4	3
2.		Recommender Systems	PEC	2	0	2	4	3
3.		Data Warehousing	PEC	3	0	0	3	3
4.		Data Mining	PEC	3	0	0	3	3
5.		Business Analytics	PEC	2	0	2	4	3
6.		Image and Video Analytics	PEC	2	0	2	4	3
7.		Natural Language Processing	PEC	3	0	0	3	3
8.		Big Data Analytics	PEC	2	0	2	4	3
9.		Bio Informatics	PEC	2	0	2	4	3

VERTICAL II: FULL STACK DEVELOPMENT

SL. NO.	COURSE CODE	COURSE TITLE	CATE GORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1.		Web Technologies	PEC	2	0	2	4	3
2.		App Development	PEC	2	0	2	4	3
3.		Cloud Services Management	PEC	2	0	2	4	3
4.		UI and UX Design	PEC	2	0	2	4	3
5.		Software Testing and Automation	PEC	2	0	2	4	3
6.		Web Application Security	PEC	2	0	2	4	3
7.		DevOps	PEC	2	0	2	4	3
8.		Programming Paradigms	PEC	3	0	0	3	3
9.		Software Project Management	PEC	3	0	0	3	3

VERTICAL III: CLOUD COMPUTING AND DATA CENTER TECHNOLOGIES

SL. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1		Cloud Computing	PEC	2	0	2	3	3
2.		Virtualization	PEC	2	0	2	3	3
3.		Information Visualization	PEC	3	0	0	3	3
4.		Unix Internals	PEC	3	0	0	3	3
5.		Storage Technologies	PEC	3	0	0	3	3
6.		Software Defined Networks	PEC	2	0	2	4	3
7.		Stream Processing	PEC	2	0	2	4	3
8.		Security and Privacy in Cloud	PEC	2	0	2	4	3
9		GPU Computing	PEC	3	0	0	3	3

VERTICAL IV- CYBER SECURITY AND DATA PRIVACY

SL. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1		Ethical Hacking	PEC	2	0	2	4	3
2.		Digital and Mobile Forensics	PEC	2	0	2	4	3
3.		Social Network Security	PEC	2	0	2	4	3
4.		Modern Cryptography	PEC	2	0	2	4	3
5.		Engineering Secure Software Systems	PEC	2	0	2	4	3
6.		Cyber security	PEC	2	0	2	4	3
7.		Network Security	PEC	2	0	2	4	3
8.		Information Security	PEC	3	0	0	3	3
9.		Mobile Networks	PEC	3	0	0	3	3

VERTICAL V: CREATIVE MEDIA

SL. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1		Distributed Systems	PEC	3	0	0	3	3
2.		Multimedia and Animation	PEC	2	0	2	4	3
3.		Video Creation and Editing	PEC	2	0	2	4	3
4.		Game Theory	PEC	2	0	2	4	3
5.		Digital marketing	PEC	2	0	2	4	3
6.		Visual Effects	PEC	2	0	2	4	3
7.		Game Development	PEC	2	0	2	4	3
8.		Graph Theory	PEC	3	0	0	3	3
9.		Image Processing	PEC	3	0	0	3	3

VERTICAL VI: EMERGING TECHNOLOGIES

SL. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1		Augmented Reality/ Virtual Reality	PEC	2	0	2	4	3
2.		Robotic Process Automation	PEC	2	0	2	4	3
3.		Health Care Analytics	PEC	3	0	0	3	3
4.		Large Language Models	PEC	3	0	0	3	3
5.		Quantum Computing	PEC	2	0	2	4	3
6.		Cryptocurrency and Blockchain Technologies	PEC	2	0	2	4	3
7.		Metaverse	PEC	3	0	0	3	3
8.		3D Printing and Design	PEC	2	0	2	4	3
9.		Autonomous Vehicle	PEC	3	0	0	3	3

VERTICAL VII: ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			TOTAL CONTACT PERIODS	CREDITS
				L	T	P		
1		Knowledge Engineering	PEC	2	0	2	4	3
2.		Soft Computing	PEC	2	0	2	4	3
3.		Deep Learning	PEC	2	0	2	4	3
4.		Text and Speech Analysis	PEC	2	0	2	4	3
5.		Optimization Techniques	PEC	2	0	2	4	3
6.		Social Network Analysis	PEC	2	0	2	4	3
7.		Cognitive Science	PEC	2	0	2	4	3
8.		Responsible AI	PEC	3	0	0	3	3
9.		Generative AI	PEC	3	0	0	3	3

Any 6 PECs can be taken by students to complete 'Honours Degree'.

