Syllabus for Bachelors of Engineering
In
Computer Science and Engineering
[2018-19: 175 Credits]

Version 2.0

UNDERGRADUATE PROGRAMME

Effective from: 1st August 2020

Department of Computer Science and Engineering
Email: hodcse@sdmcet.ac.in
SDM College of Engineering and Technology,
Dhavalagiri, Dharwad-580002
Karnataka State
Ph: 0836-2447465/ 2448327 Fax: 0836-2464638 Email: hodcse@sdmcet.ac.in principal@sdmcet.ac.in

Table of Contents

Preamble

Department of Computer Science and Engineering, established in the year of 1985, is running one undergraduate program B.E. (Computer Science and Engineering), one postgraduate program M.Tech (Computer Science and Engineering). The department is recognized as a research center by Visveswaraya Technological University, Belgavi. The department is accredited twice by the National Board of Accreditation.

B.E Computer Science and Engineering programme is transforming itself to the **Outcome Based Education (OBE) since the year 2010**. The model used for transformation is **"incremental and iterative process"** of:

1. Understanding PEOs and POs.
2. Curriculum transformation.
3. OBE based teaching learning process.
4. Measurements of attainment of PEOs, POs and COs.

This document is the first version of syllabus with 2018-19 timestamp for all courses covering detailed contents for 3rd and 4th semester. The scheme of remaining semesters along with contents of 3rd and 4th semesters are approved in the Board of Studies meeting held on 11th May 2019. The various academic terminologies and vocabularies used in this Institution along with various academic rules and regulations are made available in the handbook printed separately.

The details of DUGC and BoS are listed in the next section

Department Undergraduate Committee- DUGC:

1. Dr. U.P. Kulkarni – Chairman.
2. Prof. R.N.Yadawad – Member Secretary
3. Prof. J.V.Vadavi – Member
4. Prof. R.G. Yadawad – Member
5. Prof. Anand Vaidya – Member
6. Prof. Indira Umarji – Member
7. Dr. S.M.Joshi – Invitee.

Board of Studies- *BoS*: (2020 – 2022)

Sl. No.	Particulars	Name of experts	Address	Mobile No.
1	Chairman	Dr. U.P. Kulkarni	Head of the Department, Dept. of Computer Science & Engineering, SDMCET, Dharwad, upkulkarni@yahoo.com	9448915301
2	One Expert from outside the college to be nominated by the Honorable Vice Chancellor, VTU	Dr. RBV Subramaanyam	Professor Department of CSE, NIT Warangal-506004, rbvs66@gmail.com	9491346969
3	Two Experts in the subject from outside the college to be nominated by the Academic Council.	1) Dr.Rajshekar. K.	Assistant Professor Dept of CSE IIT, Dharwad Dharwad - 580011. rajshekar.k@iitdh.ac.in	0836-2212888
		2) Dr.Shridhar K.S.	Professor and Principal, UBDT, Davangere, ks_shreedhara@yahoo.com	9448009306
4	One representative from industry / Corporate sector / allied area relating to placement to be nominated by the Academic Council.	Mr.Sreenivas Ramanujam	Academic Relationship Manager - Karnataka TCS Bangalore Vydehi, RC- 1 Block, 82, EPIP, Whitefield, Bangalore - 560066 sreenivasa.ramanujam@tcs.com	080-67257654, 8147002870

5	One Postgraduate meritorious Alumnus to be nominated by the Principal	Mr. Vijay Upadhyay	Technologist, Torry Harris Integration Solutions, #71, Sona Towers, Millers Road, Bangalore – 560052 vijayru@gmail.com	9845221770
6	Five Faculty members at different levels covering different specializations to be nominated by the Academic Council.	1) Dr. S.M.Joshi.	Professor, Dept. of CSE, SDMCET, Dharwad joshshree@gmail.com	9036079402
		2) Prof J. V. Vadavi	Associate Professor, Dept. of CSE, SDMCET, Dharwad jvvadavi@gmail.com	9448001249
		3) Prof. R.G.Yadawad	Assistant Professor, Dept. of CSE, SDMCET, Dharwad rgyadawad@gmail.com	9448049909
		4) Prof. Indira Umarji	Assistant Professor, Dept. of CSE, SDMCET, Dharwad indira.umarji@gmail.com	9945348887
		5) Dr. R.N.Yadawad [Member Secretary]	Assistant Professor, Dept. of CSE, SDMCET, Dharwad rnyadawad@gmail.com	9164685527

Board of Studies- *BoS*: (2018 – 2020)

Sl. No.	Particulars	Name of experts	Address	Mobile No.
1	Chairman	Dr. U.P. Kulkarni	Head of the Department, Dept. of Computer Science & Engineering, SDMCET, Dharwad, upkulkarni@yahoo.com	9448915301
2	One Expert from outside the college to be nominated by the Honorable Vice Chancellor, VTU	Dr. H N Champa	Department of Computer Science & Engineering, Central College Campus, Bangalore University, Bengaluru-560001, champahn@yahoo.co.in	9448256492
3	Two Experts in the subject from outside the college to be nominated by the Academic Council.	3) Mr.Girish Aithal	4Edge IT Solutions, Bejai-Kapikad Road, Mangaluru - 575 004 gaithal@gmail.com	9740160986
		4) Dr.Shridhar K.S.	Professor, Dept. of CSE, UBDT, Davangere, ks_shreedhara@yahoo.com	9448009306

4	One representative from industry / Corporate sector / allied area relating to placement to be nominated by the Academic Council.	Mr.Sreenivas Ramanujam	Academic Relationship Manager - Karnataka TCS Bangalore Vydehi, RC- 1 Block, 82, EPIP, Whitefield, Bangalore - 560066 sreenivasa.ramanujam@tcs.com	080-67257654, 8147002870
5	One Postgraduate meritorious Alumnus to be nominated by the Principal	Mr. B. Harish Kamath	Master Technologist, Hewlett-Packard Enterprise, Sy.No.192, Whitefield Road, Mahadevapura, Bengaluru, Karnataka 560048 harish.kamath@hpe.com	9845221770
6	Five Faculty members at different levels covering different specializations to be nominated by the Academic Council.	1) Dr. U. P. Kulkarni	Professor, Dept. of CSE, SDMCET, Dharwad upkulkarni@yahoo.com	9448915301
		2) Mrs. J.C. Karur	Professor, Dept. of CSE, SDMCET, Dharwad jck1965@gmail.com	8971212216
		3) Prof J. V. Vadavi	Associate Professor, Dept. of CSE, SDMCET, Dharwad jvvadavi@gmail.com	9448001249
		4) Prof. Anand Vaidya	Assistant Professor, Dept. of CSE, SDMCET, Dharwad vaidya.anand@rediffmail.com	9036889732
		5) Prof. R.N.Yadawad	Assistant Professor, Dept. of CSE, SDMCET, Dharwad rnyadawad@gmail.com	9164685527
7	Special Invitee	1) Dr.S.M.Joshi.	Professor, Dept. of CSE, SDMCET, Dharwad joshshree@gmail.com	9036079402

Prepared By: Dr. Ramchandranayak.N. Yadawad, Asst.Professor & Member Secretary
Dept. of CSE, SDMCET, Dharwad.

Vision and Mission of the Institution

VISION:

To develop competent professionals with human values.

MISSION:

1. **To have contextually relevant Curricula.**
2. **To promote effective Teaching Learning Practices supported by Modern Educational Tools and Techniques.**
3. **To enhance Research Culture.**
4. **To involve the Industrial Expertise for connecting Classroom contents to real-life situations.**
5. **To inculcate Ethics and soft-skills leading to overall personality development.**

Vision and Mission of the Department

VISION :

To develop competent professionals in the field of Computer Science and Engineering with human values.

MISSION:

1. To have **contextually relevant curricula** in line with industry trends and body of knowledge stated by **IEEE/ACM**.
2. To promote **OBE based effective Teaching Learning Practices** supported by modern educational tools and techniques.
3. To enhance **research**.
4. To involve the **industrial expertise** for connecting classroom contents to real-life situations.
5. To inculcate **ethics and soft-skills** leading to overall personality development.

QUALITY POLICY

In line with the Institutional VISION (Competence, Commitment and Team work), and understanding education as the manifestation of the perfection already in man, the Department of Computer Science and Engineering at SDM College of Engineering & Technology, Dharwad, has redefined its quality policy as under:

Imparting quality education by which:

1. **Character is formed.**
2. **Strength of the *mind* is increased.**
3. **The intellect is expanded, and by which one can *stand on* one's own feet.**

So that our students are:

- ✓ **Acceptable as good citizens and adaptable lifelong learners.**
- ✓ **Acceptable globally in the industries and Premier Institutions of higher studies and research.**

PEOs and POs

Programme Educational Objectives (PEOs) – Programme educational objectives are broad statements that **describe the career and professional accomplishments that the programme is preparing graduates to achieve.**

Programme Outcomes (POs) – Programme Outcomes are narrower statements that describe **what students are expected to know and be able to do upon their graduation**. These relate to the skills, knowledge, and behavior that students acquire in their matriculation through the programme.

Sl. No.	Programme Educational Objectives
Organizational Core Values through Vision: **Competence, Commitment, Equity, Team work and Trust.**	
I	***To build*** technical **competence** by providing necessary background and foundation in Computer Science and Engineering domain, so that students are acceptable globally to industries, premier institutions of higher studies and research.
II	***To create*** awareness of technological trends and tools to provide an experience of developing computing systems through development phases like: inception, elaboration, construction and transition, with higher quality and standards.
III	***To prepare*** students to be **committed** citizen with social and professional concern.
IV	***To prepare*** students to be independent lifelong learner.
V	***To prepare*** students to demonstrate leadership qualities and managerial skills, thus contributing their knowledge globally at various levels of responsibilities in heterogeneous **teams**.

