SRI KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution | Approved by AICTE | Affiliated to Anna University | Accredited by NAAC with A++ Grade
Kuniamuthur, Coimbatore – 641008
Phone : (0422)-2678001 (7 Lines) | Email : info@skcet.ac.in | Website : www.skcet.ac.in

Curriculum & Syllabi

Regulation 2025

M.E COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRI KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution | Approved by AICTE | Affiliated to Anna University | Accredited by NAAC with A++ Grade
Kuniamuthur, Coimbatore – 641008
Phone : (0422)-2678001 (7 Lines) | Email : info@skcet.ac.in | Website : www.skcet.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Batch 2025-2029)

VISION OF THE INSTITUTION

- To Produce Globally Competitive Engineers with High Ethical Values and Social Responsibilities.

MISSION OF THE INSTITUTION

- To impart the highest quality state-of-the-art technical education by providing impetus to innovation, research, and development and empowering students with entrepreneurship skills.
- To instill ethical values, imbibe a sense of social responsibility, and strive for societal well-being.
- To identify the needs of society and offer sustainable solutions through outreach programs.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

- To prepare professionals with high technical, research and entrepreneurial skills as well as ethical values who will contribute to the computational world.

MISSION OF THE DEPARTMENT

- To develop human resources with the ability and attitude to adapt to emerging technological changes through academic and research-oriented events.
- To identify current socio, economic problems of national and international significance and provide solutions through competency centers.
- To impart ethics, social responsibilities and necessary professional, entrepreneurial and leadership skills through student lead activities

I.	PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)
To enable graduates to	
PEO 1	Pursue higher education and research or have a successful career in industries associated with computer science and engineering, or succeed as entrepreneurs
PEO 2	Be adaptive to the growing needs of global computational environment by engaging in life-long learning

II.	PROGRAMME OUTCOMES (POs)
PO 1	**Engineering Knowledge**: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
PO 2	**Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.
PO 3	**Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.
PO 4	**Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.
PO 5	**Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems.
PO 6	**The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.
PO 7	**Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.
PO 8	**Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO 9	**Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
PO 10	**Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.
PO 11	**Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.

Knowledge and Attitude Profile (WK)

WK1	A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.
WK2	Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
WK3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.
WK4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
WK5	Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.
WK6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.
WK7	Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
WK8	Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
WK9	Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

III. PROGRAMME SPECIFIC OUTCOMES (PSOs)	
The Graduates of B.E – CSE programme will be able to:	
PSO 1	Apply the fundamental knowledge for problem solving and analysis as well as conduct investigations in computer science and engineering for sustainable development
PSO 2	Design and develop the solutions for real time problems and implement them by using modern software tools in lieu of deploying them in the society for its growth
PSO 3	Inculcate effective communication skills and ethics for lifelong learning

IV. MAPPING OF PEOs WITH POs and PSOs														
PEO	**Pos**											**PSO**		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	1	2	3
1	3	3	3	2	2	2	2	1	1	1	2	3	3	2
2	1	1	2	2	3	3	2	1	2	2	2	1	2	3
	1- low, 2 - medium, 3 - high, '-' - no correlation													

V. MAPPING OF PEOs WITH PSOs			
	PSO 1	**PSO 2**	**PSO 3**
PEO 1	3	3	2
PEO 2	1	2	3

AUTONOMOUS CURRICULA AND SYLLABI

Regulations 2025 (ME CSE)

SEMESTER I						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Theory (Internal 40 Marks & External 60 Marks)						
1	25PF101	Logic, Graph and Automata Theory	3 / 0 / 0	3	3	4,8,9
2	25PF102	Advanced Data Structures and Algorithms	3 / 0 / 0	3	3	4, 8, 9, 17
3	25PF103	Advanced Operating Systems	3 / 0 / 0	3	3	4, 8, 9, 12
4	25PF104	Cloud Computing and Management	3 / 0 / 0	3	3	4,5,8,10
5	25PF105	Machine Learning Approaches	3 / 0 / 0	3	3	4,8,9
6	25PF106	Research Methodology and IPR	3 / 0 / 0	3	3	
Practical (Internal 60 Marks & External 40 Marks)						
6	25PF107	Data Structures and Operating Systems Laboratory	0 / 0 / 4	4	2	4, 9, 11, 12
			Total	**22**	**20**	

SEMESTER II						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Theory (Internal 40 Marks & External 60 Marks)						
1	25ITC01	Modern Computer Architecture	3 / 1 / 0	4	4	4,8,9
Theory with Practical (Internal 50 Marks & External 50 Marks)						
2	25MSC01	Materials Science	2 / 0 / 2	4	3	3, 4, 7, 9, 11, 12, 13, 16
3	25IT201	Object Oriented Programming Using C++	1 / 0 / 4	5	3	4,8,9
4	25CSC01	Data Structures	3 / 0 / 2	5	4	4,9
5	25CY201	Applied Statistics Using Python	3 / 0 / 2	5	4	4,8,9

Practical (Internal 60 Marks & External 40 Marks)						
6	25CS202	Application Development Practices	0 / 0 / 4	4	2	4,9
Indian Knowledge System - Blended Learning (Internal 100 Marks)						
7	25TA201	Tamils and Technology	1 / 0 / 0	1	1	2, 7, 9, 11, 12, 15, 17
8	25IKC01	Introduction to Indian Knowledge System	2 / 0 / 0	2	2	3, 5, 7, 9, 11, 13, 15, 17
			Total	**30**	**24**	

SEMESTER III

S. No.	Course Code	Courses	L/T/P	Total Hours	Credits	SDG Mapping
Theory-Blended Learning (Internal 100 Marks)						
1	25GE301	Universal Human Values	3 / 0 / 0	3	3	3,4,5,8,16
Theory with Practical (Internal 50 Marks & External 50 Marks)						
2	25CYC01	Operating Systems	3 / 0 / 2	5	4	4, 9
3	25ADC01	Artificial Intelligence and Its Application	2 / 0 / 2	4	3	4,9,11
4	25IT301	Relational Database Management	3 / 0 / 2	5	4	4, 9
5	25CS301	Algorithm Design Techniques	3 / 0 / 2	5	4	4, 9
6	25CS302	Programming in Java	1 / 0 / 4	5	3	4,8,9
Indian Knowledge System - Blended Learning (Internal 100 Marks)						
7	25IK301	Course on Mathematical and Computer Logic for Nyaya Sastra Studies	2 / 0 / 0	2	2	3,4,5,8,10,16
Spoken Language (Internal 100 Marks)						
8	25MCC02	Multilingual Practices	0 / 0 /2	1	1	4,11,16
Total				**30**	**24**	

SEMESTER IV

S. No.	Course Code	Courses	L/T/P	Total Hours	Credits	SDG Mapping

Theory (Internal 40 Marks & External 60 Marks)						
1	25ITC02	Formal Languages and Automata Theory	3 / 1 / 0	4	4	4,8,9
2	25CYC01	Cybersecurity Essentials	3 / 0 / 0	3	3	4,9
Theory with Practical (Internal 50 Marks & External 50 Marks)						
3	25AMC01	Machine Learning Techniques	2 / 0 / 2	4	3	4,9
4	25CY402	Data Communication and Networks	2 / 0 / 2	4	3	4,9,11
5	25IOC01	Internet of Things	2 / 0 / 2	4	3	4,9
Practical (Internal 60 Marks & External 40 Marks)						
6	25CS401	Frontend Frameworks	0 / 0 / 4	4	2	4,9
7	25CS402	Backend Frameworks	0 / 0 / 4	4	2	4,9
Indian Knowledge System - Blended Learning (Internal 100 Marks)						
8	25IK401	Multilingual and Heritage Computing	2 / 0 / 0	2	2	4,8,9
Mandatory Course-Blended Learning (Internal 100 Marks)						
9	25MCC01	Environmental Sciences	1 / 0 / 0	1	0	3,4,5,6,7,8,11,13,15,16
Total				**30**	**22**	

SEMESTER V						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Theory (Internal 40 Marks & External 60 Marks)						
1	25CSC02	Mobile and Edge Computing	3 / 0 / 0	3	3	4,9
2	25CSC03	Quantum Computing	3 / 0 / 0	3	3	4,9
Theory with Practical (Internal 50 Marks & External 50 Marks)						
3	25CS502	Signal Processing	2 / 0 / 2	4	3	4,9
4		Professional Elective – I	2 / 0 / 2	4	3	
5		Professional Elective - II	2 / 0 / 2	4	3	
Practical (Internal 60 Marks & External 40 Marks)						
6	25IT505	Cloud Infrastructure and Services Management	0 / 0 / 4	4	2	4,9

Mini Project (Internal 100 Marks)						
7	25CS503	Application Development	0 / 0 / 6	6	3	3,4,8,9
Mandatory Course-Blended Learning (Internal 100 Marks)						
8	25MCC03	Indian Constitution	1 / 0 / 0	1	0	1,2,3,5,6,8,11
Total				**29**	**20**	

SEMESTER VI						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Theory-Blended Learning (Internal 100 Marks)						
1	25GE01	Entrepreneurship and Startups	3 / 0 / 0	3	3	3,4,5,8,9,17
Theory (Internal 40 Marks & External 60 Marks)						
2	25CS601	Advanced Graph Theory	3 / 0 / 0	3	3	4,9
3	25CS602	Parallel and Distributed Computing	3 / 0 / 0	3	3	4,7,9
Theory with Practical (Internal 50 Marks & External 50 Marks)						
4	25IT602	Mobile Application Development	3 / 0 / 2	5	4	9,17
5	25CS603	Principles of Compiler Design	3 / 0 / 2	5	4	4,9
6		Professional Elective – III	2 / 0 / 2	4	3	
7		Professional Elective - IV	2 / 0 / 2	4	3	
Practical (Internal 60 Marks & External 40 Marks)						
8	25CS604	Prototype Lab	0 / 0 / 2	2	1	4,8,9,17
Total				**29**	**24**	

SEMESTER VII						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Theory (Internal 40 Marks & External 60 Marks)						
1		Open / Emerging/ Industrial Elective- II	3 / 0 / 0	3	3	-
2		Open / Emerging/ Industrial Elective- II	3 / 0 / 0	3	3	-
3		Open / Emerging/ Industrial Elective- III	3 / 0 / 0	3	3	-
Theory with Practical (Internal 50 Marks & External 50 Marks)						

4		Professional Elective – V	2 / 0 / 2	4	3	-
5		Professional Elective - VI	2 / 0 / 2	4	3	-
Project (Internal 60 Marks & External 40 Marks)						
6	25CS701	Project - I	0 / 0 / 6	6	3	4,9,17
Internship (Internal 100 Marks)						
7	25CS702	Employability Enhancement Skills (Internship)	28 Days		2	8,12,17
Total				**23**	**20**	

SEMESTER VIII						
S. No.	**Course Code**	**Courses**	**L/T/P**	**Total Hours**	**Credits**	**SDG Mapping**
Project (Internal 60 Marks & External 40 Marks)						
1	25CS801	Project - II	0 / 0 / 24	24	12	4,9,17
Total Credits					**164**	