MINOR PROGRAMME ON CYBER SECURITY

Offered by Department of Computer Science and Engineering for other Branch students.

S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS
				L-T-P	TCP*	
1		Data Structures	T	3-0-0	3	3
2		Computer Networks	T	3-0-0	3	3
3		Ethical Hacking	LIT	2-0-2	4	3
4		Cyber Security	LIT	2-0-2	4	3
5		Cyber Forensics	T	3-0-0	3	3
6		Cryptography and Network Security	T	3-0-0	3	3

EMERGING TECHNOLOGY COURSES (ETC)

S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS
				L-T-P	TCP*	
1		Embedded Systems and Internet of Things	LIT	3-0-2	5	4
2		Artificial Intelligence	T	3-0-0	3	3

OPEN ELECTIVE COURSES (OEC)

S. NO.	COURSE CODE	COURSE NAME	COURSE TYPE#	PERIODS / WEEK		CREDITS
				L-T-P	TCP*	
1		Open Elective – I	OEC	3-0-0	3	3
2		Open Elective – II	OEC	3-0-0	3	3

OPEN ELECTIVE

(TO BE OFFERED TO OTHER DEPARTMENT)

S. NO.	COURS ECODE	COURSE NAME	COURSE TYPE#	PERIODS /WEEK		CREDITS
				L-T-P	TCP*	
1		Big Data Analytics	T	3-0-0	3	3
2		Web Design and Management	T	3-0-0	3	3
3		Cyber Forensics	T	3-0-0	3	3
4		Agile Methodologies	T	3-0-0	3	3

OBJECTIVES:

- To understand the basics of random variables with emphasis on the standard discrete and continuous distributions.
- To understand the basic probability concepts with respect to two dimensional random variables along with the relationship between the random variables and the significance of the Central Limit theorem.
- To understand the basic concepts of sampling distributions and statistical properties of point and interval estimators.
- To apply the small/ large sample tests through Tests of hypothesis.
- To understand the concept of analysis of variance and use it to investigate factorial dependence.

UNIT I ONE-DIMENSIONAL RANDOM VARIABLES**9+3**

Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma and Normal distributions – Functions of a random variable.

UNIT II TWO-DIMENSIONAL RANDOM VARIABLES**9+3**

Joint distributions – Marginal and conditional distributions – Covariance – Correlation and Linear regression – Transformation of random variables – Central limit theorem (for independent and identically distributed random variables).

UNIT III ESTIMATION THEORY**9+3**

Sampling distributions – Characteristics of good estimators – Method of Moments – Maximum Likelihood Estimation – Interval estimates for mean, variance and proportions.

UNIT IV TESTS OF SIGNIFICANCE**9+3**

Type I and Type II errors – Tests for single mean, proportion, Difference of means (large and small samples) – Tests for single variance and equality of variances – χ^2 test for goodness of fit – Independence of attributes.

UNIT V DESIGN OF EXPERIMENTS**9+3**

Completely Randomized Design – Randomized Block Design – Latin Square Design – 2^2 factorial design.

TOTAL: 60 PERIODS

Laboratory based exercises / assignments / assessments will be given to students from the content of the course wherever applicable.

Branch specific / General Engineering applications based on the content of each units will be introduced to students wherever possible.

SUGGESTED LAB EXERCISES

1. Data exploration using R
2. Visualizing Probability distributions graphically
3. Evaluation of correlation coefficient

4. Creating a Linear regression model in R
5. Maximum Likelihood Estimation in R
6. Hypothesis testing in R programming
7. Chi square goodness of fit test in R
8. Design and Analysis of experiments with R

OUTCOMES:

- CO1: Can analyze the performance in terms of probabilities and distributions achieved by the determined solutions.
- CO2: Will be familiar with some of the commonly encountered two dimensional random variables and be equipped for a possible extension to multivariate analysis.
- CO3: Provides an estimate or a range of values for the population parameter from random samples of population.
- CO4: Helps to evaluate the strength of the claim/assumption on a sample data using hypothesis testing.
- CO5: Equips to study the influence of several input variables on the key output variable.