APPROVED PROGRAMME OUTCOMES (POs) and Programme Specific Outcomes (PSOs)

Outcomes are the skills and knowledge which the students have at the time of graduation. This will indicate what student **can do** from subject-wise knowledge acquired during the programme.

PO	Short Title of the PO	Description of the Programme Outcome (PO) as defined by National Board of Accreditation. **Engineering Graduates will be able to:**
PO-1	**Engineering knowledge**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO-2	**Problem analysis**	Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO-3	**Design/development of solutions**	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO-4	**Conduct investigations of complex problems**	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO-5	**Modern tool usage**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO-6	**The engineer and society**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO-7	**Environment and sustainability**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO-8	**Ethics**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO-9	**Individual and team work**:	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO-10	**Communication**	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO-11	**Project management and finance**	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO-12	**Life-long learning**	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.
PROGRAM SPECIFIC OUTCOMES (PSOs) defined by the programme. Baseline- Rational Unified Process (RUP)		
PO-13	**System Inception and Elaboration**	**Conceptualize** the software and/or hardware systems, system components and process/procedures through requirement analysis, modeling /design of the system using various **architectural / design patterns**, standard notations, procedures and algorithms.
PO-14	**System Construction**	**Implement** the systems, procedures and processes using the state of the art technologies, standards, tools and **programming paradigms**.
PO-15	**System Testing and Deployment**	**Verify and validate** the systems, procedures and processes using various **testing and verification techniques and tools**.
PO-16	**Structure and Behavior**	**Manage** the **quality through various product development strategies under revision, transition and operation** through maintainability, flexibility, testability, portability, reusability, interoperability, correctness, reliability, efficiency, integrity and usability to **adapt the system to the changing structure and behavior of the systems /environments**

Credit break-up and Distribution

Sl.No	Category	AICTE Suggested (175)	VTU Proposed (175)	SDMCET (175)
1	Humanities and Social Sciences including Management courses	14	7	8
2	Basic Science courses	27	25	24
3	Engineering Science courses including workshop, drawing, basics of electrical/mechanical/computer etc.	26	19	20
4	Professional core courses	53	83	83
5	Professional Elective courses	20	15	15
6	Open Elective subjects	19	09	09
7	Project work, seminar and internship in industry	16	17(13+3+1)	16 (13+2+1)
8	Mandatory Courses [Environmental Sciences, Induction training, Indian Constitution, Essence of Indian Traditional Knowledge]	Audit	Audit	Audit

Program Articulation Matrix (PAM)

Course Code and Title	PO -1	PO – 2	PO – 3	PO – 4	PO – 5	PO – 6	PO – 7	PO – 8	PO – 9	PO – 10	PO – 11	PO – 12	PO - 13	PO - 14	PO - 15	PO - 16
I/II Semester (Physics Cycle)																
18UMAC100 Engineering Mathematics-I	1.6	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UPHC100/200 Engineering Physics	1.8	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UEEC100/200 Basic Electrical Engineering	3.0	2.0	-	-	-	1.0	1.0	-	-	-	-	-	-	-	-	-
18UCVC100/200 Engineering Mechanics	3.0	3.0	-	-	-	-	-	-	-	-	-	1.0	-	-	-	-
18UMEC100/200 Elements of Mechanical Engineering	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UPHL100/200 Engineering Physics Lab	2.4	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UESL100/200 Basic Engineering Skills Lab	2.0	2.0	3.0	-	-	-	-	-	1.0	1.0	-	-	-	-	-	-
18UHUC100/200 Kannada	-	-	-	-	-	1.0	-	-	-	2.0	-	-	-	-	-	-
18UHUA100/200 Constitution of India & Professional Ethics	-	-	-	-	-	2.0	2.0	1.0	-	-	-	-	-	-	-	-

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
I/II Semester (Chemistry Cycle)																
18UMAC200 Engineering Mathematics-II	1.6	1.7	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UCYC100/200 Engineering Chemistry	2.5	1.5	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UECC100/200 Basic Electronics	1.5	2.0	3.0	-	-	-	-	-	-	-	-	-	-	-	-	-
18UCSC100/200 Problem Solving and Programming in C	1.7	1.7	1.7	-	-	-	-	-	-	-	-	-	-	-	-	-
18UMGC100/200 Engineering Graphics	2.0	1.0	-	-	1.6	-	-	-	-	-	-	-	-	-	-	-
18UCYL100/200 Engineering Chemistry Lab	-	-	2.3	2.3	-	-	-	-	-	-	-	-	-	-	-	-
18UCSL100/200 Problem Solving and Programming in C Lab	2.0	-	1.8	1.0	1.0	-	-	-	-	-	-	-	-	-	-	-
18UHUC101/201 Functional English	-	-	-	-	-	-	-	-	-	2.3	-	-	-	-	-	-
18UHUA101/201 Environmental Science	-	-	-	-	-	1.6	2.0	-	-	-	-	-	-	-	-	-

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
III Semester																
18UMAC300 Engineering Mathematics-III	1.2	1.3	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UCSC300 Digital Electronics	-	3.0	2.5	2.0	3.0	-	-	-	-	-	-	-	2.3	1.0	1.3	-
18UCSC301 Discrete Structures in Computer Science	3.0	2.0	-	1.0	-	-	-	-	-	-	-	-	1.0	-	1.0	-
18UCSC302 Data Structures and Applications	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0
18UCSC303 Computer Organization and Architecture	1.5	1.3	2.3	2.3	-	1.0	-	2.0	-	-	-	-	-	-	-	-
18UCSC304 Introduction to Unix Operating Systems	-	-	-	-	2.0	-	-	-	-	-	-	-	1.7	2.0	2.0	-
18UCSL305 Digital Electronics Laboratory	3.0	2.3	2.0	-	-	-	-	-	-	-	-	1.0	3.0	-	2.0	1.0
18UCSL306 Data Structures and Applications Laboratory	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
IV Semester																
18UMAC400 Engineering Mathematics-IV	1.8	2.0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
18UCSC400 ARM Processor	2.0	2.0	1.5	-	2.0	-	-	-	-	-	-	1.0	1.0	-	1.0	-
18UCSC401 Finite Automata and Formal Languages	2.0	1.9	2.0	1.0	2.0	-	-	-	-	-	-	-	2.0	-	2.0	-
18UCSC402 Object Oriented programming	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.7
18UCSC403 Analysis and Design of Algorithms	1.5	3.0	1.0	-	2.0	-	-	-	-	-	-	-	1.0	-	-	-
18UCSC404 Operating Systems	1.0	2.0	1.0	-	-	-	-	-	-	-	-	-	1.6	2.0	1.0	-
18UCSL405 Object Oriented Programming Laboratory	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.5
18UCSL406 ARM Processor Laboratory	-	2.0	1.0	2.0	2.0	-	-	-	-	-	-	1.0	3.0	2.0	1.0	1.5
18UCSL407 Introductory Project	-	1.0	-	-	1.0	1.0	1.0	-	1.0	-	1.0	1.0	3.0	-	-	-

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
V Semester																
18UHUC500 Mgmt., Entr & IPR																
18UCSC500 Data Communication																
18UCSC501 Database Management Systems																
18UCSC502 Compiler Design and System Software																
18UCSC503 Software Engineering																
18UCSL504 Database Management Systems Lab																
18UCSL505 Compiler Design and System Software Lab																
18UCSL506 Minor Project-1																
18UHUL507 Soft Skills / Aptitude																
18UCSE508 Advanced Object Oriented Programming																
18UCSE509 System Simulation and Modeling																
18UCSE510 Advanced Graph Theory																

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
VI Semester																
18UCSC600 Computer Networks																
18UCSC601 Object Oriented Modeling and Design																
18UCSL602 Computer Networks Lab																
18UCSL603 Industry Oriented Programming Practices Lab																
18UCSL604 Minor Project-2																
18UHUL605 Soft Skills/ Aptitude																
18UCSE606 Unix Systems Programming																
18UCSE607 Digital Image Processing																
18UCSE608 Principles of Programming																
18UCSE609 Data Warehousing and Mining																
18UCSE610 Advanced Data Structures and Algorithms																
18UCSE611 Pattern Recognition																

18UCSE613 Embedded Systems																

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
VII Semester																
18UCSC700 Artificial Intelligence and Machine Learning																
18UCSC701 Advanced Computer Architecture																
18UCSL702 Artificial Intelligence and Machine Learning Lab																
18UCSL703 Major Project Phase – 1																
18UCSL704 Internship																
18UCSE705 Computer Graphics																
18UCSE706 Software Testing																
18UCSE707 Web Technology																
18UCSE708 Ad-hoc Networks																
18UCSE709 Operations Research																
18UCSE710 Multicore Architecture and Programming																

Course Code and Title	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
VIII Semester																
18UCSC800 Distributed Systems																
18UCSL801 Technical Seminar / Independent Study																
18UCSL802 Major Project Phase – 2																
18UCSE803 Cryptography and Network Security																
18UCSE804 Cloud Computing																
18UCSE805 Network Management																
18UCSE806 Data Science																
18UCSE807 Mobile Applications Development																
18UCSE808 Ontology and Semantic Web																