SCHEME OF CREDIT DISTRIBUTION – SUMMARY

Sl. No.	**Stream**	**Credits/Semester**								**C**	%
		I	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**		
1	Humanities & Social Sciences Including Management (HSMC)	4	1	3			3			**11**	6.7
2	Basic Sciences (BSC)	4	3							**7**	4.3
3	Engineering Sciences (ESC)	11	4	5	2					**22**	13.2
4	Professional Core (PCC)		15	15	20	11	14			**75**	46
5	Professional Electives (PEC)					6	6	6		**18**	11
6	Open/Emerging/Industry (OEC)							9		**9**	5.4
7	Project Work (PROJ)					3	1	5	12	**21**	13
8.	Mandatory Course (MC) / Spoken Hindi			1						**1**	0.6
Total		**19**	**23**	**24**	**22**	**20**	**24**	**20**	**12**	**164**	

STRUCTURE FOR UNDERGRADUATE ENGINEERING PROGRAM

S. No.	Course Work - Subject Area	AICTE Suggested Credits	SKCET Credits (CSE)
1.	Humanities and Social Sciences (HS), including Management;	16	11
2.	Basic Sciences (BS) including Mathematics, Physics, Chemistry, Biology;	23	7
3.	Engineering Sciences (ES), including Materials, Workshop, Drawing, Basics of Electrical/Electronics/Mechanical/Computer Engineering, Instrumentation;	29	22
4.	Professional Subjects-Core (PC), relevant to the chosen specialization/branch; (May be split into Hard (no choice) and Soft (with choice), if required	59	75
5.	Professional Subjects – Electives (PE), relevant to the chosen specialization/ branch;	12	18
6.	Open Subjects- Electives (OE), from other technical and/or emerging subject areas;	9	9
7.	Project Work, Seminar and/or Internship in Industry or elsewhere.	15	21
8.	Mandatory Courses (MC)	Non-credit	1
Total		163	164
Minor Variations is allowed as per need of the respective disciplines			

HUMANITIES & SOCIAL SCIENCES INCLUDING MANAGEMENT (11 Credits)						
SL. No.	Course Code	Course Title	L/T/P	Contact hrs./Wk.	C	SDG Mapping
1.	25ENC01	Business Communication and Value Science	2 / 0 / 2	4	3	4,5,8,10
2.	25TA101	Heritage of Tamils	1 / 0 / 0	1	1	2, 7, 9, 12, 15, 16
3.	25TA201	Tamils and Technology	1 / 0 / 0	1	1	2, 7, 9, 11, 12, 15, 17
4.	25GE301	Universal Human Values	3 / 0 / 0	3	3	3,4,5,8,16
5.	25GE01	Entrepreneurship and Startups	3 / 0 / 0	3	3	3,4,5,8,9,17

BASIC SCIENCE COURSES (07 Credits)

SL. No.	Course Code	Course Title	L/T/P	Contact hrs./Wk.	C	SDG Mapping
1.	25MA102	Mathematical Foundations for Computer Science	3 / 1 / 0	4	4	4,8,9
2.	25MS101	Materials Science	2 / 0 / 2	4	3	3, 4, 7, 9, 11, 12, 13, 16

ENGINEERING SCIENCE COURSES (22 Credits)						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25CS101	Problem Solving and Programming Paradigms	3 / 0 / 0	3	3	4, 8, 9, 17
2.	25CS102	Digital Principles and System Design	3 / 1 / 0	4	4	4, 8, 9, 12
3.	25CS103	C Programming	1 / 0 / 4	5	3	4,8,9
4.	25ME203	Design Thinking and Idea Lab	0 / 0 / 2	2	1	4, 9, 11, 12
5.	25IKC01	Introduction to Indian Knowledge Systems	2 / 0 / 0	2	2	3, 5, 7, 9, 11, 13, 15, 17
6.	25CS202	Application Development Practices	0 / 0 / 4	4	2	4,9
7.	25CS302	Programming in Java	1 / 0 / 4	5	3	4,8,9
8.	25IK301	Course on Mathematical and Computer Logic for Nyaya Sastra Studies	2 / 0 / 0	2	2	3,4,5,8,10,16
9.	25IK401	Multilingual and Heritage Computing	2 / 0 / 0	2	2	4,8,9

PROFESSIONAL CORE COURSES (75 Credits)

SL. No.	Course Code	Course Title	L/T/P	Contact hrs./Wk.	C	SDG Mapping
1.	25ITC01	Modern Computer Architecture	3 / 1 / 0	4	4	4,8,9
2.	25IT201	Object Oriented Programming Using C++	1 / 0 / 4	5	3	4,8,9
3.	25CSC01	Data Structures	3 / 0 / 2	5	4	4,9
4.	25CY201	Applied Statistics Using Python	3 / 0 / 2	5	4	4,8,9
5.	25CYC01	Operating Systems	3 / 0 / 2	5	4	4, 9
6.	25ADC01	Artificial Intelligence and Its Application	2 / 0 / 2	4	3	4,9,11
7.	25IT301	Relational Database Management	3 / 0 / 2	5	4	4, 9
8.	25CS301	Algorithm Design Techniques	3 / 0 / 2	5	4	4, 9
9.	25ITC02	Formal Languages and Automata Theory	3 / 1 / 0	4	4	4,8,9
10.	25CYC01	Cybersecurity Essentials	3 / 0 / 0	3	3	4,9
11.	25AMC01	Machine Learning Techniques	2 / 0 / 2	4	3	4,9
12.	25CY402	Data Communication and Networks	2 / 0 / 2	4	3	4,9,11
13.	25IOC01	Internet of Things	2 / 0 / 2	4	3	4,9
14.	25CS401	Frontend Frameworks	0 / 0 / 4	4	2	4,9
15.	25CS402	Backend Frameworks	0 / 0 / 4	4	2	4,9
16.	25CSC02	Computer Vision	3 / 0 / 0	3	3	4,9
17.	25CSC03	Quantum Computing	3 / 0 / 0	3	3	4,9
18.	25CS501	Principles of Compiler Design	3 / 0 / 2	5	4	4,9
19.	25CS502	Signal Processing	2 / 0 / 2	4	3	4,9
20.	25ADC02	Data Analytics	2 / 0 / 2	4	3	4,9
21.	25CS601	Mobile and Edge Computing	3 / 0 / 0	3	3	4,9
22.	25CS602	Advanced Graph Theory	3 / 0 / 0	3	3	4,9
23.	25CS603	Parallel and Distributed Computing	3 / 0 / 0	3	3	4,7,9
24.	25IT602	Mobile Application Development	3 / 0 / 2	5	4	9,17

PROFESSIONAL ELECTIVE COURSES (18 Credits)

S. No.	Course Code	Course Title	L/T/P	Contact hrs./Wk.	C	Ext/ Int	Cat.
ELECTIVE STREAM I – MACHINE LEARNING ENGINEERING							
1	25AD902	Exploratory Data Analysis (EDA) using Python	2/0/2	4	3	50/50	PEC
2	25AD905	Statistical Methods and Basic Machine Learning Models	2/0/2	4	3	50/50	PEC
3	25AM901	Advanced ML Techniques	2/0/2	4	3	50/50	PEC
4	25AM902	NLP	0/0/6	6	3	40/60	PEC
5	25AM903	Computer Vision	0/0/6	6	3	40/60	PEC
6	25AD906	GenAI Advanced Prompt Engineering & LLMs	0/0/6	6	3	40/60	PEC
ELECTIVE STREAM II - DATA ANALYST WITH ML ESSENTIALS							
1	25AD901	Data Storytelling and Visualization	0/0/6	6	3	40/60	PEC
2	25AD902	Exploratory Data Analysis (EDA) using Python	2/0/2	4	3	50/50	PEC
3	25AD903	Problem solving with Analytical & Design Thinking	2/0/2	4	3	50/50	PEC
4	25AD904	PowerBI	0/0/6	6	3	40/60	PEC
5	25AD905	Statistical Methods and Machine Learning Models	2/0/2	4	3	50/50	PEC
6	25AD906	GenAI Advanced Prompt Engineering & LLMs	0/0/6	6	3	40/60	PEC
ELECTIVE STREAM III - CLOUD IT ADMINISTRATION							
1	25CS901	Implementing and Administering Enterprise Networks	2/0/2	4	3	50/50	PEC
2	25CS902	Linux System Administration	2/0/2	4	3	50/50	PEC
3	25CS903	Information Security Systems	2/0/2	4	3	50/50	PEC
4	25CS904	Low-Code No-Code Application Building	0/0/6	6	3	40/60	PEC
5	25CS905	Virtualization, Cloud Computing and SysOps	0/0/6	6	3	40/60	PEC

6	25CS906	Continuous Monitoring and Observability (AWS)	0/0/6	6	3	40/60	PEC
ELECTIVE STREAM IV – CYBER SECURITY							
1	25CS901	Implementing and Administering Enterprise Networks	2/0/2	4	3	50/50	PEC
2	25CS902	Linux System Administration	2/0/2	4	3	50/50	PEC
3	25CS903	Information Security Systems	2/0/2	4	3	50/50	PEC
4	25CY901	Cloud Computing and Containerized Virtual Infrastructure	0/0/6	6	3	40/60	PEC
5	25CY902	Penetration Testing	0/0/6	6	3	40/60	PEC
6	25CY903	Security Operations of Information systems	0/0/6	6	3	40/60	PEC
ELECTIVE STREAM V – FULL STACK SOFTWARE DEVELOPMENT (for Minor)							
1	25CS911	Managing and Querying Database (RDBMS) MySQL / Postgre SQL	2/0/2	4	3	50/50	PEC
2	25CS912	Java / Python: Object-Oriented Programming	0/0/6	6	3	40/60	PEC
3	25IT911	Web Development and UI/UX Essentials	0/0/6	6	3	40/60	PEC
4	25IT912	Build Single-Page Applications using React	2/0/2	4	3	50/50	PEC
5	25IT913	Build Back-end Application using Spring Boot / FAST API	2/0/2	4	3	50/50	PEC
6	2AD906	GenAI Advanced Prompt Engineering & LLMs	0/0/6	6	3	40/60	PEC

INDIAN KNOWLEDGE SYSTEM (11 Credits)						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25TA101	Heritage of Tamils	1 / 0 / 0	1	1	2, 7, 9, 12, 15, 16
2.	25MC101	Induction Programme			0	3,4,5,10,16
3.	25TA201	Tamils and Technology	1 / 0 / 0	1	1	2, 7, 9, 11, 12, 15, 17
4.	25IKC01	Introduction to Indian Knowledge Systems	2 / 0 / 0	2	2	3, 5, 7, 9, 11, 13, 15, 17

5.	25IK301	Course on Mathematical and Computer Logic for Nyaya Sastra Studies	2 / 0 / 0	2	2	3,4,5,8,10,16
6.	25MCC02	Multilingual Practices	0 / 0 /2	2	1	4,11,16
7.	25IK401	Multilingual and Heritage Computing	2 / 0 / 0	2	2	4,8,9
8.	25MCC01	Environmental Sciences	1 / 0 / 0	1	0	3,4,5,6,7,8,11,13,15,16
9.	25MCC03	Indian Constitution	1 / 0 / 0	1	0	1,2,3,5,6,8,11
10.	25CS702	Employability Enhancement Skills (Internship)	28 Days		2	8,12,17

OPEN/ EMERGING / INDUSTRY ELECTIVE						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25CS001	Programming with Data Structures	2/ 0/ 2	4	3	4,9

PROJECT WORK (23 Credits)						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25IT505	Cloud Infrastructure and Services Management	0 / 0 / 4	4	2	4,9
2.	25CS503	Application Development	0 / 0 / 6	6	3	3,4,8,9
3.	25CS604	Prototype Lab	0 / 0 / 2	2	1	4,8,9,17
4.	25CS701	Project - I	0 / 0 / 6	6	3	4,9,17
5.	25CS702	Employability Enhancement Skills (Internship)	28 Days		2	8,12,17
6.	25CS801	Project - II	0 / 0 / 24	24	12	4,9,17