TEXT BOOKS:

1. Irwin Miller and Marylees Miller, "John E. Freund's Mathematical Statistics with applications", Pearson India Education, Asia, 8th Edition, 2014.
2. Walpole, R.E., Myers R.H., Myres S.L., and Ye, K. "Probability and Statistics for Engineers and Scientists", Pearson Education, Asia, 9th Edition, 2024.

REFERENCES:

1. Richard A. Johnson, Irwin Miller, John Freund "Miller & Freund's Probability and Statistics for Engineers", Person Education, 8th Edition, 2015.
2. Ross, S.M. "Introduction to Probability and Statistics for Engineers and Scientists", Elsevier, New Delhi, 5th Edition, 2014.
3. Spiegel, M.R., Schiller, J., Srinivasan, R.A. and Goswami, D. "Schaum's Outline of Theory and Problems for Probability and Statistics", McGraw Hill Education, 3rd Edition, Reprint, 2017.
4. Devore, J.L. "Probability and Statistics for Engineering and the Sciences", Cengage Learning, 9th Edition, 2016.

CO – PO Mapping:

COURSE OUTCOMES	PROGRAMME OUTCOMES											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	3	3	2	3	1	2	1	1	1	1	1	3
CO2	3	3	2	3	1	2	1	1	1	1	1	3
CO3	3	3	2	3	1	2	1	1	1	1	1	3
CO4	3	3	2	3	1	2	1	1	1	1	1	3
CO5	3	3	2	3	1	2	1	1	1	1	1	3

UNIT – I SOFTWARE PROCESS MODELS**9L**

Introduction – Software Development Life Cycle (SDLC) – Traditional SDLC Models- Alternative Techniques - Rapid Application Development (RAD) - Agile Development Models – Scrum- JIRA tool : backlog management, sprint planning, Product Discovery, team collaboration – XP Programming

Suggested activities

- Analyze case studies comparing different software process models (e.g., Waterfall vs. Agile).
- Conduct a Scrum simulation where students assume roles such as Product Owner, Scrum Master, and Development Team.
- Conduct a retrospective meeting at the end of a project or simulation to reflect on what went well, what didn't, and areas for improvement.

Suggested Evaluations

- Pair programming where students practice XP techniques like test-driven development and continuous integration - Bus/Train Route Information: Provide online information about bus routes, frequencies, and fares

UNIT – II SOFTWARE PROJECT MANAGEMENT**9L**

Responsibilities of a Software Project Manager - Project Planning - Metrics for Project Size Estimation: Project Estimation Techniques, COCOMO—A Heuristic Estimation Technique (Basic, Intermediate & Complete) – Scheduling: CPM & PERT - Risk Management: Identification, Assessment, Mitigation

Suggested activities:

- Analyse case studies of successful and unsuccessful software projects to identify key project management practices and lessons learned.
- A workshop/ group activity, apply COCOMO (Basic, Intermediate, and Complete) to estimate the size and effort of a software project.

Suggested Evaluations:

- Use project management software (e.g., Microsoft Project, Primavera) to create and manage a project plan. Include tasks such as defining milestones, scheduling activities using CPM and PERT, and assigning resources.
- Use a software tool (e.g., COCOMO II Model) to input project parameters and calculate effort, cost, and time estimates for different project scenarios

UNIT – III REQUIREMENTS ANALYSIS**9L**

Requirements Gathering and Analysis - Software Requirements Specification (SRS) - Formal System Specification - Executable Specification and 4GL – Eliciting Accurate Requirements - Validating Requirements - Achieving Requirements Traceability - Managing Changing Requirements - Reviews, Walkthroughs, and Inspections - Tools: Jama Software, JIRA & IBM Rational Doors. Discussion on management tools (e.g., Jama Software, JIRA, IBM Rational Doors) to document, track, and manage software requirements throughout the project lifecycle.