PAM Analysis

Perspective	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PO-13	PO-14	PO-15	PO-16
						[Curriculum Gap Analysis]										
Average of Mapping Levels.	1.91	1.98	1.71	1.66	1.84	1.27	1.50	2.25	1.00	1.77	1.0	1.00	2.13	2.13	1.33	1.62
% Theory Courses	77 20/26	73 19/26	38 10/26	15 4/26	23 6/26	19 5/26	12 3/26	12 3/26	- -	8 2/26	- -	8 2/26	30 8/26	19 5/26	27 7/26	8 2/26
% Laboratory Courses	67 6/9	67 6/9	78 7/9	33 3/9	33 3/9	11 1/9	11 1/9	11 1/9	22 2/9	11 1/9	11 1/9	33 3/9	44 4/9	33 3/9	33 3/9	44 4/9
% Credits	86 76.5/89	84 74.5/89	51 45/89	20 17.5/89	26 23.5/89	9 8/89	5 4/89	10 8.5/89	2 2/89	3 3/89	1 1/89	11 10/89	39 34.5/89	26 23.5/89	33 29.5/89	15 13/89

Note: The above calculations pertain to the **semester 1 to 4 syllabus contents ONLY**. Contents of the courses of remaining semesters will be added based on previous Scheme of 200 credits/ Contextual relevance/market requirements and also **various POs that need to be strengthened[Identified in 200 credits syllabus]**.

FIRST_Year_Scheme

I Semester B.E.

Physics cycle

Course Code	Course Category	Course Title	Teaching		Examination				
					CIE	Theory (SEE)		Practical(SEE)	
			L-T-P (Hrs/Week)	Credits	Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC100	BS	Engineering Mathematics-I	3 - 1 - 0	4	50	100	3	-	-
18UPHC100	BS	Engineering Physics	3 - 1 - 0	4	50	100	3	-	-
18UEEC100	ES	Basic Electrical Engineering	3 - 0 - 0	3	50	100	3	-	-
18UCVC100	ES	Engineering Mechanics	3 - 0 - 0	3	50	100	3	-	-
18UMEC100	ES	Elements of Mechanical Engineering	2 - 0 - 0	2	50	100	--	-	-
18UPHL100	BS	Engineering Physics Lab	0 - 0 - 2	1	50	--	--	50	3
18UESL100	ES	Basic Engineering Skills Lab	0 - 0 - 3	1	50	--	--	50	3
18UHUC100	HU	Kannada	2 - 0 - 0	1	50	50	2		
18UHUA100	HU	Constitution of India & Professional Ethics	2 - 0 - 0	Audit	100	--	--	--	--
Total			**18 - 2 -5**	**19**	**500**	**550**		**100**	

Chemistry cycle

Course Code	Course Category	Course Title	Teaching		Examination				
					CIE	Theory (SEE)		Practical (SEE)	
			L-T-P (Hrs/Week)	Credits	Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC100	BS	Engineering Mathematics-I	3 - 1 - 0	4	50	100	3	-	-
18UCYC100	BS	Engineering Chemistry	3 - 1 - 0	4	50	100	3	-	-
18UECC100	ES	Basic Electronics	3 - 0 - 0	3	50	100	3	-	-
18UCSC100	ES	Problem Solving & Programming in C	4 - 0 - 0	4	50	100	3	-	-
18UMGC100	ES	Engineering Graphics	2 - 0 - 2	3	50	--	-	50	3
18UCYL100	BS	Engineering Chemistry Lab	0 - 0 - 2	1	50	--	--	50	3
18UCSL100	ES	Problem Solving & Programming in C Lab	0 - 0 - 2	1	50	--	--	50	3
18UHUC101	HU	Functional English	2 - 0 -0	1	50	50	2	--	--
18UHUA102	HU	Environmental Science	2 - 0 - 0	Audit	100	--	--	--	--
Total			**19 - 2 - 6**	**21**	**500**	**450**		**150**	

II Semester B.E.

Physics cycle

Course Code	Course Category	Course Title	Teaching		Examination				
					CIE	Theory (SEE)		Practical(SEE)	
			L-T-P (Hrs/Week)	Credits	Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC200	BS	Engineering Mathematics-II	3 - 1 - 0	4	50	100	3	-	-
18UPHC200	BS	Engineering Physics	3 - 1 - 0	4	50	100	3	-	-
18UEEC200	ES	Basic Electrical Engineering	3 - 0 - 0	3	50	100	3	-	-
18UCVC200	ES	Engineering Mechanics	3 - 0 - 0	3	50	100	3	-	-
18UMEC200	ES	Elements of Mechanical Engineering	2 - 0 - 0	2	50	100	--	-	-
18UPHL200	BS	Engineering Physics Lab	0 - 0 - 2	1	50	--	--	50	3
18UESL200	ES	Basic Engineering Skills Lab	0 - 0 - 3	1	50	--	--	50	3
18UHUC200	HU	Kannada	2- 0 - 0	1	50	50	2		
18UHUA200	HU	Constitution of India & Professional Ethics	2- 0 - 0	Audit	100	--	--	--	--
Total			**18 - 2 -5**	**19**	**500**	**550**		**100**	

Chemistry cycle

Course Code	Course Category	Course Title	Teaching		Examination				
					CIE	Theory (SEE)		Practical (SEE)	
			L-T-P (Hrs/Week)	Credits	Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC200	BS	Engineering Mathematics-II	3 - 1 - 0	4	50	100	3	-	-
18UCYC200	BS	Engineering Chemistry	3 - 1 - 0	4	50	100	3	-	-
18UECC200	ES	Basic Electronics	3 - 0 - 0	3	50	100	3	-	-
18UCSC200	ES	Problem Solving & Programming in C	4 - 0 - 0	4	50	100	3	-	-
18UMGC200	ES	Engineering Graphics	2 - 0 - 2	3	50	--	-	50	3
18UCYL200	BS	Engineering Chemistry Lab	0 - 0 - 2	1	50	--	--	50	3
18UCSL200	ES	Problem Solving & Programming in C Lab	0 - 0 - 2	1	50	--	--	50	3
18UHUC201	HU	Functional English	2 - 0 -0	1	50	50	2	--	--
18UHUA202	HU	Environmental Science	2 - 0 - 0	Audit	100	--	--	--	--
Total			**19 - 2 - 6**	**21**	**500**	**450**		**150**	

18UCSC100/200	Problem Solving and Programming in C	(3-2-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This is a 4 credit course, to be covered in 52 hours for UG program at introductory level for all branches of Engineering. It focuses on the following learning results:

- Developing the problem solving skills that can be applied to problems in different areas which enables students to take-up subsequent course work and professional career.
- Provides a comprehensive study of the features of C programming language.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Design** a solution by **analyzing** the given problem scenario and **represent** it using algorithm / flowchart. [Usage] [BL-3]	-	1,2,3	-
CO-2	**Explain** the C language primitives, language principles and use them in writing simple programs.[Familarity] [BL-2]	-	1,2,3	-
CO-3	**Write** a C program using proper control structures to solve simple problems. [Usage] [BL-3]	-	1,2, 3	-
CO-4	**Write** a C program using arrays, and strings to solve simple problems. [Usage] [BL-3]	-	2, 6	-
CO-5	**Explain** the usage of pointers and the need for writing modular programs and **demonstrate** its use in writing programs. [Usage] [BL-3]	-	-	1,2,3
CO-6	**Write** a C program using structures and unions, to solve simple problems. [Usage] [BL-3]			1,2,3
CO-7	**Write** C program using file processing features and libraries to solve simple problems. [Usage] [BL-3]			1,2,3

PO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.7	1.7	1.7	-	-	-	-	-	-	-	-	-	-	-	-	-

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Flow-Chart and Algorithm:** Solving various scientific, engineering and business related problems of varying complexity. **Fundamentals of C Programming Language:** Program structure and execution. Character set, data types, operators, type conversion, expression evaluation. Input and output statements. **Decision making and Branching:** if statement and its different forms, switch statement.	10	2	1 2 3	1	1. **Class Test/ Quiz** (CO-1,2,3) 2. **Written test**- CIE / SEE (CO-1,2,3) 3. **CTA – Assignment** on writing programs using switch statement (CO-3).
2	**Decision making and Looping:** loops and their behavior – entry and exit controlled loops, conditional and unconditional jump statements, Nested loops.	6	2	3	2	1. **Class test / Quiz** (CO-3) 2. **Written test**- CIE/SEE (CO-3) 3. **CTA – Assignment** on writing programs using nested loops. (CO-3)
3	**Arrays:** Single and multidimensional arrays, advantages and disadvantages of arrays, searching and sorting	8	2	4	2	1. **Written test** – CIE / SEE (CO-4)
4	**Strings:** Definition, Different ways of reading and printing strings, string handling functions, applications. **Modular Programming:** Declaration, definition and use	10	2	4 5	2, 3	1. **Written test**- CIE / SEE (CO-4,5)

	of functions, passing parameters to function.					
5	**Building Blocks of Data Structure:** structures, unions, pointer and file operations	**8**	**2**	**5,6,7**	**3**	**1. Written test-** CIE / SEE **(CO-6,7)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						**1.** Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. **2. Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. **3. Group Activity - Development of an application** using concepts covered as a course project

Reference Books:

1) E Balagurusamy, "Programming in ANSI C", 6th Edition, Tata McGraw Hill, 2012.
2) Brian W Kernighan & Dennis M Ritchie, "The C programming language", 2nd Edition, Prentice-Hall India, 2004.
3) R.G.Dromey., "How to solve it by Computer", Prentice-Hall India, 2008
4) B A Forouzan and R F Gilberg, "Computer Program: A structured programming approach using C", 3rd Edition, Thomson Learning, 2005
5) Brain W. Kernighan and Rob Pike, "The Practice of Programming", Pearson Education Inc. 2008.