PROFESSIONAL ELECTIVE COURSES: VERTICALS				
Vertical I - ML Engineering	**Vertical II - Data Analyst with ML Essentials**	**Vertical III - Cloud IT Administration**	**Vertical IV - Cybersecurity**	**Vertical V – Full Stack Software Development (Minor)**
Exploratory Data Analysis (EDA) using Python	Data Storytelling and Visualization	Implementing and Administering Enterprise Networks	Implementing and Administering Enterprise Networks	Managing and Querying Database (RDBMS) MySQL / Postgre SQL
Statistical Methods and Basic Machine Learning Models	Exploratory Data Analysis (EDA) using Python	Linux System Administration	Linux System Administration	Java / Python: Object-Oriented Programming
Advanced ML Techniques	Problem solving with Analytical & Design Thinking	Information Security Systems	Information Security Systems	Web Development and UI/UX Essentials
NLP	PowerBI	Low-Code No-Code Application Building	Cloud Computing and Containerized Virtual Infrastructure	Build Single-Page Applications using React
Computer Vision	Statistical Methods and Machine Learning Models	Virtualization, Cloud Computing and SysOps	Penetration Testing	Build Back-end Application using Spring Boot / FAST API
GenAI Advanced Prompt Engineering & LLMs	GenAI Advanced Prompt Engineering & LLMs	Continuous Monitoring and Observability (AWS)	Security Operations of Information systems	GenAI Advanced Prompt Engineering & LLMs

INTERN (02 Credits)						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25CS702	Employability Enhancement Skills (Internship)	28 Days		2	8,12,17

VALUE ADDED COURSES (Based on student's interest)							
S. No	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**	**Sem**
1.	25VA101	PEGA Certification	1 / 0 / 0	1	1	4, 9,12	III
2.	25VA102	Amazon Web Service (AWS)	1 / 0 / 0	1	1	4, 9,12	IV
3.	25VA103	Google Certified Cloud Professional (GCCP)	1 / 0 / 0	1	1	4, 9,12	V
4.	25VA104	CISCO Certification	1 / 0 / 0	1	1	4, 9,12	VI
5.	25VA105	Wipro PRP Certification	1 / 0 / 0	1	1	4, 9,12	VI

MANDATORY COURSES (Courses conducted either by internal faculty or through MOOCs)						
SL. No.	**Course Code**	**Course Title**	**L/T/P**	**Contact hrs./Wk.**	**C**	**SDG Mapping**
1.	25MC101	Induction Programme			0	3,4,5,10,16
2.	25MCC02	Multilingual Practices	0 / 0 /2	2	1	4,11,16
3.	25MCC01	Environmental Sciences	1 / 0 / 0	1	0	3,4,5,6,7,8,11,13,15,16
4.	25MCC03	Indian Constitution	1 / 0 / 0	1	0	1,2,3,5,6,8,11

L: Lecture **T:** Tutorial **P:** Practical **C:** Credit **O:** Outside Class hours **Cat.:** Category

HSMC : Humanities and Social Sciences including Management
BSC : Basic Science Courses
ESC : Engineering Science Courses
PCC : Professional Core Courses
PEC : Professional Elective Courses
OEC : Open and Emerging Elective Courses
PRJ : Project Work
INT : Internship
MC : Mandatory Course

Definition of Credit:

L – Lecture	1 Hr. Lecture (L) per week	1 credit
T – Tutorial	1 Hr. Tutorial (T) per week	1 credit
P - Practical/Practice (Project and Industry based Courses)	1 Hr. Practical (P) per week	0.5 credit

SEMESTER – I

25MA102 SDG No. 4,8,9	MATHEMATICAL FOUNDATION OF COMPUTER SCIENCE (Common to all Computing Sciences)	L	T	P	C
		3	1	0	4

COURSE OBJECTIVES

1. To Construct and interpret truth tables to determine the validity of logical expressions, including tautologies, contradictions, and logical equivalences.
2. To learn the different types of relations and functions.
3. To apply fundamental concepts of algebraic structures, including groups and their properties.
4. To evaluate definite integrals using Bernoulli's formula and find the derivatives.
5. To acquire skills in probability and Standard distributions.

PROPOSITIONAL CALCULUS **12**

Basic concepts – Propositions – Connectives – Truth tables – Tautologies and contradictions – Contrapositive – Logical equivalences and implications – Normal forms – Rules of inference – Predicates – Free and bound variables – Quantifiers – Validity of arguments. Case Study: Propositional logic in AI to automate logical reasoning.

RELATIONS AND FUNCTIONS **12**

Relations – Relational matrix and the graph of a relation – Equivalence relations – Partial ordering relation. Functions - Classification of functions – Composition and inverse of functions – Permutation functions. Case Study: Building a recommendation system using functional mappings and graphs.

ALGEBRAIC STRUCTURES **12**

Groups - Permutation group - Cyclic group - Properties - Subgroup- Group homomorphism - Properties - Coding theory - Group code - Hamming codes - Error correction using matrices - Error correction - Decoding group codes. Case Study: Applications of group theory in cryptography.

CALCULUS **12**

Limit of a function - Continuity - Derivatives - Differentiation rules - Partial derivatives – Total derivative – Differentiation of implicit functions – Jacobian - Taylor's series for functions of two variables - Definite integrals - Bernoulli's formula - Double integration in Cartesian coordinates –-Triple integration in Cartesian coordinates – Case Study: Evaluation of Beta and Gamma integrals using MATLAB.

PROBABILITY **12**

Axioms of probability – Conditional probability – Baye's theorem - Discrete and continuous random variables – Probability function - Moment generating functions – Binomial, Poisson and Normal distributions. Case Study: Modeling exam scores using the Normal Distribution.

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Construct mathematical arguments using logical connectives, quantifiers and verify the correctness of an argument using symbolic logic, truth tables.	[AP]
CO2	Apply the concepts of set theory and its operations in data structures and mathematical modelling languages	[AP]
CO3	Inculcate the curiosity for applying the concepts of algebraic structures to coding theory.	[AP]
CO4	Estimate derivatives and integrals to compute probabilities and expectations in statistical models.	[U]
CO5	Implement the concepts of probability and random variables in modeling and analysis	[AP]

TEXTBOOKS

1. Kenneth H. Rosen., “Discrete Mathematics and Its Applications”, 8th Edition, McGraw Hill publications, 2019.
2. Tremblay T.P, and Manohar. R, "Discrete Mathematical Structures with application to Computer Science”, Tata McGraw Hill, 2017.
3. Kreyszig. E, “Advanced Engineering Mathematics” 10th Edition, John Wiley and Sons (Asia) Limited, Singapore, 2020.

REFERENCE BOOKS

1. Susanna S. Epp, “Discrete Mathematics with Applications”, 4th Edition, Brooks/Cole, Cengage Learning, 2010.
2. Miller. I.R, Freund. J.E and Johnson. R, “Probability and Statistics for Engineers, Pearson Education, Asia, 9th Edition, 2016.
3. Ralph P.Grimaldi, “Discrete and Combinatorial Mathematics, Pearson, 2006.

WEB RESOURCES

1. https://mathworld.wolfram.com/AbstractAlgebra.html
2. https://www.coursera.org/specializations/discrete-mathematics
3. https://ocw.mit.edu/courses/6-042j-mathematics-for-computer-science-fall-2010/
4. https://www.coursera.org/learn/integration-calculus
5. https://archive.nptel.ac.in/courses/111/105/111105090/

25CS101 SDG NO. 4, 8	PROBLEM SOLVING AND PROGRAMMING PARADIGMS	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

1. To familiarize the fundamental concepts of problem-solving techniques and differences between Computational Thinking and Logical Thinking
2. To introduce the different programming paradigms and their concepts
3. To understand how to create a hybrid programming paradigm.
4. To study the suitability of programming paradigms for solving complex computational tasks.
5. To develop algorithmic thinking and efficiency analysis skills for solving real-world computational problems.

PROBLEM SOLVING TECHNIQUES **9**

Problem solving aspect - Top-down design - Implementation of algorithms- Program Verification - Efficiency - Analysis of Algorithms – Fundamental Algorithms - Swapping - counting - Factorial - Reversing the digits - Base conversion Algorithms. Computational Thinking - Logical Thinking - Flowcharts- Algorithmic thinking - Characteristics of algorithms - Pseudo code - Example problems.

IMPERATIVE PROGRAMMING PARADIGM **9**

Basics of Programming Paradigms, Expression Semantics, Data Semantics, Imperative Design principles – Structured, Turing Complete, Modularity, Procedural abstraction, Structured Theorem - Sequencing, Selection, Iteration and Recursion.

SEQUENTIAL PROGRAMMING PARADIGM **9**

Procedural Programming Paradigm: Procedures, Parameter passing, Procedural abstraction, Case Study - C Programming. Object Oriented Programming Paradigm: OOP design principles - Groupings of Data and Operations, Encapsulation, Data Abstraction, Inheritance and Polymorphism, Exceptions, Case Study - Object Oriented Programming Paradigm using Java Programming.

CONCURRENT & SCRIPTING PROGRAMMING PARADIGM **9**

Concurrent Programming Paradigm: Concurrency design with interleaving of process, Safe Access to Shared Data – Semaphore, Liveness property of concurrent programs - Case Study- Concurrency in ADA. Scripting Paradigm: – Basics of Web design language and Scripting Language, Case Study: JavaScript – Syntax of writing JavaScript, Variables and Functions.

FUNCTIONAL & LOGIC PROGRAMMING PARADIGM **9**

Functional Programming Paradigm: Declarative programming design – Functional and logic paradigm, Basics of Functional programming paradigm, Expressions – Syntax, Parsing, Types and Values, Assigning names to expressions, Lambda Calculus fundamentals, Function Abstraction and Recursive Functions - Case Study Haskell. The Logic Programming Paradigm:

Clauses and Predicates, Operations and Arithmetic, List and Operations, Unification and Backtracking.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Explain fundamental problem-solving strategies and algorithmic principles such as top-down design, program verification, and computational thinking.	[U]
CO2	Apply simple algorithms to solve structured problems.	[AP]
CO3	Develop modular programs using imperative, sequential and Object Oriented programming paradigms.	[AP]
CO4	Implement simple concurrent and scripting programs by applying concurrency design concepts and scripting language syntax.	[AP]
CO5	Understand the principles of functional and logic programming paradigms in solving problems using recursive functions, lambda expressions, unification, and backtracking.	[U]

TEXT BOOKS

1. Rebert W Sebesta, "Concepts of Programming Languages", 12th Edition, Pearson Edition, 2019.
2. Ravi Sethi, “Programming Languages: Concepts and Constructs”, 2nd Edition, Pearson India, 2006.
3. Seyed Mohamed Buhari, “Principles of Programming Languages: A Paradigm approach”, McGraw-Hill, 2011.
4. Michael L. Scott, “Programming Language Pragmatics”, 4th Edition, Morgan Kaufmann, 2016.

REFERENCE BOOKS

1. Seyed H. Roosta, “Foundations of Programming Languages: Design and Implementation”, 1st Edition, Cengage Learning, 2003.
2. David A. Watt, “Programming Language Design Concepts”, 1st Edition, John Wiley & Sons Ltd, 2004.
3. Minh Quang Tran, “The Art of Functional Programming Paradigm”, 1st Edition, Pragmatic Bookshelf, 2022.
4. Alejandro Serrano Mena, “Practical Haskell: A Real-World Guide to Functional Programming”, 3rd Edition Apress Media, 2022.