Suggested activities:

- Students create a Software Requirements Specification (SRS) document for a hypothetical

software project, including functional and non-functional requirements.

- Students work on a capstone project incorporating all aspects of requirements gathering and analysis.

Suggested Evaluations:

- The SRS document should include all necessary sections, such as introduction, overall description, specific requirements, appendices, and glossary.

UNIT – IV SOFTWARE MODELLING AND DESIGN

9L

Elements of Software Modelling and Design -Translating Requirement model into design model- Design notations: Data Flow Diagram (DFD), Structured Flowcharts, Decision Tables (Low Level Design, High Level Design) – UML - Data Modelling- Analysis Modelling: Elements of Analysis model - Design modelling: The Design Process, Design Concepts –Architectural Mapping using Data Flow. Design Tool: Sparx Enterprise Architect - DevOps : Core elements, Life cycle, Adoption of DevOps - DevOps Tools.

Suggested activities

- Sparx Enterprise Architect to create design models from their requirements, including class diagrams, sequence diagrams, and data flow diagrams.
- Facilitate a rapid innovation sprint where students follow a structured process to ideate, prototype, and pitch solutions within a short timeframe.

Suggested Evaluations:

- Conduct user interviews to gather insights and create empathy maps to visualize user needs and pain points.
- Use DevOps tools (e.g., Jenkins, Docker, Kubernetes) to set up a continuous integration and continuous deployment (CI/CD) pipeline.

UNIT – V CODING, TESTING & MAINTENANCE

9L

Coding Standards and Guidelines - Code Review - Development Tools: IDEs (e.g., XCode, Eclipse, IntelliJ IDEA, Atom) - Testing: Black-box Testing, White-Box Testing, Integration Testing, System Testing – Tools: Junit, Selenium - Characteristics of Software Maintenance -Software Reverse Engineering - Software Maintenance Process Models - Estimation of Maintenance Cost.

Suggested Activities:

- Implement and execute various tests (black-box, white-box, integration, system) on a provided codebase using testing tools like JUnit and Selenium.
- Use automated code review tools (e.g., SonarQube, CodeClimate) to analyze and improve the quality of a codebase.

Suggested Evaluations:

Develop a small software which accommodate the learning objectives of the course Projects:

- Criminal Record Management: Develop a system to manage criminal records for jailers, police officers, and CBI officers.
- Car Pooling: Create a web-based intranet application to facilitate carpooling among corporate employees within an organization.

- Patient Appointment and Prescription Management System: Develop a system to manage patient appointments and prescriptions.
- Examination and Result Computation System: Develop a system for managing examinations and computing result
- Automatic Internal Assessment System: Implement a system for automatic internal assessments.
- Any other use cases.

TOTAL : 45 PERIODS

COURSE OUTCOMES:

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

1. Explain and compare various software process models, demonstrating knowledge of their uses.
2. Competence in planning and managing software projects using estimation and scheduling methods like COCOMO, CPM, and PERT.
3. Employ design thinking techniques to create innovative, user-focused solutions and prototypes.
4. Skill in writing clean code, conducting effective code reviews, and applying testing methods such as black-box and white-box testing.
5. Knowledge of DevOps principles and tools, and ability to implement DevOps practices in continuous integration and delivery.

REFERENCES:

1. Rajib Mall, Fundamentals of Software Engineering, Fourth Edition, 5th Edition, PHI Learning
2. Roger S. Pressman, Software Engineering – A Practitioner's Approach", 7 th Edition McGraw Hill Publications
3. Sommerville, Software Engineering", 8th Edition Pearson Education
4. Pankaj Jalote, Software Engineering -A Precise Approach, Wiley India
5. Waman S Jawadekar Software Engineering principles and practice, The McGraw-Hill Companies
6. Roman Pichler, Agile Product Management with Scrum
7. Ken Schwaber, Agile Project Management with Scrum (Microsoft Professional)