18UCSL100/200 Problem Solving and Programming in C Laboratory (0-0-2) 1

Contact Hours: 36 Hrs

Course Learning Objectives (CLOs):

This is a 1 credit laboratory course, to be covered in 36 hours for UG program at introductory level for all branches of Engineering. It focuses on the following learning results through practice:

- Conceptualization of the solutions for the given simple problems.
- Representation of the solutions using algorithm and flow chart.
- Writing modular C program to solve simple problems.
- Practicing coding and debugging standards to understand maintainability, testability and other quality parameters.

Course outcomes (COs):

COs	Description of Course Outcomes: At the end of the course, the student will be able to:	Substantial (3)	Moderate (2)	Low (1)
CO-1	**Design** a solution by **analyzing** the given problem scenario and **represent** it using algorithm/flowchart.	-	1,2,3	-
CO-2	**Explain** the C language primitives, language principles and **use** them in writing simple programs.	-	1,2,3	-
CO-3	**Write a** C program using proper control structures to solve simple problems.	-	1,2,3	-
CO-4	**Write a** C program using arrays, and strings to solve simple problems.	-	1,2,3	-
CO-5	**Explain** the need for writing modular programs and **demonstrate** its use in writing programs.	-	-	1,2,3
CO-6	**Write** C program using structures and unions, to solve simple problems.	-	-	1,2,3
CO-7	**Write a** C program using file processing features and libraries to solve simple problems.	-	-	1,2,3
CO-8	**Explain** the need for pointers and **Write** programs using pointers to solve simple problems.	-	-	1,2,3

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.7	1.7	1.7	-	-	-	-	-	-	-	-	-	-	-	-	-

Working Platform: Linux Operating System

Expected Coding Practices:

1. Use of Good Programming practices: Declaration of variables, Indentation, Documentation, Simplicity of logic, Efficiency of logic, uniformity etc.
2. Generic and Reusable code.
3. Inclusions of exceptional cases.
4. Better usability

Course Contents:

Programming exrcises of varying complexity, to meet the learning results stated in course outcomes for this course.

Books:

1. E Balagurusamy, "Programming in ANSI C", 6/E, TMH 2012.
2. Brian W Kernighan & Dennis M Ritchie, "The C programming language", 2/E, PHI 2004.
3. R G Dromey, "How to solve it by computer", PHI 2008.
4. B A Forouzan and R F Gilberg, "Computer Program: A structured programming approach using C", 3/E, Thomson Learning, 2005
5. Brain W. Kernighan and Rob Pike, "The Practice of Programming", Pearson Education Inc. 2008.

III Semester

Course Code	Course Category	Course Title	Teaching		Examination				
			L-T-P (Hrs/Week)	Credits	CIE	Theory (SEE)		Practical (SEE)	
					Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC300	BS	Engg.Mathematics-III	**3-0-0**	3	50	100	3	-	-
18UCSC300	PC	Digital Electronics	**4-0-0**	4	50	100	3	-	-
18UCSC301	PC	Discrete Structures in Computer Science	**3-2-0**	4	50	100	3	-	-
18UCSC302	PC	Data Structures and Applications	**4-0-0**	4	50	100	3	-	-
18UCSC303	PC	Computer Organization and Architecture	**3-0-0**	3	50	100	3	--	--
18UCSC304	PC	Introduction to Unix Operating Systems	**2-0-2**	3	50	100	3	--	--
18UCSL305	PC	Digital Electronics Laboratory	**0-0-3**	1.5	50	--	--	50	3
18UCSL306	PC	Data Structures and Applications Laboratory	**0-0-3**	1.5	50	--	--	50	3
Total			**19-2-8**	**24**	**400**	**600**	-	**100**	-

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination
L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks

III Semester

18UMAC300 **Engineering Mathematics - III** **(3-0-0) 3**

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To have an insight into Laplace transforms, Fourier series, Fourier Transforms, Difference equations and Z-transforms.
- To develop the proficiency in variational calculus.
- To solve ODE's arising in engineering applications, using numerical methods.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Transform** the given function using Laplace/Fourier transforms depending on the nature of engineering applications.	-	-	1
CO-2	**Express** periodic function as a Fourier series and obtain the various harmonics of the Fourier series expansion for the given numerical data.	-	-	1,2
CO-3	**Solve** difference equations using Z-transform.	-	-	1
CO-4	**Solve** first and second order differential equations arising in engineering problems using single step and multistep numerical methods.	-	1,2	-
CO-5	**Determine** the extremals of functional using calculus of variations and solve problems arising in engineering.	-	-	1,2

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.2	1.3	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Laplace Transforms:** Definition and Properties, Laplace transform of elementary functions, Laplace transforms of Periodic functions and unit-step functions – problems. Inverse Laplace Transforms: Problems, Convolution theorem (without proof), to find the inverse Laplace transform and problems, solution of linear differential equations using Laplace transform	8	-	1	1	1. Class Test/ Quiz (CO-1) 2. Written test- CIE / SEE (CO-1)
2	**Fourier series:** Periodic functions, Dirichlet's condition, Fouries series of periodic functions of period 2π and arbitrary period, Half-range Fourier series, Practical harmonic analysis, examples from engineering field.	8	-	2	1	1. Class test / Quiz (CO-2) 2. Written test- CIE/SEE (CO-2)
3	**Fourier Transforms**: Infinite Fourier transforms, Fourier sine and cosine transforms, Inverse Fourier transforms, Simple problems. **Z-transforms and Difference Equations:** Z-transform – definition, Standard Z-transforms, Damping and shifting rules, Initial value and Final value theorems (without proof) with problems, Inverse Z-transform, Simple problems, Difference equations – basic definition, Application of Z-transform to solve Difference equation.	8	-	3	2	1. Class test / Quiz (CO-3) 2. Written test – CIE / SEE (CO-3)

4	**Numerical Solutions of Ordinary Differential Equations (ODE's)**: Numerical solution of ODE's of first order and first degree – Taylor's series method, Modified Euler's method, Runge-Kutta method of fourth order, Milne's predictor and corrector method, (No derivations of formulae). Problems.	7	-	4	3	1. **Class test / Quiz** (CO-4) 2. **Written test**- CIE / SEE (CO-4)
5	**Numerical Solution of Second Order ODE's:** Runge-Kutta method and Milne's predictor and Corrector method. (No derivations of formulae). **Calculus of Variations:** Variation of function and functional, variational problems, Euler's equation (without proof), Geodesics (plane), hanging chain problems.	8	-	5	3	1. **Class test / Quiz** (CO-5) 2. **Written test**- CIE / SEE (CO-5)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. B. S. Grewal, Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2017.
2. Kreyszig E, Advanced Engineering Mathematics, 10th Edition, John Wiley & Sons, 2016.
3. Srimanta Pal et al Engineering mathematics, 3rd Edition, Oxford Univeristy Press, 2016.
4. C.Ray Wylie, Louis C Barrett, Advanced Engineering Mathematics, 6th Edition, McGraw-Hill Book Co, New York, 1995.
5. S.S.Sastry, Introductory Methods of Numerical Analysis, 4th Edition, Prentice Hall of India, 2010.
6. B.V.Ramana, Higher Engineering Mathematics, 11th Edition, Tata McGraw-Hill, 2010.
7. N.P.Bali and Manish Goyal, A Text Book of Engineering Mathematics, 7th Edition, Laxmi Publishers, 2014.

8. Veeraranjan T, Engineering Mathematics for First Year, Tata McGraw-Hill, 2008.
9. Thomas G.B. and Finney R.L., Calculus and Analytical Geometry, 9th Edition, Pearson, 2012.

18UCSC300	Digital Electronics	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To introduce the fundamental principles of digital electronics commonly used in Computer Science and Engineering.
- To facilitate them to gain experience with the design of logic devices.
- To provide the student with an understanding of basic digital electronics abstractions on which analysis and design of electronic circuits/systems are based and the capability to model and analyze complex circuits.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Solve** the given boolean expression to obtain simplified optimal solutions.	2	4	-
CO-2	**Conceptualize and solve** the given real time application by employing suitable reduction techniques.	13	3	15
CO-3	**Design** combinational subsystems to perform different logical and arithmetic operations.	3	15	13
CO-4	**Design** sequential circuits for the given problem statement.	3	13	-
CO-5	**Design** a finite state machine by **modeling** different states of a system in a given problem scenario.	13	3	15
CO-6	**Use** MSI chips to **build system** for the given real time application.	5	4	14

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	2.5	2.0	3.0	-	-	-	-	-	-	-	2.3	1.0	1.3	-

Prerequisites: Knowledge of Basic Electronics.