WEB RESOURCES

1. https://www.freecodecamp.org/news/an-introduction-to-programming-paradigms/
2. https://www.geeksforgeeks.org/introduction-of-programming-paradigms/ language/
3. https://archive.nptel.ac.in/courses/106/102/106102067/
4. https://www.decipherzone.com/blog-detail/programming-paradigms
5. https://hackr.io/blog/programming-paradigms
6. https://www.docsity.com/en/lecture-notes/computer-science/programming-paradigms/

25CS102 SDG NO. 9, 17	DIGITAL PRINCIPLES AND SYSTEM DESIGN	L	T	P	C
		3	1	0	4

COURSE OBJECTIVES

1. To acquire knowledge on Boolean algebra and simplification of Boolean functions.
2. To design and analyze various combinational logic circuits.
3. To familiarize the basics and design of sequential logic circuits.
4. To design and analyze clocked sequential circuits.
5. To understand the basics of memory devices and implementation using PLD's.

INTRODUCTION TO DIGITAL LOGIC **12**

Boolean Algebra, Canonical and Standard Forms, Simplification of Boolean functions. Logic Gates, Gate-Level Minimization, K Map Method, Product of Sums and Sum of Products Simplification, NAND and NOR Implementation. Introduction to Verilog HDL. Boolean functions and Digital Logic Gates Simulation using Virtual Lab.

COMBINATIONAL LOGIC CIRCUITS **12**

Design Procedure, Half Adder, Full Adder, Half Subtractor, Full Subtractor, Decoders, Encoders, Multiplexers, De-multiplexers, Implementation of Boolean functions using Decoder, Multiplexer and Demultiplexer. Modeling of basic combinational logic circuits using Verilog HDL and Virtual Lab.

SEQUENTIAL LOGIC CIRCUITS **12**

Latches, Flip-Flops - SR, D, JK & T, Shift Registers - SISO, SIPO, PISO, PIPO, Applications of shift Register - Johnson, Ring Counter - Design of 3 bit synchronous counter, Asynchronous counter, Modulo-n counter. Simulation of sequential logic circuits using Virtual Lab.

CLOCKED SEQUENTIAL LOGIC CIRCUITS **12**

Clocked sequential Circuits –Mealy Model, Moore Model, Analysis and design of Clocked Sequential Circuits using D,JK and T Flip flops – State Equations, State table, State Assignment, State Reduction, State Diagram.

PROGRAMMABLE LOGIC DEVICES **12**

Programmable Logic Devices: PLA, PAL, CPLD, SPLD, FPGA Generic Architecture. Implementation of combinational Logic functions using PLA, PAL.

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Demonstrate knowledge on canonical forms and minimization of Boolean functions using Karnaugh map.	[AP]
CO2	Design of various combinational logic circuits and simulation using Verilog HDL.	[AP]
CO3	Design of various sequential logic circuits	[A]
CO4	Apply synchronous sequential logic for state reduction and assignment.	[AP]
CO5	Implement logic functions using programmable logic devices (PLDs)	[A]

TEXT BOOKS

1. M.Morris Mano, Michael D. Ciletti," Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog", 6th Edition, Pearson education, 2018.
2. Ming-Bo Lin, "Digital Systems Design and Practice: Using Verilog HDL and FPGAs", 2nd Edition, Createspace Independent Pub, 2015.
3. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", 2nd Edition, Prentice Hall of India, 2009.

REFERENCE BOOKS

1. Jr. Roth, Charles H., Larry L. Kinney, "Fundamentals of Logic Design",7th Edition, CI-Engineering, 2013.
2. John F Wakerly, "Digital Design: Principles and Practices", 4th Edition, Pearson Education, 2008.
3. Donald D.Givone, "Digital Principles and Design", 1st Edition, McGraw Hill Education, 2017.

WEB REFERENCES

1. https://www.coursera.org/learn/digital-systems
2. https://digitalsystemdesign.in/digital-system-design-basics/
3. https://da-iitb.vlabs.ac.in/List%20of%20experiments.html
4. https://dec-iitkgp.vlabs.ac.in/List%20of%20experiments.html

25ENC01 SDG NO. 4,8,9,16	BUSINESS COMMUNICATION AND VALUE SCIENCE	L	T	P	C
		2	0	2	3

COURSE OBJECTIVES

1. Recognize the significance of life skills and their role in leading a happy and well-balanced life.
2. Motivate students to look within and create a better version of self.
3. Introduce the key concepts of values, life skills and business communication.
4. Apply clarity, conciseness, and coherence in workplace communication to ensure effective message delivery.
5. Identify best practices for communication in teamwork and leadership scenarios.

INTRODUCTION TO VALUES AND COMMUNICATION **6**

Introduction to values-Its importance and necessity–Overview of business communication (Importance of oral & written communication) - Listening skills (Hearing Vs Listening) – Body language- SATORI (sharing personal experience)

WORKPLACE COMMUNICATION ESSENTIALS **6**

Tenses – Subject-verb agreement – Voice -Summary writing – Story writing –Email writing - Its advantages & disadvantages- Business etiquette and protocol across cultures- Cross cultural leadership and conflict resolution-Virtual communication (Dos and Don'ts)-AI application in business communication-Meeting summaries and Content creation.

EFFECTIVE COMMUNICATION SKILLS **6**

Effective communication-Types of communication (Verbal, Written & Non-verbal communication) – Barriers to effective communication – Tips to develop communication skills-Branding and its types- Introduction to basic presentation skills- Storytelling in Writing and Presentations.

LISTENING AND READING SKILLS **6**

Principles of listening –The Process of listening–Types of listening-- – Pronunciation and enunciation- Speed Reading - Introduction to skimming and scanning – Reading Reviews (Book blurb) - Reading and understanding technical articles.

LIFE SKILLS **6**

Analyzing personality traits- Dr.MeredithBelbin and his research on teamwork - Belbin's 9 Team Roles and Lindgren's Big 5 personality traits - Belbin's 9 team player styles- Diversity & Inclusion- Different forms of Diversity in our society- Life skills –Importance and necessity – Thinking skills – Social skills – Emotional skills – Howard Gardner's Multiple Intelligence– Embracing adversity.

TOTAL (THEORY): 30 PERIODS

Lab Components

1. Listening skills- British Council B2&C1
2. Listeningto technical lectures (Audio/Video)
3. Immersion activity
4. Create resume
5. Group assignment
6. Trek followed by project
7. Group activities
8. Record a conversation
9. Group case study
10. News room buzz

TOTAL (LAB): 30 PERIODS

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have the ability to

CO1	Comprehend the importance of life skills and values.	[U]
CO2	Recognize own strengths and opportunities of Business Communication.	[R]
CO3	Analyze the life skills in different fields.	[A]
CO4	Apply the basic tenets of communication in workplace.	[AP]
CO5	Analyze the basic communication practices in different types of situations.	[A]

TEXT BOOKS:

1. Floyd Kory, “Interpersonal Communication”, McGraw Hill Publication, 2023.
2. BharadwajApoorva, “Leadership Communication Skills for Intercultural Management: Strategies for Effective Intercultural Management (Contemporary Themes in Business and Management)”, Routledge India; 1st edition, 2024.
3. Hasan Mehdi, “Win Every Argument: The Art of Debating, Persuading, and Public Speaking”, Macmillan Publisher, 2023.

REFERENCE BOOKS:

1. Helen Spencer-Oatey and DomnaLazidou**, "**Making Working Relationships Work: The TRIPS Toolkit for Handling Relationship Challenges and Promoting Rapport",Castledown Publishers,2023.
2. Dr.Praveen Sam and K N Shoba - A Course in Technical English by Cambridge University press, 2020.
3. C. Gangalakshmi, B. Rathika, L. Saranraj, "Professional English for Engineers", Cengage India, 2023 (2nd Edition).

WEB RESOURCES:

1. Train your mind to perform under pressure- Simon Sinek https://curiosity.com/videos/simon-sinek-on-training-your-mind-to-perform-under-pressure-capture-your-flag/
2. Brilliant way one CEO rallied his team in the middle of layoffs https://www.inc.com/video/simon-sinek-explains-why-you-should-put-people-before-numbers.html
3. Will Smith's Top Ten rules for success https://www.youtube.com/watch?v=bBsT9omTeh0
4. https://www.coursera.org/specializations/business-english
5. https://www.coursera.org/specializations/effective-business-communication

25CS103 SDG NO. 4, 9	C PROGRAMMING	L	T	P	C
		1	0	4	3

COURSE OBJECTIVES:

1. To understand the basics of C language, including syntax, data types, variables, constants, operators, and control structures.
2. To familiarize the fundamental logic to write, debug, and optimize code using conditional statements, loops, and functions to solve real-life problems.
3. To acquire the skills to manipulate and store data using arrays, strings, pointers, and dynamic memory allocation.
4. To learn how to implement structures, unions, and file handling to create modular and scalable programs.
5. To familiarize the development of small projects to apply their learning in practical, real-world scenarios by adhering to best programming practices.

BASICS OF C PROGRAMMING AND CONTROL STRUCTURES **3**

Introduction to C Programming - Structure of a C Program, Keywords, Identifiers, Constants, Variables, Basic Data Types, Type Conversion, Input & Output Functions (printf(), scanf(), getchar(), putchar()). Operators in C - Arithmetic, Relational, Logical, Assignment, Bitwise Operators, Operator Precedence, Associativity, and Typecasting.

CONTROL FLOW AND LOOPING **3**

Control Structures - Decision-Making: if, if-else, nested if-else, switch-case, Looping: for, while, do-while, Loop Control Statements: break, continue, goto.

ARRAYS, STRINGS, AND POINTERS **3**

Arrays - Defining, Declaring, Initializing, Accessing Arrays, Multi-Dimensional Arrays (2D Arrays and Matrix Operations). Strings - Character Arrays, String Handling Functions (strlen(), strcpy(), strcmp(), strcat()), String Manipulation.

POINTERS, FUNCTIONS AND RECURSION **3**

Pointers and Dynamic Memory Allocation - Basics of Pointers, Pointer Arithmetic, Pointer to Functions, Dynamic Memory Allocation: malloc(), calloc(), free(), realloc(), Dangling Pointers, NULL Pointers. Functions - User-defined and Library Functions, Parameter Passing: Call by Value, Call by Reference, Recursion.

STRUCTURES AND FILE HANDLING **3**

Structures and Unions - Declaring and Accessing Structures, Array of Structures, Nested Structures, Structures vs Unions. File Handling and Command-Line Arguments - File I/O Functions: fopen(), fclose(), fread(), fwrite(), fprintf(), fscanf(), Text Files and Binary Files, Command-line Arguments. Bitwise Operators and Preprocessor Directives, Bit Manipulation, Bitwise AND, OR, XOR, Left Shift, Right Shift. Preprocessor Directives: #define, #include, Macros, Conditional Compilation.

TOTAL (THEORY): 15 PERIODS

LAB EXPERIMENTS

1. Programs illustrating Input/output operations, arithmetic operators

Sample - Billing System for a Retail Store

Scenario: A retail store wants to automate its billing system. Write a C program that takes item names, quantities, and prices as input and calculates the total bill, including tax.
Concepts Used: Input/output operations, arithmetic operators, and control structures.