Links for the tools

- <https://www.atlassian.com/software/jira/download>.
- <https://www.microsoft.com/en-in/microsoft-365/project/project-management-software>
- <https://www.oracle.com/in/construction-engineering/primavera-p6/>
- <http://softwarecost.org/tools/COCOMO/>
- <https://www.jamasoftware.com/platform/jama-connect/trial/>
- <https://www.ibm.com/support/pages/ibm-rational-doors-version-9304>
- <https://www.smartdraw.com/software/pert-chart-software.htm>
- <https://www.smartdraw.com/software/pert-chart-software.htm>

UNIT – I LINEAR DATA STRUCTURES**10L, 12P**

Introduction to Data Structures & Algorithms – Asymptotic notations - ADT – Array - List - Linked List - Singly Linked List - Doubly Linked List – Circular List – Elementary Operations - Stack – Queue - Array Implementation – Linked List Implementation – Applications.

PRACTICALS:

1. Array implementation of Stack, Queue ADTs
2. Linked list implementation of Stack and Queue ADTs
3. Applications of Stack and Queue ADTs.

UNIT – II NON-LINEAR DATA STRUCTURES**9L, 12P**

Tree - Binary Tree – Properties – Representation - Binary Search Tree – Operation - Traversal – Threaded Binary Tree – Heap – Priority Queue implementation with heap - Graph - Terminologies – Representation – BFS - DFS - Connected Components – Minimum Spanning Tree.

PRACTICALS:

1. Implementation of Binary Trees, Traversal operations
2. Implementing Heap structure
3. Implementing search algorithms on graph

UNIT – III SORTING**8L, 12P**

Insertion Sort – Heap Sort - Merge Sort – Quick Sort – Counting Sort – External Sorting – Multi-way Merge Sort.

PRACTICALS:

1. Implementation of Sorting Algorithms

UNIT – IV SEARCHING**6L, 12P**

Linear Search – Binary Search - Hash Function – Separate Chaining - Linear Probing – Quadratic Probing – Double Hashing – Rehashing.

PRACTICALS:

1. Implementation of Searching Algorithm
2. Operations on Binary Search Trees

UNIT – V SEARCH STRUCTURES**12L, 12P**

Balanced Tree - AVL Tree – Red Black Tree – Multi-way Search Tree - B-Tree – Binary Trie – Multi-way Trie – Suffix tree.

PRACTICALS:

1. Implementing balanced BST

TOTAL: 45L + 60P = 105 PERIODS**COURSE OUTCOMES:**

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

1. Apply efficient data structures required for an application
2. Compare sorting algorithms on time complexity
3. Understand, design and implement linear and non-linear data structures

4. Appreciate basic and optimal search structures
5. Select suitable search algorithm for an application

REFERENCES:

1. Ellis Horowitz and Sartaj Sahni, Anderson Freed "Fundamentals of Data Structures in C", University Press, 2008.
2. Ellis Horowitz and Sartaj Sahni, Dinesh Mehta "Fundamentals of Data Structures in C++", Silicon Press, 2007.
3. Yashavant Kanetkar, "Data Structures Through C", BPB press, 4th edition, 2022.
4. Michael T. Goodrich, Roberto Tamassia "Data Structures and Algorithms in Python", Wiley, 2021.
5. Jean-Paul Tremblay and Paul G Sorenson, "An Introduction to Data Structures with Applications", Second Edition, McGrawHill, 2017.
6. Thomas H Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Third Edition, Prentice Hall, 2010.
7. Ellis Horowitz and Sartaj Sahni, "Fundamental of Computer Algorithms", Galgotia, 1985.

Mapping of CO with PO

CO-PO Mapping

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

1 - low, 2 - medium, 3 - high, '-' - no correlation

UNIT – I BOOLEAN ALGEBRA AND LOGIC GATES**9L, 12P**

Review of Number Systems – Arithmetic Operations – Binary Codes – Boolean Algebra and Theorems – Boolean Functions – Simplification of Boolean Functions using Karnaugh Map, Tabulation – Logic Gates – NAND and NOR Implementations.

PRACTICALS:

1. Verification of Boolean Theorems using basic gates.
2. Design and implementation of combinational circuits using basic gates for arbitrary functions, code converters.