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction:** Revision of logic gates. Definitions for Digital Signals, Digital Waveforms, Digital Logic, 7400 TTL Series Working of Op-amp, Non-linear applications – Comparator, Schmitt Trigger. **Combinational Logic Circuits:** Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs, Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method, Hazards	8	3	1,2	1	1. **Online Quiz** (CO-1) 2. **Written test**- CIE / SEE (CO-1) 3. **CTA** – Assignment on designing combinational logic circuits.
2	**Data-Processing Circuits:** Multiplexers, De-multiplexers, 1-to-16 Decoder, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays. **Clocks, Flip-Flops:** Clock Waveforms, TTL Clock, Clocked D FLIP-FLOP, Edge-triggered D FLIP-FLOP, Edge-triggered JK FLIP-FLOP, FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs,	8	4	3	1,2	1. **Written test** - CIE/SEE (CO-3) 2. **Quiz** (CO-3)

	Analysis of Sequential Circuits.					
3	**Registers:** Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers. **Counters:** Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Pre-settable Counters, Counter Design as a Synthesis problem.	8	4	4	2	1. **Written test** – CIE / SEE (CO-4) 2. **CTA – Assignment** on designing counters. (CO-4)
4	**Design of Synchronous Sequential Circuits:** Synchronous Sequential Circuit: Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and Circuit Diagram, Algorithmic State Machine, State Reduction Technique.	7	2	5	3	1. **Written test**- CIE / SEE (CO-5) 2. **CTA – Assignment** on designing different models. (CO-5)
5	**Design of Asynchronous Sequential Circuits:** Analysis of Asynchronous Sequential Circuit, Problems with Asynchronous Sequential Circuits, Design of Asynchronous Sequential Circuit. **D/A Conversion and A/D Conversion:** Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion, A/D Accuracy and Resolution.	12	-	5	3	1. **Written test**- CIE / SEE (CO-5)

Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. **Group Activity - Development of an application** using concepts covered as a course project **(CO – 6)**

Reference Books:

1. Donald P Leach, Albert Paul Malvino and Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
3. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning, 2004.
4. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.

18UCSC301	Discrete Structures in Computer Science	(3-2-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The basic terminologies of mathematical and logical reasoning, functions, and relations associated with its properties and corresponding practical examples.
- Various counting principle methods to solve complex problems in combinatorics.
- Demonstration with examples, the basic terminologies of graphs and its types.
- Identify the applications of mathematical structures in other fields of computer science such as data structures and algorithms, databases, networks, operating systems etc.

Course outcomes (COs):

CO	Description of the Course Outcome At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Verify** the correctness of an argument using various techniques and strategies.	1	2	4, 15
CO-2	**Solve** problems using counting techniques and combinatorics.	1	2	4, 15
CO-3	**Solve** the problems on different types of functions, relations, and Generating functions.	1	2	4, 13
CO-4	**Solve** the problems pertaining to graphs and related discrete structures.	1	2	4, 13
CO-5	**Explain** the concepts and properties of algebraic structures such as groups and coding theory.	1	2	4, 15

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.0	-	1.0	-	-	-	-	-	-	-	-	1.0	-	1.0	-

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Fundamentals of Logic:** Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference, The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems	9	3	1	1	1. **Class Test/ Quiz** (CO-1) 2. **Written test**- CIE / SEE (CO-1) 3. **CTA – Assignments** based on **logical implications** and **Proofs** for the given application scenarios. (CO-1)
2	**Fundamental Counting:** The Rules of Sum and Product, Permutations and Combinations, The Binomial coefficients, The Pigeon-hole Principle. **Relations:** Cartesian Products and Relations, Properties of Relations, Equivalence Relations and Partitions	8	3	2 3	1	1. **Class Test / Quiz** (CO-2, 3) 2. **Written test**- CIE / SEE (CO-2, 3) 3. **CTA – Assignments** based on **binomial coefficients** and **Equivalence relations** for the given application scenarios. (CO-1)
3	**Functions:** Definition, Plain and One-to-One, Onto Functions, Function Composition, Inverse Functions, Directed Graphs, Hasse Diagrams.	7	3	-	2	1. **Written test** – CIE / SEE (CO-3)

						2. **CTA – Assignment** on Hasse diagrams
4	**Generating Functions:** Definitions and examples, Exponential Gnerating Functions. **Recurrence Relations:** Recursive definitions. First Order Linear Recurrence Relations, Second order linear homogenious recurrence retelation with constant coefficients.	7	3	3 3	2 3	1. **Class Test / Quiz (CO- 3)** 2. **Written test**- CIE / SEE **(CO-3)** 3. **CTA – Assignments** on Generating Functions and Recurrence relations.
5	**Graphs:** Elements of graph theory, Graphs and its properties, Directed graphs, Subgrapgh, Complements, Planar graphs, Euler Graph, Hamiltonian Graphs, Graph Coloring, Representation of graphs, Trees. **Groups:** Definitions, Examples, and Elementary Properties, Homomorphisms, Isomorphisms	7	2	4 5	3	1. **Class Test / Quiz (CO- 3)** 2. **Written test**- CIE / SEE **(CO-3)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education. 2006.
2. Kenneth H. Rosen: Discrete Mathematics and its Applications, 7th Edition, McGraw Hill, 2012.
3. Kolman B & Busby R C, "Discrete and Mathematical Structures for Computer Science", 5/E, Prentice Hall of India.
4. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

18UCSC302	Data Structures and Applications	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Working of various basic data structures and their implementation.
- Implementation issues of data structure in programming language.
- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3, 15, 16
CO-2	**Write** programs to solve problems using files.	-	14	1,3, 15, 16
CO-3	**Write** programs to solve simple problems using stack and **explain** its working principles.	-	14	1,3, 15, 16
CO-4	**Write** programs to solve problems using queue and **explain** its working principles.	-	14	1,3, 15, 16
CO-5	**Write** programs to solve problems using Linked Lists and **explain** its working principles.	-	14	1,3, 15, 16
CO-6	**Write** programs to solve problems using trees and **explain** its working principles.	-	14	1,3, 15, 16
CO-7	**Write** programs to solve problems using Hashing and **explain** its working principles.	-	14	1,3, 15, 16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Prerequisites: Problem Solving skills and knowledge of Programming in C language.

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Structures, unions and Pointers:** Motivation for using structures. Pointer, access data from memory through pointer, pointer to structures. Motivation for dynamic memory requirement. Realizing arrays using pointer and dynamic memory allocation. Importance of memory management during allocation and de-allocation of memory.	10	-	1,2	1	1. Class Test/ Quiz (CO-1,2) 2. Written test- CIE / SEE (CO-1,2)
2	**Stack:** Realization of stack and its operations using static and dynamic structures. Application of stack in converting an expression from infix to postfix and evaluating a postfix expression. Heterogeneous stack using Unions. Applications of Stacks.	10	-	3	1	1. Class test / Quiz (CO-3) 2. Written test- CIE/SEE (CO-3)
3	**Queues:** Realization of queues (FIFO, Double-ended queue, Priority queue) and its operations using static and dynamic data structures, Applications of Queues.	10	-	4	2	1. Class test / Quiz (CO-4) 2. Written test- CIE/SEE (CO-4)
4	**Lists:** Constructing dynamic data structures using self-referential structure (using the same realized linked Lists), operations on lists. Doubly Linked list.	12	-	5	2,3	1. Class test / Quiz (CO-5) 2. Written test- CIE/SEE (CO-5)

	Application of Lists in sorting.					
5	**Trees:** Types of trees and their properties, Realization of trees using static and dynamic data structures. Operations on Binary trees and their application in searching (BST and AVL Tree), Binary heap as priority queues, Applications of Trees. **Hash Table:** Realizing effective hash table with proper data stricture and hash function, its application.	**10**	-	**6** **7**	**3**	1. **Class test / Quiz** (CO-6,7) 2. **Written test**- CIE/SEE (CO-6,7)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. **Group Activity - Development of an application** using concepts covered as a course project

Reference Books:

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein: Data Structures using C and C ++, Pearson Education / PHI, 2006
2. E. Balagurusamy: Programming in ANSI C, 7th Edition, Tata McGraw-Hill, 2016.
3. Behrouz A. Forouzan and Richard F. Gilberg: Computer Science: A Structured Programming Approach Using C, 2nd Edition, Cengage Learning, 2003.

18UCSC303	Computer Organization and Architecture	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Basics of sub systems of a computer, their organization, structure and operation.
- Basic concept of programs as sequences of machine instructions.
- Operational aspects of I/O devices and standard I/O interfaces.
- Memory hierarchy and concept of virtual memory.
- Arithmetic and logical operations with integer and floating-point operands.
- Organization of a simple processor, pipelined processor and other computing systems.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the basic working principles of various sub-systems of a computer system.	-	-	1, 2, 8
CO-2	**Explain** the working principles of different sub systems, such as processor, Input/output, and memory.	-	3,8	1, 2, 4, 6
CO-3	**Design** the required memory bank using basic memory units.	-	3	-
CO-4	**Explain** hardwired control and micro programmed control, pipelining, embedded and other computing systems.	3,4	1	2
CO-5	**Design** simple arithmetic and logical units for a given operational features.	4, 8	1, 2, 3	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.5	1.3	2.3	2.3	-	1.0	-	2.0	-	-	-	-	-	-	-	-

Prerequisites: Knowledge of Digital Electronics and Programming lanhuage

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools / Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Basic Structure and Machine Instructions:** Basic Operational Concepts, Bus Structures, Performance - Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. **Machine Instructions and Programs:** Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions.	8	-	1	1	1. **Class Test/ Quiz** (CO-1) 2. **Written test**- CIE / SEE (CO-1) 3. **CTA** – Assignment on **Number Systems** and **Airthmetic.** (CO-1)
2	**Input / Output Organization:** Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.	8	-	2	1	1. **Class test / Quiz** (CO-2) 2. **Written test**- CIE/SEE (CO-2) 3. Demonstration of **Physical Devices.** (CO-2)
3	**Memory System:** Basic Concepts, Semiconductor RAM and ROM Memories, Speed, Size and Cost, Cache	8	-	3	2	1. **Class test / Quiz** (CO-3) 2. **Written test**- CIE/SEE

	Memories – Mapping Functions, Replacement Algorithms, Performance Considerations.					(CO-3)
4	**Arithmetic:** Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations.	8	-	4	2	1. **Class test / Quiz** (CO-4) 2. **Written test**- CIE/SEE (CO-4) 3. **CTA** – Assignment on **Memory Interfaces** and **Airthmetic Hardware Units** (CO-3,4)
5	**Basic Processing Unit:** Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Pipelining, Embedded Systems and Large Computer Systems: Basic Concepts of pipelining, Examples of Embedded Systems, Processor chips for embedded applications, Simple Microcontroller.	7	-	5	3	1. **Class test / Quiz** (CO-5) 2. **Written test**- CIE/SEE (CO-5)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc....						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1) Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5/E, TMH, 2011.
2) William Stallings, Computer Organization & Architecture, 9/E, PHI, 2012.
3) Vincent P. Heuring & Harry F. Jordan, Computer Systems Design and Architecture, 2/E, Pearson education, 2004.