2. Programs illustrating Control Structures - Decision Making

Sample - Movie Ticket Booking System

Scenario: A multiplex offers discounts on movie tickets based on age. Write a program to accept the **age** of the user and display the ticket price using the following conditions:

- Children (below 12) - ₹100
- Seniors (above 60) - ₹150
- Others - ₹250

Concepts: if-else, Logical Operators

3. Programs illustrating Control Structures - Iteration

Sample - ATM PIN Validation

Scenario: A banking system allows users a maximum of three attempts to enter the correct PIN. Implement a program that asks the user for a 4-digit PIN. If the entered PIN is incorrect, the program should allow up to three attempts before locking the user out.

Concepts: while loop, Conditional statements

4. Programs illustrating Arrays

Sample - Smart Attendance Tracker

Scenario: A school wants to track student attendance. Develop a C program that allows input of student roll numbers and marks attendance. The program should display present and absent students.
Concepts Used: Arrays, loops, and conditional statements.

5. Programs illustrating String Manipulation

Sample - User Authentication System

Scenario: A company wants a basic user login system. Write a program that accepts a username and password from the user and checks if they match a pre-stored username and password. If they match, print "Login Successful"; otherwise, print "Invalid Credentials".

Concepts: strcmp(), gets(), puts(), Input Validation

6. Programs illustrating Functions

Sample - Banking Transaction System

Scenario: A bank needs an application to handle transactions. Write a program that allows users to deposit, withdraw, and check their account balance. Ensure that withdrawal doesn't exceed the account balance.
Concepts Used: Functions, conditional statements, and loops.

7. Programs illustrating Pointers and Dynamic Memory Allocation

Sample - Dynamic Array Allocation

Scenario: A **data-processing application** needs to store n numbers dynamically.
Task: Write a program that takes n as input, dynamically allocates memory for an array, accepts n numbers, and prints them.
Concepts: malloc(), free(), Pointer arithmetic

8. Programs illustrating Structures and Unions

Sample - Online Shopping Cart

Scenario: An e-commerce website wants a shopping cart system. Write a program that allows users to add products to the cart, calculate the total price, and apply discounts for orders above a certain amount.
Concepts Used: Arrays, structures, and conditional statements.

9. Programs illustrating File Handling

Sample - File Encryption & Decryption

Scenario: A company wants to protect sensitive data by encrypting text files. Write a program that reads a text file, encrypts the content using a simple character shift algorithm, and saves it to a new file.
Concepts Used: File handling, character manipulation, and loops.

TOTAL (LAB): 60 PERIODS

TOTAL: 75 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to:

CO1	Describe the C programming fundamental concepts and to select suitable keywords and data types for simple programs.	[U]
CO2	Choose appropriate control structures and looping techniques to design structured and efficient programs	[AN]
CO3	Apply arrays, strings, and pointers concepts to manipulate and process data for problem-solving in C programming	[AP]
CO4	Examine the effectiveness of functions and dynamic memory allocation in enhancing program modularity and efficiency	[AN]
CO5	Develop applications using structures, file handling and preprocessor directives to manage data and system interactions	[AP]

TEXT BOOKS

1. Jens Gustedt, "Modern C", 3rd Edition, Manning Publications, 2023.
2. Robert C. Seacord, "Effective C: An Introduction to Professional C Programming", 2nd Edition, No Starch Press, 2023.
3. Dan Gookin, "C Programming for Dummies", 2nd Edition, For Dummies Publication, 2020.

REFERENCE BOOKS

1. Brian W. Kernighan and Dennis M. Ritchie, "The C Programming Language", 2nd Edition, Prentice Hall, 2015.
2. Stephen G. Kochan, "Programming in C", 4th Edition, Developer's Library, 2014.
3. King. K. N., "C Programming: A Modern Approach", 2nd Edition, W. W. Norton & Company, 2008.

WEB RESOURCES

1. https://cs50.harvard.edu/x/
2. https://www.learn-c.org/
3. https://www.geeksforgeeks.org/c-programming-language
4. https://www.programiz.com/python-programming
5. https://www.w3schools.com/c/
6. https://www.codechef.com/cpp-online-compiler

25MEC03 SDG NO. 4, 9, 11, 12	DESIGN THINKING AND IDEA LABORATORY (Common to All Branches)	L	T	P	C
		0	0	2	1

COURSE OBJECTIVES

1. To accelerate development of indigenous products in line with the “Make in India” campaign.
2. To encourage aspiring engineers to actualize their ideas under one roof.
3. To impart multidisciplinary education to all students to promote innovation and product development.
4. To initiate new ways of creative thinking and enable the students to learn the cycle of Design Thinking process for developing innovative products.
5. To promote experiential learning and entrepreneurial skills among the students.

DESIGN THINKING

Design Thinking: Definition, Need and Objective, Concepts and Brainstorming, Stages – Empathize, Define, Ideate, Prototype, Test. Practical Examples of Customer Challenges, Alignment of Customer Expectations with Product Design - Feedback, Re-Design and Re-Create.

INTRODUCTION TO TOOLS AND EQUIPMENT

Introduction to Hand Tools and Power Tools - 3-axis CNC routing, basic turning, milling, drilling and grinding operations, Laser cutting, Laser engraving etc.

Basic 2D and 3D designing using CAD tools such as FreeCAD, Sketchup, Prusa Slicer, FlatCAM, Inkspace and OpenBSP - 2D and 3D structures for prototype building using CNC machine - Basic welding and other joining techniques for assembly - Basics of 3D scanning, Point cloud data generation for reverse engineering.

Exposure to PCB prototype fabrication - Familiarity and use of soldering and de-soldering equipment - Usage of Arduino, Raspberry Pi and Beagle Bone.

EXPERIMENTAL LEARNING

1. 2D profile cutting of press fit box / casing in acrylic (3 or 6 mm thickness) / polymer / cardboard / MDF (2 mm thickness) board using laser cutter and engraver.
2. Machine 3D geometry on soft material such as soft wood using CNC router.
3. Fabricate products like trusses using cutting and welding tools.
4. 3D printing of scanned geometry using FDM or SLA printer.
5. Designing a suitable PCB layout, fabrication and testing of the circuit.
6. Assemble and disassemble electronic components on a PCB using soldering and de-soldering equipment.
7. Embedded programming using Arduino, Raspberry Pi and Beagle Bone.

DESIGN THINKING PROJECT

1. Design and implementation of a capstone project.

TOTAL PERIODS: 30

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Summarize the concepts of Design Thinking.	[U]
CO2	Utilize the equipment, tools and inventories associated with Design Thinking Laboratory.	[AP]
CO3	Perform fundamental fabrication operation using hand tools, power tools, welding equipment, laser cutter and engraver.	[AP]
CO4	Perform fundamental electrical and electronic circuit design using PCB machine.	[AP]
CO5	Develop innovative products by implementing the design thinking approach	[C]

TEXT BOOKS

1. Veeranna D.K, “Workshop / Manufacturing Practices (with Lab Manual)”, AICTE’s Prescribed Textbook, Khanna Book Publishing, 1st Edition, 2022.
2. E. Balaguruswamy, “Design Thinking: A Beginner's Perspective”, McGraw-Hill Education, 1st Edition, 2024.
3. Anuja Agarwal, “Design Thinking: A Framework for Applying Design Thinking in Problem Solving”, Cengage India, 1st Edition, 2024.

REFERENCE BOOKS

1. Lal, D. M., “Design Thinking- Beyond the Sticky Notes”, Sage Publications India Pvt. Ltd., 1st Edition, 2021.
2. Kaushik Kumar and Muralidhar Kurni, “Design Thinking: A Forefront Insight”, CRC Press, 1st Edition, 2023.
3. Shalini Rahul Tiwari, “Design Thinking: A Comprehensive Textbook”, Wiley India, 1st Edition, 2024.

WEB RESOURCES

1. https://fab-coep.vlabs.ac.in/List%20of%20experiments.html
2. https://www.innovationtraining.org/how-to-use-design-thinking-to-design-an-innovation-lab/
3. https://www.erdster.co.in/design-thinking-lab.html
4. https://www.coursera.org/learn/uva-darden-design-thinking-innovation

25TA101 SDG : 4,11, 16	HERITAGE OF TAMILS	L	T	P	C
		1	0	0	1

COURSE OBJECTIVES

1. To know various concepts of Tamil Language families.
2. To know about the essentialities of Heritage.
3. To understand the Aram concepts of Tamils and the cultural influence.

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.

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.

Folk and Martial Arts **3**

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

Thinai Concept of Tamils **3**

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.

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 PERIODS: 15

Course Outcomes:
Upon completion of the course, students shall have ability to,

CO1	Know about the language families in India, impact of religions and the contribution of Bharathiyar and Bharathidhasan.	[U]
CO2	Observe the growth of sculpture, making of musical instruments and the role of temples in socio and economic lives.	[U]
CO3	Understand the significance of folklore and martial arts.	[U]
CO4	Learn the sangam literature, sangam age and overseas conquest of Cholas.	[U]
CO5	Understand the contribution of Tamils to Indian Freedom Struggle, role of Siddha medicine and print history of Tamil Books.	[U]

TEXT-CUM-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)

SEMESTER - II

25ITC01 SDG NO. 4,8,9	MODERN COMPUTER ARCHITECTURE	L	T	P	C
		3	1	0	4

COURSE OBJECTIVES

1. To understand the operations of digital computer systems and the significance of integrated circuits
2. To analyze the execution speed and performance of pipelining and superscalar techniques
3. To synthesize performance optimization of multicore architectures and memory technologies in advanced computing systems.
4. To familiarize the concepts of distributed computing systems and concurrency.
5. To analyze HPC architectures and design CUDA programs using threads, blocks, and efficient GPU-CPU data handling.

INTRODUCTION TO FUNCTIONAL UNITS **12**

Functional units, Basic operational concepts, Bus structures, Instruction set - Memory locations and addresses - computer registers and instructions, timing and control, instructions cycle, memory reference instruction, I/O interruption -Multiplication of signed numbers, Fast multiplication, Integer division, Floating point numbers and operations. Case study on mobile architectures (smartphones, tablets).

PIPELINING AND SUPERSCALAR TECHNIQUES **12**

Fundamental concepts - Execution of a complete instruction – Multiple bus organization - Hardwired control – Micro and Nano programmed control. Pipelining: Basic concepts - Data hazards - Instruction Hazards - Influence on Instruction sets - Data path and control consideration - Superscalar operation. Use of CPUs, GPUs, FPGAs, and other accelerators in modern computing systems.

MULTI-CORE ARCHITECTURE **12**

Memory technologies, hierarchical memory systems, the locality principle and caching, direct-mapped caches, block size, cache conflicts, associative caches, write strategies, advanced optimisations, performance improvement techniques, DRAM – organisation, access techniques. Tiled Chip Multicore Processors (TCMP), Network on Chips (NoC), NoC router – architecture, design.

DISTRIBUTED COMPUTING SYSTEMS AND CONCURRENCY **12**

Relation to Parallel Multiprocessors/multicomputer Systems, Distributed and Concurrent Programs, Message Passing vs. Shared Memory Systems, Synchronous vs. Asynchronous Executions, Design Issues and Challenges, Distributed Computing Technologies, Clocks and Synchronization, Coordination and Agreement Algorithms, Global State and Distributed Transactions.