UNIT – II COMBINATIONAL LOGIC**9L, 16P**

Combinational Circuits – Analysis and Design Procedures – Circuits for Arithmetic Operations, Code Conversion – Decoders and Encoders – Multiplexers – Real Time Application of Combinational Circuits- Introduction to HDL – HDL Models of Combinational circuits.

PRACTICALS:

1. Design and implementation of Parity generator / checker.
2. Design and implementation of Magnitude Comparator.
3. Design and implementation of an application using multiplexers.
4. Combinational circuits using HDL.

UNIT – III SYNCHRONOUS SEQUENTIAL LOGIC**10L, 12P**

Synchronous Sequential Logic: Sequential Circuits – Latches and Flip Flops – Shift Registers – Counters- State Reduction and State Assignment - Analysis and Design Procedures – HDL for Sequential Logic Circuits.

PRACTICALS:

1. Design and implementation of shift –registers.
2. Design and implementation of synchronous counters.
3. Sequential circuits using HDL.

UNIT – IV ASYNCHRONOUS SEQUENTIAL LOGIC**9L, 12P**

Asynchronous Sequential Logic: Analysis and Design of Asynchronous Sequential Circuits – Reduction of State and Flow Tables – Race-free State Assignment.

PRACTICALS:

1. Design and implementation of asynchronous counters.

UNIT – V MEMORY AND PROGRAMMABLE LOGIC**8L, 8P**

RAM and ROM – Memory Decoding – Error Detection and Correction – PROM- Programmable Logic Array – Programmable Array Logic – Sequential Programmable Devices.

PRACTICALS:

1. Design and implementation of a simple digital system.

TOTAL: 45L + 60P = 105 PERIODS

COURSE OUTCOMES:

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

1. Use theorems and K-maps to simplify Boolean functions.
2. Design, analyze and Implement combinational circuits.
3. Design, analyze and implement sequential circuits.
4. Design digital circuits using MSI chips and PLDs.
5. Use HDL to build digital systems

REFERENCES:

1. M.Morris Mano and Michael D.Ciletti, "Digital Design", VI Edition, Pearson Education, 2018.
2. G. K.Kharate, "Digital Electronics", First Edition, Oxford University Press, 2010.
3. John F.Wakerly, "Digital Design Principles and Practices", Fourth Edition, Pearson Education, 2007.
4. Charles H.Roth Jr, "Fundamentals of Logic Design", Seventh Edition – Jaico Publishing House, Mumbai, 2013.
5. Donald D.Givone, "Digital Principles and Design", Tata McGraw Hill, 2003.

CO-PO Mapping

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

1-low, 2-medium, 3-high, ‘-’- no correlation

UNIT – I INTRODUCTION TO JAVA**9L, 12P**

Introduction to Java – JVM - Data Types, Variables, Operators, Expressions – Control flow Statements – Methods - Arrays – Classes and Objects –Constructors - Access Specifiers –Static members- this –constants- - String Class- Working with Data and Time API

PRACTICALS:

1. Develop programs using Java Basic Constructs and Arrays using any standard IDE like NETBEANS / ECLIPSE
2. Programs to illustrate concept of class and static classes and methods
3. Implement programs using string class, Date and Time API

UNIT – II POLYMORPHISM AND INHERITANCE**9L, 12P**

Overloading Methods – Static, Nested and Inner Classes. Inheritance– super Classes and sub classes -Overriding –Downcasting – Operator instance of - Abstract Classes – final classes - Packages – Interfaces.

PRACTICALS:

1. Implement abstract classes, overloading and overriding
2. Implement Interfaces

UNIT – III EXCEPTION HANDLING AND MULTITHREADING**8L, 12P**

Exception Handling – Java's Built-in Exceptions – User defined Exception - Assertions. Multithreading- Priorities–Synchronization – Avoiding Deadlocks– Wrappers– Autoboxing and Unboxing.

PRACTICALS:

1. Programs using Exception handling
2. Programs using Multi-Threading and synchronisation

UNIT – IV FILE STREAMS AND DATABASE**9L, 12P**

Java I/O- Reading and Writing Files - Regular Expressions – Streams API - Object Serialization – Generic collections – Generic Classes - Lists, set, Maps – GenericMethods - Lambda expressions – Databases with JDBC.