18UCSC304	Introduction to UNIX Operating System	(2-0-2) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To provide the student with an exposure on UNIX platform so that various other domain specific project activities can be performed with ease and comfort.
- To provide the student with an exposure on the structure and working principles of UNIX operating systems at introductory level, focusing on OS services, commands and scripting language for administration of UNIX operating system.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the structure and working principles of UNIX operating system.	-	-	13
CO-2	**Use** different UNIX commands and System Calls to perform system administration and user specified tasks.	-	14,15	-
CO-3	**Write** shell scripts to perform different system administrative task.	-	5,13,14,15	-
CO-4	**Write** awk scripts to perform different system administrative task.	-	5,13,14,15	-

PO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	-	-	-	2.0	-	-	-	-	-	-	-	1.7	2.0	2.0	-

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction to Operating Systems:** Computer System organization/ architecture; Operating System structure; Operating System operations; Introduction to Process management; Memory management; Storage management; Protection and security; Distributed system; Computing environments. Operating System Services; System calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; System boot	4	-	1	1	1. **Written test-** CIE / SEE (CO-1) 2. **CTA** – Assignment on Study of UNIX OS evolution.
2	**Unix System Architecture, commands and System calls:** Unix System structure, Commands for performing various activities related to process, files, directories, devices, password protection, vi editors, and other administrative task. Daemon process. System calls related to process, files and directories.	5	3	1,2	1,2	1. **Class test / Quiz** (CO-1,2) 2. **Written test-** CIE/SEE (CO-1,2) 3. Programming Exercises on use of System Calls (CO-2). Ex: • Process creation using fork() and vfork(). Use of exec system call • Directory Listing • Displaying file

						attributes • Implementation of Unix command: ls -l.
3	**Filters in Unix:** Paginating Files, head: Displaying the Beginning of a File, tail: Displaying the End of a File, cut: Splitting a File Vertically, paste: Pasting Files, sort Ordering a File, uniq Locate Repeated and Non repeated Lines, tr Translating Characters, An Example: Displaying a Word count List. grep Searching for a Pattern, Basic Regular Expressions (BRE) – An Introduction, Extended Regular Expressions (ERE) and egrep.	6	3	2	2	1. **Class test / Quiz (CO-2)** 2. **Written test-** CIE/SEE **(CO-2)** 3. CTA – Command level Exercises on Filters **(CO-2)**
4	**Shell Programming:** Environment Variables, Aliases (bash), Command History (bash). Shell Scripts, read and readonly commands, Using Command Line Arguments, exit and Exit Status of Command, The Logical Operators && and || Conditional Execution, The if Conditional, Using test and [ ] to Evaluate Expressions, The case Conditional, expr: Computation and String Handling, $0: Calling a Script by Different names, while: Looping, for: Looping with a List, set and shift: Manipulating the Positional Parameters.	6	3	3	2,3	1. **Class test / Quiz (CO-3)** 2. **Written test-** CIE/SEE **(CO-3)** 3. **CTA** – Programming Exercises on **Shell Scripts (CO-3)**

5	**Awk Scripting Language:** awk program line and script structure, awk's operational mechanism, Records and fields, special variables $0, $1, $2, etc., patterns, The BEGIN and END, Variables, built in variables, built in functions, length, split, getline, print, printf, sprintf, index, system, substr, etc., control structures, operators in awk, associative arrays, writing simple awk scripts, Running awk scripts from the shell.	6	3	-	-	1. **Class test / Quiz** (CO-4) 2. **Written test**- CIE/SEE (CO-)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: Course teachers' assessment (CTA) consists of study of various commands, mastery over vi editors, structure of UNIX operating system, in depth writing shell and awk scripts for simple administrative task. The report is to be submitted by individual students. Students are expected to spend approximately 26 hours on practice based learning and its evaluation.

Reference Books:

1) Sumitabha Das UNIX Concepts and Applications, Third edition, Tata McGraw Hill, 2003
2) Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
3) Behrouz A. Forouzan and Richard F.Gilberg UNIX and Shell Programming A Text book, Thomson, edition 2003.

18UCSL305	Digital Electronics Laboratory	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Combinational circuit design and simplification techniques used for realizing them.
- Sequential circuit design and working of a basic storage element.
- Simple circuits using passive elements (resistors, capacitors, inductors).

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Design and implement** combinational circuit for the problem scenarios.	1,13	2,3	12
CO-2	**Design and implement** sequential circuit for problem scenarios.	1,13	2,3	12
CO-3	**Design and implement** an application circuit to simulate given problem.	1,2	15	16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.3	2.0	-	-	-	-	-	-	-	-	1.0	3.0	-	2.0	1.0

Pre-requisites: Basic Electronics.

List of Experiments:

1 Study and verification of the truth table of various logic gates.

2 Realization of Boolean Functions:

i) Simplify the given Boolean expression and to realize it using Basic gates and Universal gates.
ii) Realize the adder and subtract or circuits using basic gates and universal gates.
iii) Simplify given Boolean expression using Map Entered Variable (MEV) technique and realize the simplified expression using 8:1 Multiplexers.
iv) To implement given Boolean function using decoders.
v) To design and realize the code converters using basic gates.
vi) To realize Two Bit Comparator using basic gates

3 Flip-Flops (Sequential Circuits):

i) To realize flip-flop conversions.
ii) Applications Flip-Flops:
   a) To design and implement mod-n synchronous counter.
   b) Design and implement a mod-n asynchronous counter.
   c) To realize and study Shift Registers / Ring counter and Johnson counter.

Reference Books:

1. Donald P Leach, Albert Paul Malvino and Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
3. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning, 2004.
4. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.

18UCSL306 Data Structures and Applications Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Realization of fundamental data structures like stacks, queues, linked lists and trees.
- Compare and contrast the benefits of dynamic and static data structure implementations.
- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3, 15, 16
CO-2	**Write** programs to solve problems using files.	-	14	1,3, 15, 16
CO-3	**Write** programs to solve problems using stack.	-	14	1,3, 15, 16
CO-4	**Write** programs to solve problems using queue.	-	14	1,3, 15, 16
CO-5	**Write** programs to solve problems using Linked Lists.	-	14	1,3, 15, 16
CO-6	**Write** programs to solve problems using trees.	-	14	1,3, 15, 16
CO-7	**Write** programs to solve problems using Hashing.	-	14	1,3, 15, 16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Suggested list of term works:

The list of experiments is based on the following concepts:

1) Pointers and Structures/Unions
2) Files
3) Stack
4) Queue
5) Linked Lists
6) Trees
7) Hashing

Reference Books:

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein: Data Structures using C and C ++, Pearson Education / PHI, 2006,
2. E. Balagurusamy: Programming in ANSI C, 7th Edition, Tata McGraw-Hill, 2016.
3. Behrouz A. Forouzan and Richard F. Gilberg: Computer Science: A Structured Programming Approach Using C, 2nd Edition, Cengage Learning, 2003.