HIGH PERFORMANCE COMPUTING (HPC) WITH CUDA **12**

HPC Architecture, Parallel Processing, Parallel Memory Models, Data vs. Task Parallelism, High Throughput Computing, Vectorization, Multithreading - CUDA programming model, Basic principles of CUDA programming, Concepts of threads and blocks

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Understand the functional units, memory instructions and I/O interruptions of computer systems.	[U]
CO2	Evaluate the parallel processor performance using pipelining techniques.	[AN]
CO3	Analyze the system performance using memory technologies, cache memories, multicore architectures and Noc routing architectures.	[AN]
CO4	Apply distributed computing and concurrency concepts to design efficient scalable systems.	[AP]
CO5	Apply parallel processing and CUDA programming to optimize HPC systems using threads, blocks, and GPU-CPU data exchange.	[AP]

TEXTBOOKS

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian, “Computer Organization”, McGraw- Hill, 6th Edition 2017
2. John L. Hennessy and David A. Patterson, “Computer Architecture: A Quantitative Approach”, Morgan Kaufmann, 6th Edition, 2017
3. John P. Hayes, “Computer Architecture and Organization”, McGraw-Hill, 3rd Edition, 2017.

REFERENCE BOOKS

1. David A. Patterson and John L. Hennessy, “Computer Organization and Design: The Hardware/Software Interface”, Elsevier, 5th Edition, 2013.
2. Flynn M. J., “Computer Architecture: Pipelined and Parallel Processor Design”, Narosa Publishing House, 2020.
3. William Stallings, “Computer Organization and Architecture Designing for Performance”, 11th Edition, Pearson Education, 2021.

WEB RESOURCES

1. https://www.cs.cmu.edu/~fp/courses/15213-s07/lectures/27-multicore.pdf
2. https://fdocuments.in/document/intel-core-i7-processor.html
3. https://www.intel.com/content/dam/www/public/us/en/documents/manuals/64-ia-32-architectures-software-developer-instruction-set-reference-manual-325383.pdf
4. https://www.coursera.org/learn/introduction-high-performance-computing#syllabus
5. https://onlinecourses.nptel.ac.in/noc20_cs41/preview

25MSC01 SDG NO. 3, 4, 7, 9, 11, 12, 13, 16	MATERIALS SCIENCE	L	T	P	C
		2	0	2	3

COURSE OBJECTIVES:

1. To explore the basic concepts of physics for computing engineering.
2. To impart the physics concepts in solving real time engineering problems.
3. To explore and visualize theoretical concepts of physics by computational methods.
4. To enhance the ability to record, analyse and interpret experimental data.
5. To familiarize the basic concepts, synthesis and application of nano materials.

WAVE OPTICS **6**

An introduction to Wave Optics, Interference - Principle of superposition - Michelson interferometer. Diffraction definition - Fresnel's diffraction and Fraunhofer's diffraction - Difference between interference and diffraction - Fraunhofer diffraction at Single slit-plane diffraction grating (Multi slit diffraction). Polarization definition – Brewsters law, Double refraction – Polarimeter – construction and working

QUANTUM MECHANICS **6**

Quantum Mechanics Introduction – Planck's quantum theory - Matter waves, de-Broglie wavelength, Heisenberg's uncertainty principle, Time independent and Time – dependent Schodinger's wave equation, Physical significance of wave function, Particle in one dimensional potential box

LASER AND FIBER OPTICS **6**

Characteristics of laser Principle of spontaneous emission and stimulated emission – Einstein's theory of matter radiation interaction and A and B coefficients (derivation) – Population inversion Pumping CO2 laser, Fiber optics: principle and propagation of light in optical fibers, Numerical aperture and acceptance angle, Application of optical fiber in communication system

ELECTRONICS AND OPTOELECTRONIC DEVICES **6**

Introduction to semiconductors Basic concept of Band theory - Basic of Intrinsic and extrinsic semiconductors PN Junction diode and its IV characteristics - Transistor Bi-polar Junction Transistor (BJT). Optoelectronic devices: light detectors and solar cells light emitting diode

NANO MATERIALS **6**

Nanomaterials Definition - Types - Comparison of nanomaterials with bulk materials. Nanoparticles, nanoclusters and nanorods - Preparation by sol-gel and solvo-thermal methods- Properties and applications. Carbon nanotubes preparation by chemical vapor deposition -

Properties and applications in electronics. Nanowires - Preparation by electrochemical deposition and electro-spinning - Properties and sensing applications

TOTAL (THEORY): 30 PERIODS

LAB EXPERIMENTS

1. Calculate the Wavelength of mercury spectrum using Spectrometer and grating.
2. Calculate the thickness of a thin wire using Air wedge interference method.
3. Exploring light and energy: A practical Planck's constant experiment.
4. Estimate the value Stefan's constant using an incandescent Bulb through V- Lab.
5. Analyze the dust particle size using the given LASER source.
6. Calculate the Numerical aperture and acceptance angle of the given optic fibre for data communication.
7. Estimate the Band gap of the given semiconductor.
8. Calculate the efficiency of the given Solar panel.
9. Analyze the Characteristics of the Light dependent resistor.
10. Evaluate the cathode efficiency of nickel using electro-deposition process.
11. Controlled synthesis of silver Nano particles using reducing agent.
12. Estimate the dissolved oxygen in wastewater using Winkler's method through V-lab.

LIFE SKILL EXPERIMENTS

1. Estimate the value of the given resistor, capacitor and voltage/Current fluctuations using Multimeter.
2. Identify the earth, neutral and phase line in an AC circuit.

TOTAL (LAB): 30 PERIODS
TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to:

CO1	To apply the principles of simple harmonic motion and wave optics in practical and theoretical contexts.	[AP]
CO2	Apply quantum principles to explain atomic and subatomic system behaviour.	[AP]
CO3	Understand the principles of Lasers, Einstein's coefficients and the fundamentals of Fiber optics along with their practical applications.	[U]
CO4	Apply the principles of semiconducting materials to analyze and design electronic circuits and systems.	[U]
CO5	Utilize the fundamental principles of nanomaterials to explore and implement their applications in Engineering.	[AP]

TEXT BOOKS

1. Bhattacharya,D.K. & Poonam,T. "Engineering Physics", Oxford University Press, 2015.
2. Malik,H.K. & Singh,A.K. "Engineering Physics", Tata McGraw-Hill Education, 2017.
3. Gary N. Felder & Kenny M. Felder, "Modern Physics", Cambridge University Press, 2022.
4. Sulabha K. Kulkarnl, Nanotechnology: principles and practices, Springer publications, , 3rd Edition, 2018.

REFERENCE BOOKS

1. Halliday, D., Resnick, R. & Walker, J., "Principles of Physics: Extended, International Adaptation", 12th Edition, Wiley, 2023.
2. Serway, R.A. & Jewett, J.W., "Physics for Scientists and Engineers", Cengage Learning, 2019.
3. Jenkins & White, "Fundamentals of Optics", 4th Edition, McGraw Hill Education, 2017.
4. B. Rogers, J Adams and S. Pennathur, Nanotechnology the whole story, CRC Press, 2013.

WEB RESOURCES

1. https://www.khanacademy.org/science/physics
2. http://www.feynmanlectures.caltech.edu/info/
3. https://physics.info/
4. http://hyperphysics.phy-astr.gsu.edu/hbase/index.html

5IT201 SDG NO. 4,8,9	OBJECT ORIENTED PROGRAMMING USING C++	L	T	P	C
		1	0	4	3

COURSE OBJECTIVES

1. To outline the basic structure, syntax, and semantics of C++ programming.
2. To illustrate the use of control structures, functions, and recursion in solving computational problems.
3. To illustrate the control structures, functions, and recursion to solve real-world problems.
4. To explore object-oriented programming concepts such as classes, objects, inheritance, and polymorphism
5. To inculcate the insights on pointers, memory allocation, and file handling for advanced programming solutions.
6. To use the Standard Template Library (STL) and modern programming concepts like lambda functions and exception handling

INTRODUCTION TO C++ AND PROGRAMMING BASICS **3**

Introduction to C++: History, Features, Structure of C++ Program, Basic Syntax: Variables, Data Types, Constants, Header Files, Operators in C++: Arithmetic, Relational, Logical, Assignment, Bitwise, and Precedence, Scope Resolution Operator, Conditional Statements: if, if-else, switch-case, Looping Statements: for, while, do-while, break, continue, goto, Arrays: 1D and 2D Arrays, Array Manipulations, Strings and String Functions: String Handling in C++.

FUNCTIONS, POINTERS, AND MEMORY MANAGEMENT **3**

Functions: Function Basics, Call by Value, Call by Reference, Return by Reference. Inline Functions, Function Overloading. Recursion: Recursive Functions in C++. Pointers: Pointers and Dynamic Memory Allocation (malloc, free, new, delete).

CLASSES, CONSTRUCTORS, AND OPERATOR OVERLOADING **3**

Object Oriented Concepts – Features of OOP - Classes and Objects: Defining Classes, Access Specifiers (public, private, protected), Static Data members and Static Member Function. Constructors and Destructors: Default, Parameterized, Copy Constructors, Shallow Copying, Deep Copying. Friend Functions and Friend Classes, Overloading 'this' Pointer. Operator Overloading: Implementing Operator Overloading.

INHERITANCE, POLYMORPHISM, AND EXCEPTION HANDLING **3**

Inheritance: Single, Multilevel, and Multiple Inheritance, Function Overriding, Overloading vs. Overriding, Polymorphism: Compile-time and Run-time Polymorphism, Virtual Functions, Pure Virtual Functions, Abstract Classes. Exception Handling: Built-in and Custom Exceptions, try-catch Blocks.

FILE HANDLING, STL, AND LAMBDA **3**

File Handling in C++: File I/O, File Streams (ifstream, ofstream, fstream), Templates: Function Templates, User Defined Templates, Class Templates. Standard Template Library (STL): Introduction to STL, Generic Programming. Lambda Functions: Syntax, Applications and Use Cases.

TOTAL (THEORY): 15 PERIODS

LIST OF EXPERIMENTS

1. You are building a grade calculator where students are given scores in three subjects. You need to calculate the total score and then determine the grade based on the total score.
   Question: Write a C++ program where you:
   a. Accept three subject scores (out of 100).
   b. Calculate the total score using the arithmetic + operator.
   c. Use a relational operator to determine if the total score is above a passing grade (e.g., 50).
   d. Print a message based on the result (either "Pass" or "Fail").
2. Simple C++ Programs to Implement Various Control Structures.
   a. If statement
   b. Switch case statement and
   c. For loop
   d. While loop
   e. do while loop
3. Implement a C++ Program to the following
   a. Perform sorting and find maximum, and minimum elements in the given 1D array
   b. Perform Matrix addition, Matrix Multiplication, and Matrix transposition using 2D array
4. You are building a simple utility that checks whether a given string is a palindrome. A palindrome is a word, phrase, number, or other sequence of characters that reads the same forward and backward, ignoring spaces, punctuation, and letter case.
   Question: Write a C++ program that:
   a. Accepts a string as input from the user.
   b. Removes spaces and converts the string to lowercase.
   c. Checks if the string is a palindrome.
   d. Prints the result: either "The string is a palindrome" or "The string is not a palindrome".
5. Implement a C++ program to Factorial and Fibonacci concepts using Recursion
6. Implement C++ program using dynamic memory allocation.
7. Implement a C++ program using class and object to solve real world problem
8. Implement a C++ program using constructor and destructor.

9. Implement a C++ program using the Friend Function to perform vector addition, subtraction, and multiplication.
10. Implement a C++ program to perform operator overloading using unary and binary operators.
11. Implement a C++ program using Inheritance with real-world examples.
12. Implement a C++ program Using Polymorphism.
13. Implement a C++ program using virtual and pure virtual Function.
14. Implement a C++ program using exception handling.
15. Implement a C++ program to perform Student Grades Management using file concept.
16. Implement a C++ program Using Template.
17. Develop an application in C++ using the oops Concept.