PRACTICALS:

1. Programs using Streams API and File I/O for reading and writing the contents in sequential and random order.
2. Programs using Generics classes and methods
3. Event-driven programs for GUI applications with Database Connectivity

UNIT – V**WEB DEVELOPMENT AND FRAMEWORKS****10L, 12P**

Event handling: Event, Listeners and adapter classes, anonymous inner classes. Abstract Windowing Toolkit (AWT): Button, Label, Checkbox, Checkbox Group, Text Field, Text Area, Choice, List, Menu, Panel, Scrollbar and Swing components, Layout managers, Complex Components - Java Servlets - Apache Tomcat –Java Web Framework -Spring Boot

PRACTICALS:

1. Creation of servlet based applications
2. Design a Java Web application using Spring Boot

TOTAL: 45L + 60P = 105 PERIODS**COURSE OUTCOMES:**

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

1. Construct programs using Object Oriented Design principles like encapsulation, abstraction, polymorphism, inheritance and typing. (Apply)
2. Develop applications with handlers for user-defined exceptions, according to the given requirements (Apply)
3. Construct efficient multi-threaded programs with synchronization constructs. (Apply)
4. Develop interactive GUI applications with event handling that provide rich user experience. (Apply)
5. Construct programs using the suitable Collection classes and interfaces for efficient modelling of the objects and entities of the program. (Apply)
6. Develop Web Applications that use file input and output using any framework (Apply)

REFERENCES:

1. Y. Daniel Liang, "Introduction to Java Programming and Data Structures, Comprehensive Version", 12th Edition, Pearson Education, 2024.
2. Paul Dietel and Harvey Deitel, "Java - How to Program Early Objects", 11th Edition, Pearson Education, 2018.
3. Craig Walls, "Spring in Action", 3rd edition, Manning Publications, 2011.
4. Herbert Schildt, "Java: The Complete Reference", 11th Edition, McGraw-Hill Education, 2018.
5. Sachin Malhotra, Sourabh Choudhary, "Programming in Java", Revised 2nd Edition, Oxford University Press, 2018.
6. Cay S. Horstmann, "Core Java - Vol. 1, Fundamentals", 11th Edition, Pearson Education, 2018.
7. <https://spring.io/projects/spring-boot>

CO-PO Mapping

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

1-low,2-medium,3-high,'-'-no correlation

CS23U01	STANDARDS - COMPUTER SCIENCE & ENGINEERING	L	T	P	C
		1	0	0	1

MODULE I – OVERVIEW OF STANDARDS

6hrs

Basic concepts of standardization: Purpose of Standardization, marking and certification of articles and processes; Importance of standards to industry, policy makers, trade, sustainability and innovation. Objectives, roles and functions of BIS, Bureau of Indian Standards Act, ISO/IEC Directives; WTO Good Practices for Standardization. Important Indian and International Standards.

MODULE II INTERNATIONAL STANDARDS IN COMPUTER SCIENCE

9hrs

Introduction -Importance of standards in IT-Overview of key international standards organizations
ANSI and IEEE Standards - ANSI standards for software engineering (e.g., ANSI/ISO/IEC 12207:2008 - Software Life Cycle Processes)- IEEE standards and their applications in software engineering (e.g., IEEE 830-1998 - Requirements Specifications)-**ISO/IEC 20000:** IT Service Management -Scope and requirements-Service delivery process-Certification and implementation challenges- ISO 9000 Series: Quality Management -Overview of ISO 9001-Quality management principles-Certification process and benefits-

ITU-T Standards in Telecommunications-Overview of ITU-T series (e.g., ITU-T X.509 for public key infrastructure)-Impact on global telecommunications standards- **IETF Standards in Internet Protocols**-Overview of key IETF standards (e.g., RFC 791 for IPv4)-Evolution and adoption of internet protocols-**W3C Standards for the World Wide Web** -Key W3C standards (e.g., HTML5, CSS3, Web Accessibility Guidelines)-Role of standards in web development and interoperability