IV Semester

Course Code	Course Category	Course Title	Teaching		Examination				
					CIE	Theory (SEE)		Practical (SEE)	
			L-T-P (Hrs/Week)	Credits	Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UMAC400	BS	Engineering Mathematics-IV	**3-0-0**	3	50	100	3	-	-
18UCSC400	PC	ARM Processor	**3-0-0**	3	50	100	3	-	-
18UCSC401	PC	Finite Automata and Formal Languages	**3-0-0**	3	50	100	3	-	-
18UCSC402	PC	Object Oriented Programming	**4-0-0**	4	50	100	3	-	-
18UCSC403	PC	Analysis and Design of Algorithms	**3-0-2**	4	50	100	3	-	-
18UCSC404	PC	Operating Systems	**4-0-0**	4	50	100	3	-	-
18UCSL405	PC	Object Oriented Programming Lab	**0-0-3**	1.5	50	--	-	50	3
18UCSL406	PC	ARM Processor Lab	**0- 0-3**	1.5	50	--	-	50	3
18UCSL407	PC	Introductory Project	**0–0-2**	1	50	--	--	--	--
Total			**20-0-10**	**25**	**450**	**600**		**100**	

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination
L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks

18UMAC400	Engineering Mathematics-IV	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To provide an insight into applications of conformal mapping, integration of complex functions and application of probability distributions in Engineering.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Construct** and **use** the concepts of analytic function to solve the problems arising in Engineering field.	-	-	1
CO-2	**Utilize** conformal transformation and complex integral to transform irregular domain onto a relatively simple domain.	-	1	-
CO-3	**Apply** discrete and continuous probability distributions in analyzing the probability models arising in engineering field.	-	1	-
CO-4	**Make use** of the correlation and regression analysis to fit a suitable mathematical model for the statistical data.		1,2	-
CO-5	**Estimate** the correlation, covariance using joint probability distributions. Also use student's t-distribution, Chi-square distribution as a test of goodness of fit.	-	1,2	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

- Differentiation of Functions
- Integration of Functions.
- Basics of Probability
- Statistical Averages

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Calculus of complex functions:** Review of function of a complex variable, limits, continuity, and differentiability. Analytic functions: Cauchy-Riemann equations in Cartesian and polar forms. Construction of analytic functions: Milne-Thomson method-Problems	7	-	1	1	1. Class Test/ Quiz (CO-1) 2. Written test- CIE / SEE (CO-1)
2	**Conformal transformations:** Introduction. Discussion of transformations: $w = e^z; w = z^2$, $w = z + \frac{1}{z}, z \neq o)$. Bilinear transformations- Problems. **Complex integration:** Line integral of a complex function, Cauchy's theorem and Cauchy's Integral theorem.	8	-	2	2	1. Class Test/ Quiz (CO-2) 2. Written test- CIE / SEE (CO-2)

3	**Probability Distributions:** Review of basic probability theory. Random variables (discrete and continuous), probability mass/density functions. Binomial, Poisson, exponential and normal distributions- problems (No derivation for mean and standard deviation)-Illustrative examples.	8		-	-	1. **Class Test/ Quiz (CO-3)** 2. **Written test**- CIE / SEE **(CO-3)**
4	**Statistical Methods:** Correlation and regression-Karl Pearson's coefficient of correlation and rank correlation-problems. Regression analysis- lines of regression–problems. **Curve Fitting:** Curve fitting by the method of least squares- fitting the curves of the form $= ax + b; y = ax^2 + bx + c; y = ax^b$.	8		-	-	1. **Class Test/ Quiz (CO-4)** 2. **Written test**- CIE / SEE **(CO-4)**
5	**Joint probability distribution:** Joint Probability distribution for two discrete random variables, expectation and covariance. **Sampling Theory:** Introduction to sampling distributions, standard error, Type-I and Type-II errors. Test of hypothesis for means, student's t-distribution, Chi-square distribution as a test of goodness of fit.	8		-	-	1. **Written test**- CIE / SEE **(CO-5)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc....						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 10th Ed.(Reprint) 2016.
2. B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 44th Ed., 2017.
3. Srimanta Pal et al: Engineering Mathematics, Oxford University Press, 3rd Edition, 2016.
4. C. Ray Wylie, Louis C. Barrett : "Advanced Engineering Mathematics", 6th Edition, McGraw-Hill Book Co., New York, 1995.
5. S. S. Sastry, Introductory Methods of Numerical Analysis, Prentice Hall of India, 4th Edition 2010.
6. B. V. Ramana: "Higher Engineering Mathematics" 11th Edition, Tata McGraw-Hill, 2010.
7. N. P. Bali and Manish Goyal : A Text Book of Engineering Mathematics, Laxmi Publishers, 7th Ed., 2014.

18UCSC400	ARM Processor	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Understand the use of interrupts & other advanced concepts related to ARM7
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the features of embedded systems, architecture of ARM7 and applications.	-	1	-
CO-2	**Write** a program using the instruction set of ARM and THUMB state to solve the engineering problems.	-	2,5	13
CO-3	**Explain** the exception, interrupts and interrupt handling schemes and **write** program to solve simple problems.	-	3	-
CO-4	**Explain** the architectural features of LPC2148 microcontrollers.	-	1,2	13,15
CO-5	**Write** a program to interface hardware to LPC2148 microcontrollers.	-	5	3,12

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	1.5	-	2.0	-	-	-	-	-	-	1.0	1.0	-	1.0	-

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**ARM Embedded Systems and ARM Processor Fundamentals:** Evolution of Microcontroller and Microprocessor, The RISC design philosophy, ARM design philosophy, embedded system hardware- AMBA bus protocol, embedded system software- applications. ARM core data flow model, Registers, CPSR-Processor modes, Banked registers. Pipeline- Characteristics.	9	-	1	1	1. **Class Test** (CO-1) 2. **Written test**- CIE / SEE (CO-1) 3. **Assignment** (CO-1)
2	**ARM Instruction Set:** Fundamentals of ARM instructions, Barrel shifter, Classification and explanation of instructions with examples-Data processing, Branch, Load-store, SWI and Program Status.	5	2	1	1	1. **Class test / Quiz** (CO-2) 2. **Written test**- CIE/SEE (CO-2) 3. **CTA** – Writing ALP programs.(CO-2)
3	**Introduction to THUMB and ARM Programming:** Introduction to THUMB, Differences between ARM and THUMB, Register usage in Thumb, ARM Thumb Interworking. General Structure of ARM assembly module, Assembler directives. Simple ALP programs on	6	2	2	2	1. **Written test** – CIE / SEE (CO-3) 2. **CTA** – Writing ALP programs. (CO-3)

	Arithmetic & logical operations, Factorial, string operation, sorting, searching, and Scan.					
4	**Exception and Interrupt handling schemes:** Exception handling- ARM processor exceptions and modes, vector table, exception priorities, link register offsets. Interrupts- assigning interrupts, interrupt latency, IRQ and FIQ exceptions with example- code for enabling and disabling IRQ and FIQ exceptions, Comparison between exception and interrupts. Interrupt handling schemes- nested interrupt handler, non-nested interrupt handler. Basic interrupt stack design.	7	-	3	2,3	1. **Written test**- CIE / SEE **(CO-4)** 2. **CTA – Assignment** on Exception and Interrupt handling schemes. **(CO-4)**
5	**LPC2148 ARM CPU:** Salient features, applications, block diagram, memory mapping. Functional features of Interrupt controller, RTC, USB, UART, I2C, SPI, SSP controllers, watch dog timers and other system control units. **Peripherals - GPIO, PLL & Timers:** Features, Register description with example and Applications.	8	-	4,5	3	1. **Written test**- CIE / SEE **(CO-5)** 2. **CTA – Assignment (CO-5)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. Andrew N. Sloss, ARM System Developer's guide, ELSEVIER Publications, 2016
2. William Hohl, ARM Assembly Language, CRC Press.
3. Steve Furber, ARM System-on-chip Architecture by, Pearson Education, 2012
4. James K. Peckol, Embedded Systems: A Contemporary Design Tool, 2008
5. Jonathan W. Valvano, Brookes / Cole, Embedded Microcomputer Systems, Real Time Interfacing, 1999
6. LPC 2148 USER MANUAL.

18UCSC401	Finite Automata & Formal Languages	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Study abstract computing machines, Language representation techniques, regular expressions, grammar constructions and associated theories and tools to realize formal language.
- Employ finite state machines to solve problems in computing.
- Comprehend the hierarchy of problems arising in the computer sciences.
- Understand the Turing theory and its significance.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Construct** a finite automaton for a given pattern and **explain** its working principles.	-	1,2,3,13	-
CO-2	**Write** regular expressions for given patterns and **explain** different techniques and principles used.	-	1,2,3,13	-
CO-3	**Verify** the properties of given languages using standard procedures and **explain** the language properties.	-	1,15	2, 4
CO-4	**Design** grammar for a given language specification and **explain** the design principles.	-	1,2,3,13	-
CO-5	**Write** lexical analyzer and parser for simple programming constructs using standard compiler writing tools.	-	1,2,3,5	-
CO-6	**Design and verify** pushdown automata for a given language specification and **explain** its underlying working principles.	-	1,2,3,15	-
CO-7	**Design and verify** Turing Machine for a given language specification and **explain** its underlying working principles.	-	1,2,3,15	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	1.9	2.0	1.0	2.0	-	-	-	-	-	-	-	2.0	-	2.0	-

Prerequisites: Knowledge of: Programming Language (any).