TOTAL (LAB): 60 PERIODS
TOTAL: 75 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Apply the basic syntax and operators in C++ to develop functional programs	[AP]
CO2	Construct programs using control structures, arrays, and functions to solve various computational problems	[AP]
CO3	Utilize key +concepts such as encapsulation, inheritance, and polymorphism to design and implement software solutions	[AP]
CO4	Design and develop robust programs using dynamic memory allocation, exception handling, and file operations	[AP]
CO5	Utilize STL, lambda expressions, and abstract classes to improve code efficiency and reusability	[AP]

TEXTBOOKS

1. Robert Lafore, “Object-Oriented Programming in C++”, Sams Publishing, 4th Edition, 2023.
2. Yashavant Kanetkar, “Let Us C++”, BPB Publications,17th Edition, 2022.
3. Ashok N. Kamthane, “Object-Oriented Programming with ANSI & Turbo C++”, Pearson Education, 7th Edition, 2016.
4. Bjarne Stroustrup, “The C++ Programming Language”, Addison-Wesley, 4th Edition, 2013.

REFERENCE BOOKS

1. Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo, "C++ Primer", Addison-Wesley, 5th Edition, 2010.
2. Herbert Schildt, "C++: The Complete Reference", McGraw Hill Education, 4th Edition, 2021.
3. Paul Deitel and Harvey Deitel, "C++ How to Program", Pearson Education,10th Edition 2017.

WEB RESOURCES

1. https://www.programiz.com/cpp-programming/operators
2. https://www.tpointtech.com/cpp-inheritance
3. https://cplusplus.com/doc/tutorial/files/
4. https://onlinecourses.nptel.ac.in/noc21_cs38/preview
5. https://www.coursera.org/specializations/hands-on-cpp

25CSC01	DATA STRUCTURES	L	T	P	C

SDG NO. 4, 9		3	0	2	4

COURSE OBJECTIVES:

1. To learn the basics of data structures and linked lists, and implement common sorting techniques.
2. To gain hands-on experience in implementing stacks and queues using arrays and linked lists, and apply them to solve problems like expression evaluation and priority queuing.
3. To understand the usage of binary trees, binary search trees, and searching algorithms for efficient data storage and manipulation.
4. To explore graph representations and traversal techniques, and implement advanced sorting algorithms.
5. To learn hashing techniques and work with advanced data structures such as AVL trees, B-trees, red-black trees, and heaps.

INTRODUCTION TO DATA STRUCTURES AND LINKED LISTS **9**

Introduction to Data Structures in Linked Lists - Understanding Data Structures and their importance of Linked List, Comparison: Arrays vs Linked Lists, Types of Linked Lists: Singly Linked List: Creation, Insertion, Deletion, and Traversal, Doubly Linked List: Creation, Insertion, Deletion, and Traversal, Circular Linked List: Creation and Applications. Sorting Algorithms - Basic Sorting Techniques: Bubble Sort, Insertion Sort, Selection Sort: Asymptotic notations Implementation and Time Complexity Analysis

STACK, QUEUE IMPLEMENTATIONS, AND APPLICATIONS **9**

Stack Implementations: Stack Model and Concept, Array and Linked List Implementations of Stack. Applications of Stack - Infix, Prefix, and Postfix Expressions: Conversion and Evaluation, Balancing Parentheses using Stack. Queue Implementations - Array and Linked List Implementations of Queues, Priority Queue and Applications.

TREES AND TREE TRAVERSALS **9**

Trees: Creation, Traversal, and Searching - Binary Tree: Structure, Creation, Insertion, and Deletion, Binary Search Tree (BST): Concept and Operations, Tree Traversals: Inorder, Preorder, Postorder, and Level Order Traversals, Searching Algorithms - Linear Search and Binary Search, Applications: Search in Sorted and Rotated Arrays,

GRAPHS AND ADVANCED SORTING TECHNIQUES **9**

Graph Implementations - Weighted and Directed Graphs, Adjacency Matrix and Adjacency List Implementations. Graph Traversal Techniques: Breadth-First Search (BFS) and Depth-First Search (DFS), Advanced Sorting Techniques: Merge Sort, Quick Sort, Counting Sort, Radix Sort.

HASHING AND ADVANCED DATA STRUCTURES **9**

Hashing Techniques: Direct Address Table, Hash Functions, Collision Resolution Techniques: Linear Probing, Quadratic Probing, Double Hashing. Advanced Data Structures - AVL Trees, B-Trees, Red-Black Trees, Heaps.

TOTAL (THEORY): 45 PERIODS

List of Lab Experiments:

1. Implement Singly Linked List operations (Creation, Insertion, Deletion, Traversal)
2. Implement Doubly Linked List operations (Creation, Insertion, Deletion, Traversal)
3. Implement Circular Linked List operations and explore its applications
4. Implement Stack operations using Array and Linked List
5. Evaluate Infix, Prefix, and Postfix Expressions using Stacks
6. Implement Queue operations using Array and Linked List
7. Implement Breadth-First Search (BFS) and Depth-First Search (DFS) on Graphs
8. Implement Binary Tree operations (Creation, Insertion, Deletion) and Tree Traversals
9. Implement Hashing with Collision Resolution Techniques (Linear and Quadratic Probing)
10. Implement Sorting Algorithms (Merge Sort, Quick Sort, and Counting Sort) and evaluate their time complexities

TOTAL (LAB): 30 PERIODS

TOTAL: 75 PERIODS

COURSE OUTCOMES (COs):

Upon successful completion of this course, students will be able to:

CO1	Analyze the appropriateness of using core data structures like arrays and linked lists to implement efficient algorithms.	[AN]
CO2	Utilize stacks and queues to develop solutions for real-world computational challenges.	[AP]
CO3	Apply binary trees, binary search trees (BSTs), traversal techniques, and searching algorithms to organize and retrieve data efficiently.	[AP]
CO4	Evaluate the effectiveness of various graph algorithms and advanced sorting techniques in solving computational problems.	[AN]
CO5	Design efficient solutions using hashing, searching, and sorting techniques to optimize problem-solving in real-world applications.	[AP]

TEXT BOOKS:

1. Narasimha Karumanchi, “Data Structures and Algorithms Made Easy”, 6th Edition, CareerMonk Publications, 2022.
2. Seymour Lipschutz, “Data Structures with C (SIE)”, 1st Edition, McGraw Hill Education, 2019.
3. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, 4th Edition, Pearson, 2014.
4. Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed, “Fundamentals of Data Structures in C/C++”, 2nd Edition, Silicon Press, 2008.

REFERENCE BOOKS:

1. Reema Thareja, “Data Structures Using C”, 3rd Edition, Oxford University Press, 2022.
2. Jean-Paul Tremblay and Paul G. Sorenson, “An Introduction to Data Structures with Applications”, 2nd Edition, McGraw-Hill Education, 1984.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, “Introduction to Algorithms”, 3rd Edition, MIT Press, 2009.
4. Debasis Samanta, “Classic Data Structures”, 2nd Edition, PHI Learning, 2009.

WEB REFERENCES:

1. https://www.khanacademy.org/computing/computer-science/algorithms
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-spring-2011/
3. https://www.w3schools.com/dsa/dsa_intro.php
4. https://techdevguide.withgoogle.com/paths/data-structures-and-algorithms/

25CY201 SDG No. 4,9,17	APPLIED STATISTICS USING PYTHON (Common to all Computing Sciences)	L	T	P	C
		3	0	2	4

COURSE OBJECTIVES

1. To understand and gain the knowledge of linear regression models
2. To learn the concept of testing hypotheses using statistical analysis
3. To develop a sound understanding of current, modern computational statistical approaches and their application to a variety of datasets.
4. To understand the key technologies in data science and business analytics.
5. To effectively visualize the data using tools Matplotlib and Seaborn to communicate statistical findings.

DESCRIPTIVE MEASURES AND NORMAL DISTRIBUTION 9

Central tendency- Mean, Median and Mode, Dispersion- Range, Quartile deviation, Standard deviation. Bivariate data. Summarization, marginal and conditional frequency distribution. Linear Regression Model. Multivariate Normal Distribution Functions, Conditional Distribution and its relation to regression model, Estimation of parameters. Standard multiple regression models.

TESTING OF HYPOTHESIS 9

Concept and formulation – Type I and Type II errors – Neyman Pearson lemma – Procedures of testing: Small samples: Student's t test – F test – Chi square test – Large samples: Difference of Means and proportions. Multivariate Analysis of Variance (MANOVA).

.

PRINCIPAL COMPONENT ANALYSIS AND CLUSTERING 9

Principal Component Analysis: Principal components, Algorithm for conducting principal component analysis, deciding on how many principal components to retain, H-plot. Cluster Analysis: Introduction, Types of clustering, Correlations and distances, clustering by partitioning methods, hierarchical clustering, overlapping clustering, K-Means Clustering-Profiling and Interpreting Clusters.

PYTHON CONCEPTS, DATA STRUCTURES, CLASSES 9

Interpreter, Program Execution, Statements, Expressions, Flow Controls, Functions, Numeric Types, Sequences and Class Definition, Constructors, Text & Binary Files - Reading and Writing. Data Wrangling: Combining and Merging Datasets, Reshaping and Pivoting, Data Transformation, String Manipulation, Regular Expressions

DATA AGGREGATION, GROUP OPERATIONS AND VISUALIZATION 9

Group by Mechanics, Data Aggregation, Group wise Operations and Transformations, Pivot Tables and Cross Tabulations, Time Series Basics, Data Ranges, Frequencies and Shifting Visualization in Python: Matplotlib package, Plotting Graphs, Controlling Graph, Adding Text, More Graph Types, Getting and setting values, Patches.

TOTAL PERIODS: 45

Lab Exercises:

1. Generation of prime numbers and computation of GCD for foundational cryptographic operations.
2. Analysis of student scores to determine statistical measures and grade classification.
3. Execution of matrix operations using NumPy to simulate basic image processing functions.
4. Solving linear equations for engineering and business use cases using matrix algebra in Python.
5. Conducting hypothesis testing using t-test and chi-square test on small and large samples.
6. Building a logistic regression model to predict student performance outcomes.
7. Performing sales data analysis through data wrangling techniques using the pandas library.
8. Analyzing and visualizing time-series trends from datasets such as COVID-19 case statistics.
9. Application of K-Means clustering for customer segmentation using Scikit-learn.
10. Generating and interpreting statistical visualizations using the Seaborn package.

TOTAL (LAB) PERIODS: 30
TOTAL PEROIDS: 75

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Recognize multivariate normal distributions and understand their relevance in modeling complex datasets.	[U]
CO2	Use testing hypotheses in statistical analysis, estimators and time series analysis.	[AP]
CO3	Categorize datasets using modern computational statistical approaches.	[U]
CO4	Demonstrate original, non-trivial Python programs.	[AP]
CO5	Apply algorithms to build machine intelligence.	[AP]

TEXTBOOKS

1. Gupta, S.C. and Kapoor, V.K, Fundamentals of Mathematical Statistics. 12th Edition, 12thEdition, Sultan Chand & Sons, 2020.
2. Jobson. J.D, “Applied Multivariate Data Analysis”, Vol II, Springer, 2012.
3. Wes Mc Kinney, “Python for Data Analysis”, O’Reilly, 3rd Edition 2022.