ISO/IEC 27001: Information Security Management -Principles and Framework-Risk assessment and Management-Controls and compliance-**NIST Standards and Frameworks** -NIST Cybersecurity Framework (CSF)NIST Special Publications (e.g., SP 800 series) for cybersecurity
ACM Standards and Guidelines -ACM Code of Ethics and Professional Conduct-ACM Computing Classification System (CCS) and its role in standardization

Total : 15 PERIODS

REFERENCES:

1. Manual for Standards Formulation 2022, Bureau of Indian Standards
2. Kunas, Michael, "Implementing service quality based on ISO/IEC 20000: A management guide" IT Governance publishing, 2012.
3. Kan, S. H. "Standards for Information Technology and Systems", Prentice Hall, 2017.
4. IEEE Computer Society. (2014) "IEEE Guide to the Software Engineering Body of Knowledge (SWEBOK)", Version 3.0. IEEE. Retrieved from IEEE Xplore
5. Calder, Alan. "ISO/IEC 27001:2013 – A Pocket Guide" IT Governance Publishing, 2013.
6. Sikos, Leslie, "Web Standards: Mastering HTML5, CSS3, and XML." Apress, 2011.
7. Association for Computing Machinery. "ACM Code of Ethics and Professional Conduct: A Guide" ACM, 2018
8. Calder, Alan, "NIST Cybersecurity Framework: A Pocket Guide. IT Governance Publishing" 2018.

COURSE OBJECTIVE:

The objective of the course is four-fold:

1. Development of a holistic perspective based on self-exploration about themselves (human being), family, society and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
3. Strengthening of self-reflection.
4. Development of commitment and courage to act.

MODULE I: INTRODUCTION**(3L,6P)**

Purpose and motivation for the course, recapitulation from Universal Human Values-I, Self-Exploration– 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.

Practical Session: Include 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

MODULE II: HARMONY IN THE HUMAN BEING**(3L,6P)**

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.

Practical Session: Include 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 program for ensuring health vs dealing with disease.

MODULE III: HARMONY IN THE FAMILY AND SOCIETY**(3L,6P)**

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.

Practical Session: Include sessions to reflect on relationships in family, hostel and institute as 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

MODULE IV: HARMONY IN THE NATURE AND EXISTENCE (3L,6P)

Understanding the harmony in the Nature, Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self regulation in nature, Understanding Existence as Co-existence of mutually interacting units in all- pervasive space, Holistic perception of harmony at all levels of existence.

Practical Session: Include 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.

MODULE V: IMPLICATIONS OF HARMONY ON PROFESSIONAL ETHICS (3L,6P)

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.

Practical Session: Include Exercises and Case Studies will be taken up in Sessions E.g. To discuss the conduct as an engineer or scientist etc.

TOTAL: 45 (15 Lectures + 30 Practicals) PERIODS

COURSE OUTCOME:

By the end of the course, the students will be able to:

1. Become more aware of themselves, and their surroundings (family, society, nature);
2. Have more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.
3. Have better critical ability.
4. Become sensitive to their commitment towards what they have understood (human values, human relationship and human society).
5. 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.

REFERENCES:

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 3rd revised edition, 2023.
2. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
3. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
4. The Story of Stuff (Book).
5. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
6. Small is Beautiful - E. F Schumacher.
7. Slow is Beautiful - Cecile Andrews.
8. Economy of Permanence - J C Kumarappa
9. Bharat Mein Angreji Raj - PanditSunderlal
10. Rediscovering India - by Dharampal
11. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
12. India Wins Freedom - Maulana Abdul Kalam Azad
13. Vivekananda - Romain Rolland (English)
14. Gandhi - Romain Rolland (English)

Web URLs:

1. Class preparations: <https://fdp-si.aicte-india.org/UHV-II%20Class%20Note.php>
2. Lecture presentations: https://fdp-si.aicte-india.org/UHV-II_Lectures_PPTs.php
3. Practice and Tutorial Sessions: <https://fdp-si.aicte-india.org/UHV-II%20Practice%20Sessions.php>

Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01						1	1	1	3			3
C02						1	1	1	3			3
C03						3	3	2	3		1	3
C04						3	3	2	3		1	3
C05						3	3	3	3		2	3