REFERENCE BOOKS

1. Miller. I.R, Freund. J.E and Johnson. R, “Probability and Statistics for Engineers, Pearson Education, Asia, 9th Edition, 2016.
2. Brian S. Everitt, Graham Dunn, “Applied Multivariate Data Analysis” John Wiley & Sons, Ltd, 2nd Edition, 2001.
3. Mark Summerfield, Programming in Python 3: A Complete introduction to the Python Language, Addison-Wesley Professional, 2009.

WEB RESOURCES

1. https://www.edx.org/course/statistical-modeling-and-regression-analysis
2. https://www.cin.ufpe.br/~embat/Python%20for%20Data%20Analysis.pdf
3. https://www.kdnuggets.com/2016/07/statistical-data-analysis-python.html
4. https://people.duke.edu/~ccc14/sta-663/

25CS202 SDG NO. 4,8,9,11	APPLICATION DEVELOPMENT PRACTICES	L	T	P	C
		0	0	4	2

COURSE OBJECTIVES

1. To impart the knowledge of fundamental concepts in HTML, CSS, and Bootstrap.
2. To familiarize the concepts of JavaScript and TypeScript features in creating dynamic, data-driven applications.
3. To gain proficiency in Linux shell scripting and basic Linux commands.
4. To utilize branching, merging, and remote repository operations in Git and GitHub.
5. To effectively use Git and GitHub for version control, collaboration, and managing software development projects.

LIST OF EXPERIMENTS

1. Create a responsive web page with a registration form using HTML5, CSS3, and Bootstrap grid system.
2. Enhance web elements with CSS: Shadows, Rounded Borders, and Multiple Backgrounds
3. Implement a CSS grid-based layout to create a responsive web page.
4. Implement a JavaScript-based form validation for user input fields (email, password, etc.).
5. Develop a dynamic webpage using JavaScript and DOM manipulation.
6. Work with APIs and fetch data asynchronously using JavaScript.
7. Create a TypeScript application demonstrating object-oriented programming concepts.
8. Write a Linux shell script to automate file management tasks.
9. Schedule background jobs using Linux cron and batch scripting.
10. Perform Git version control operations: cloning, committing, branching, and merging.
11. Collaborate on a project using GitHub: Forking, pull requests, and resolving merge conflicts.
12. Deploy a static website using GitHub Pages.

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Develop structured and responsive web pages using HTML, CSS, and Bootstrap	[AP]
CO2	Design JavaScript and TypeScript programs to create dynamic, interactive applications.	[AP]
CO3	Build and execute Linux shell scripts to automate tasks, manage files, and schedule jobs	[AP]
CO4	Utilize Git and GitHub for version control, including repository management, branching, merging, and collaborative development	[AP]

CO5 Apply Agile principles for project management, using GitHub tools to track progress, manage tasks, and enhance team collaboration [AP]

TEXT BOOKS

1. Duckett, J., "HTML and CSS: Design and Build Websites", 1st Edition, Wiley, 2011.
2. Flanagan, D., "JavaScript: The Definitive Guide", 7th Edition, O'Reilly Media, 2020.
3. Cherny, B., "Learning TypeScript: Enhance Your Web Development Skills", O'Reilly Media, 2022.
4. Tushar, S., & Lakshman, S, "Linux Shell Scripting Cookbook", 2nd Edition. Packt Publishing, 2013.
5. Chacon, S., & Straub, B., "Pro Git", 2nd Edition, Apress, 2014.

REFERENCE BOOKS

1. Harris, A., "HTML5 and CSS3 All-in-One For Dummies". Wiley, 2014.
2. Haverbeke, M, "Eloquent JavaScript", 3rd Edition, No Starch Press, 2018.
3. Simpson, K, "You Don't Know JS: Scope & Closures", O'Reilly Media, 2014.
4. Loeliger, J., & McCullough, M. "Version Control with Git", 2nd Edition, O'Reilly Media, 2012.
5. Sass, H., "Mastering Bootstrap 5", Packt Publishing, 2021.

WEB RESOURCES

1. https://developer.mozilla.org/en-US/docs/Web/HTML
2. https://developer.mozilla.org/en-US/docs/Web/CSS
3. https://developer.mozilla.org/en-US/docs/Web/JavaScript
4. https://www.coursera.org/learn/html-css-javascript-for-web-developers
5. https://online-learning.harvard.edu/subject/javascript

25TA201 SDG: 4,11,16	TAMILS AND TECHNOLOGY / தமிழரும் தொழில்நுட்பமும்	L	T	P	C
		1	0	0	1

COURSE OBJECTIVES

1. To know about weaving, ceramic, design and construction technologies in sangam age.
2. To know the significance of technologies such as manufacturing, agriculture and irrigation.
3. To be aware of the development of Scientific Tamils and Tamil Computing.

Weaving And Ceramic Technology **3**

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

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.

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.

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.

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 PERIODS: 15

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Describe about the weaving industry in sangam age and ceramic technology.	[U]
CO2	Observe the design of houses, sculptures and construction of temples.	[U]
CO3	Relate the various manufacturing materials and stone types in Silappathikaram.	[U]
CO4	Understand the significance of agriculture and irrigation technology in ancient period.	[U]
CO5	Explain the growth of scientific Tamil, Tamil computing and digitization of Tamil books.	[U]

TEXTBOOKS

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)

REFERENCE BOOKS

1. “Social Life of Tamils” (Dr.K.K.Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
2. “Social Life of the Tamils - The Classical Period” (Dr.S.Singaravelu) (Published by: International Institute of Tamil Studies.

WEB RESOURCES

1. https://archive.org/details/keeladibookenglish18092019/page/n13/mode/2up
2. https://www.tamildigitallibrary.in/admin/assets/book/TVA_BOK_0005812_Language_and_Literature.pdf
3. https://archive.org/details/lLXZ_historical-heritage-of-the-tamils-edited-by-s-v-subrahmanian-and-k-d-thirunavukk

25IKC01 SDG : 3,4,5,11,16	INTRODUCTION TO INDIAN KNOWLEDGE SYSTEM - I	L	T	P	C
		2	0	0	2

COURSE OBJECTIVES

1. Identify key aspects of Indian knowledge systems, including philosophy, science, mathematics, medicine (Ayurveda), astronomy, and architecture.
2. Highlight the contributions of Indian philosophy and spirituality to global thought and ethics.
3. Illustrate the cultural diversity of India through its languages, festivals, arts, and traditions
4. Implement practices from Indian knowledge systems such as yoga, meditation, and sustainable living in daily routines.
5. Compare ancient Indian scientific theories with contemporary scientific principles.

INTRODUCTION TO INDIAN KNOWLEDGE SYSTEMS (IKS) **6**

About Indian Knowledge System; Definition of Indigenous/ Traditional Knowledge; Scope, and Importance of Traditional Knowledge
Ancient India - Bharat Varsha: People of Ancient Bharat Varsha; Our great natural heritage: The great Himalayas and the rivers; The civilizations of the Sindhu -Ganga valley, Ancient Indian Traditional Knowledge and Wisdom about nature and climate.

INDIAN HERITAGE OF KNOWLEDGE **6**

Ancient Indian Knowledge: The Vedas and its components - the Vedangas Ancient Indian books and treaties: The Sastras.; The Great Indian Epics: The Ramayana and The Mahabharata Epics.

Languages and language studies in India

What is linguistics? Script and Language; Alphabet of the Indian; languages Varnamala: Origin, Evolution, and phonetic features; Languages of India; Important texts of Indian languages: Skills Siksha, Expression/Prounciation-Nirukta.

INTRODUCTION TO FINE ARTS AND PERFORMING ARTS OF INDIA **6**

Ancient Indian classical music and dance forms: The Science of Dramas- Natyasastra and the Science of Music-Gandharva-Veda; Aesthetics in Indian Art and Culture; Folk music and traditional dance forms of the Northeast.

Indian Science & Technology:

Ancient India's contribution to Mathematics - Number System. Algebra and Arithmetic, Geometry and Trigonometry; Origin of Decimal system in India; nomenclature of numbers in the Vedas. Zero and Infinity Sulba-sutras.

INDIAN ASTRONOMY **6**

Planetary System. Motion of the Planets; Velocity of Light; Eclipse. Astronomy. Navagrahas. Important works in Indian Astronomy. Aryabhata and Nilakantha: Contribution to Astronomical Studies

Indian Metal Works: Mining Techniques. Types of Metals. Tools & Techniques for Metal Smelting with examples.

CONTRIBUTION OF ANCIENT INDIA TO HEALTH SCIENCES **6**

Traditional Indigenous systems of medicines in India: - Ayurveda and Yoga; Elements of Ayurveda: Gunas and Doshas, Pancha Mahabhuta and Sapta-dhatu; Concept of disease in Ayurveda; Ayurvedic lifestyle practices: Dinacharya and Ritucharya; Important Ayurvedic Texts; Hospitals in Ancient India; Ayurveda: Gift of India to the modern world.

The experiential learning sessions may include:

1. Field Visits: Organizing visits to historical sites, museums, traditional craft centers, and other places relevant to Indian knowledge systems.
2. Interactive Sessions: Engaging students in discussions with experts and practitioners in various fields of Indian knowledge systems to gain insights and practical knowledge.
3. Online Lecture Series: Providing the students with online lectures by distinguished experts in the field of the Indian Knowledge System.
4. Hands-on Activities: Providing opportunities for students to participate in activities related to traditional arts, crafts, music, dance, agriculture, etc., to understand the practical aspects of Indian knowledge systems.
5. Practical Demonstrations: Conducting workshops or sessions to demonstrate traditional practices, such as yoga, Ayurveda, Vastu Shastra, etc., for the students.

TOTAL PERIODS: 30

COURSE OUTCOMES

Upon completion of the course, students shall have ability to

CO1	Recall the rich heritage of Indian knowledge systems	[U]
CO2	Describe the contribution of Indian knowledge systems to the world.	[R]
CO3	Demonstrate knowledge of sociocultural and ethnolinguistic diversity that constitutes the soul of Bharatvarsha.	[R]
CO4	Apply traditional knowledge and techniques in day-to-day life.	[AP]
CO5	Distinguish knowledge traditions that originated in the Indian subcontinent.	[AP]

TEXT BOOKS

1. Mahadevan, B., Bhat Vinayak Rajat, Nagendra Pavan RN, “Introduction to Indian Knowledge System: Concepts and Applications”, PHI Learning Private Ltd.2022.
2. Mukul Chandra Bora, “Foundations of Bharatiya Knowledge System”, Khanna Book Publishing, 2023.

REFERENCE BOOKS

1. Dharampal, “The Beautiful Tree: Indian Indigenous Education in the Eighteenth Century”, Dharampal Classics Series, 2021.
2. Baladev Upadhyaya, “Samskrta Śāstrom ka Itihās”, Chowkhambha, Varanasi, 2010.
3. Bose,D.M.,Sen, S.N. and Subbarayappa,B.V., Eds., “A Concise History of Science in India”, 2nd Ed., Universities Press, 2010.
4. Astāngahrdaya, Vol. I, Sūtrasthāna and Śarīrasthāna, Translated by K. R. Srikantha Murthy, Vol. I, Krishnadas Academy, 1991.
5. Bajaj,J.K. and Srinivas,M.D., “Indian Economy, and Polity in Eighteenth-century Chengalpattu”,J. K. Bajaj ed., Indian Economy and Polity, Centre for Policy Studies, Chennai, 1995.