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction to Finite Automata:** Structural Representation. The central concepts of Automata theory – Alphabet, Strings & Languages. Finite Automata: Introduction, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Equivalence of NFA and DFA, Applications of Finite automata, FA with Epsilon (ϵ) transitions.	9	-	1	1	1. Class Test (CO-1) 2. **Written test-** CIE / SEE (CO-1) 3. **CTA – Quiz** on solving GATE questions
2	**Regular Expressions and languages:** Regular Expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions. Properties of Regular Languages (RL): Proving Languages not to be Regular. Closure properties of Regular Languages, Decision properties of Regular Languages, Equivalence and Minimization of Automata.	7	-	2,3	1,2	1. Class test (CO-2) 2. **Written test-** CIE/SEE (CO-2) 3. **CTA – Quiz** on solving GATE questions
3	**Context-Free Grammars (CFG) and Languages (CFL):** Context-Free Grammars, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages. **Compiler Writing Tools:** Structure of compiler writing tools like LEX and YACC. Simple programs on LEX and	8	-	4,5	2	1. **Written test** – CIE / SEE (CO-3) 2. **CTA – Write** simple **lex** and **YACC programs** to recognize the patterns

	YACC.					and expressions.
4	**Pushdown Automata (PDA):** Definition of Pushdown Automata, The languages of a PDA, Equivalence of PDA's and CFG'S, Deterministic Pushdown Automata. Properties of Context Free Languages: Normal forms for Context Free Grammar, Pumping lemma for Context Free Languages, Closure properties of Context Free languages.	7	-	6	3	1. **Written test-** CIE / SEE **(CO-4)** 2. **CTA** – Assignment on **Designing** Pushdown Automata for the context free languages.
5	**Turing Machines (TM):** Introduction, Design of Turing Machine, Extensions to Basic Turing Machine.	8	-	7	3	1. **Written test-** CIE / SEE **(CO-5)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1) John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman Introduction to Automata Theory, Languages and Computation, Pearson Education, 3/E, 2013.
2) Elaine A.Rich, Automata, Computability, and Complexity, Pearson Publication, 2007.
3) Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers Principles, Techniques and Tools, Pearson Education, 2/E, 2008.
4) John R. Levine and Tony Mason and Doug Brown UNIX programming tools 2/E, 1992.
5) Peter Linz, An Introduction to Formal Languages and Automata, Narosa Publishing House, 5/E, 2011.
6) John Martin, Introduction to languages and theory of computation, Tata McGraw-Hill, 4/E, 2010.

18UCSC402	Object Oriented Programming	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Prepare** an abstract data type for the given business scenario and **write** simple programs to represent ADT and **use** in the given application scenario.	13	1	-
CO-2	**Write** programs to solve given problem using different reusability features like inheritance and composite objects.	2, 14	1, 16	3
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept to solve conflicts due to interleaved execution of threads and write simple programs.	2, 14	1	3
CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3

CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.7

Prerequisites: Knowledge of any programming language

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction to Object Oriented Paradigm (OOP):** Object Oriented Philosophy, Key Concepts of OOP, Encapsulation, Polymorphism, Inheritance.	9	-	1	1	1. Class Test/ Quiz (CO-1) 2. **Written test**- CIE / SEE (CO-1)
2	**Basic Features of Java - 1:** Introduction to JAVA, Data Types, Variables and Arrays, String Handling in Java, Control Structures.	7	2	1	1	1. Class test / Quiz (CO-1) 2. **Written test**- CIE/SEE (CO-1) 3. **CTA – Writing** simple programs on handling arrays and strings. (CO-1)
3	**Basic Features of Java - 2:** Classes, Objects, Methods, Constructors, Overloading methods, Methods and Classes, Inheritance, Packages and Interfaces.	9	3	2,4,5	2	1. **Written test** – CIE / SEE (CO-2,4,5) 2. **CTA – Writing** programs on method overloading and overriding, implementation of Interfaces. (CO-2,4,5)
4	**Core Features of Java - 1:** Exception Handling, Multi-Threaded Programming, Streams.	6	4	6,7,8	3	1. **Written test**- CIE / SEE (CO-6,7,8) 2. **CTA – Writing** programs incorporating

						exception handling and multi-threaded programming. **(CO-6,7,8)**
5	**Core Features of Java - 2:** AWT and Swings, Applets, Events	**8**	**4**	**9**	**3**	1. **Written test**- CIE / SEE **(CO-9)** 2. **Design** of GUI using AWT features. **(CO-9)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1) Herbert Schildt, Java-The Complete Reference, 9th Edition, Tata McGraw Hill, 2014.
2) Grady Booch, Object-Oriented Analysis and Design with Applications, 3rd Edition, Pearson Education, 2007.

18UCSC403	Analysis and Design of Algorithms	(3-0-2) 4

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Analyze the performance of algorithms.
- Demonstrate familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the importance of algorithmic/mathematical approach in solving different types of problems.	-	1	-
CO-2	**Analyze** time and space complexity for a given algorithm.	2	-	1
CO-3	**Apply and analyze** brute force technique and compare it with other techniques.	2	5	3,13
CO-4	**Apply and analyze** divide and conquer technique and compare it with other techniques.	2	5	3,13
CO-5	**Apply and analyze** greedy technique and compare it with other techniques.	2	5	3,13
CO-6	**Apply and analyze** dynamic programming technique and compare it with other techniques.	2	5	3,13
CO-7	**Apply and analyze** backtracking and branch & bound technique and compare it with other techniques.	2	5	3,13

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.5	3.0	1.0	-	2.0	-	-	-	-	-	-	-	1.0	-	-	-

Prerequisites: Knowledge of: Discrete Mathematics and Data Structures.

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction:** Algorithm, Fundamentals of problem solving, Problem types, Principles of Algorithm Design. Analysis framework, Asymptotic notations, Mathematical analysis of Non recursive algorithms, Recurrence relations; Mathematical analysis of recursive analysis. **Brute force strategy:** Selection Sort, Bubble sort, string matching	9	-	1,2,3	1	1. Class Test/ Quiz (CO-1) 2. **Written test**- CIE / SEE (CO-1)
2	**Divide and Conquer:** Introduction and General method, Binary search, Merge sort, Quick sort, Matrix multiplication using Stressen's Matrix multiplication. **Basic Traversal and Search techniques:** Depth First search, Breadth First Search, connected components, labeling of components, Path.	5	2	4	1 2	1. Class test / Quiz (CO-2) 2. **Written test**- CIE/SEE (CO-2) 3. **CTA – Implentation** of sorting technqiue **(any one),** breadth first search **(BFS),** and depth first search **(DFS).**
3	**Dynamic Programming:** Introduction and General method, Computing a binomial coefficient, Warshall's algorithm, Floyd's algorithm, knapsack problem.	5	3	6	2	1. **Written test** – CIE / SEE (CO-3) 2. **CTA – Implentation** of Warshall's and Floyd's

						algorithms.
4	**Greedy Strategy:** Introduction and General Method, Knapsack problem, Job sequencing with dead-lines, min cost spanning tree (Prim's & Kruskal's), single source shortest path. Huffman Tree	5	2	5	3	1. **Written test-** CIE / SEE **(CO-4)** 2. **CTA – Implentation** of minimum spanning tree algorithms.
5	**Back tracking and Branch and Bound:** Introduction General Method for both strategies Back Tracking: Sum of Sub sets, Knapsack problem, Traveling Sales person (TSP). **Limitations of Algorithm Power:** Lower bound arguments, decision trees, P, NP and NP Complete Problems.	8	-	7	3	1. **Written test-** CIE / SEE **(CO-5)** 2. **CTA – Assignment** on solving sum of subsets and knapsack problems.
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. Anany Levitin, Introduction to the Design and analysis of algorithms, Pearson Education 3rd Edition, 2011

2. Horowitz, Sahani et.al Fundamentals of Computer Algorithms, Galgotia Publication, 2nd Edition, 2004.
3. Marks Allen Weiss, Data Structure and Algorithm Analysis, Pearson Education, 3rd Edition, 2009
4. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest, Introduction to Algorithms, 2nd Edition, PHI 2003.

18UCSC404	Operating System	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs):

The contents of this course deal with the structure and working principles of generic operating systems at introductory level, focusing on process management, memory management, file system and device management. It also focuses on architecture and programming aspects of Linux based OS at fundamental level.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Substantial	Moderate	Low
CO-1	**Explain** the fundamental concepts of operating system and **Write** programs to demonstrate working principles of process/threads, related issues using system calls and standard libraries.	-	13,14	1,15
CO-2	**Compare** different scheduling algorithms.	-	2	3,13
CO-3	**Compare** and contrast various memory allocation strategies.	-	2	3,13
CO-4	**Explain** the structure and working principles of a file organization and **Write** programs to demonstrate the various file operations using system calls.	-	13,14	1,15
CO-5	**Explain** the structure and working principles of secondary storage and issues related to protection/access strategies.	-	13	1
CO 6	**Explain** the architecture and working principles of industry **standard OS**.	-	13	1

PO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	2.0	1.0	-	-	-	-	-	-	-	-	-	1.6	2.0	1.0	-

Prerequisites: Knowledge of Computer Organization, Digital Electronics and Computer Programming at introductory level.

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Process Management:** Process concept; Process scheduling; Operations on processes; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.	10	-	1	1	1. **Class Test/ Quiz (CO-1)** 2. **Written test-** CIE / SEE **(CO-1)**
2	**Process Synchronization:** Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors **Case Study:** UNIX process synchronization	10	-	1	1 2	1. **Class test / Quiz (CO-1)** 2. **Written test-** CIE/SEE **(CO-1)** 3. **CTA-Writing programs** to demonstrate synchronization issues and solutions using p-thread libraries **(CO-1)** 4. **CTA- Group Activity: Course project(CO-1)** Design and implementation of

						protocols using RFCs / Sockets
3	**Deadlocks:** System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Case Study:** UNIX process/thread management and programming related to process creation, conflict management and IPCs using system calls and p-thread libraries.	10		2 3	2	1. Class Test/Quiz **(CO-3)** 2. **Written test** – CIE / SEE.- **(CO-3)**
4	**Memory Management:** Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **Case Study:** UNIX memory management.	10		4	3	1. Class Test/Quiz **(CO-4)** 2. **Written test** – CIE / SEE.- **(CO-4)**
5	**File System, Implementation of File System:** File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management **Case Study:** UNIX file structure and programming on various file operations like creation, listing attributes, directory listing and lock operations. **Secondary Storage Structures & Protection:** Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of	12		5 6	3	1. **Written test**- CIE / SEE **(CO-5)** 2. **CTA-Writing programs** to demonstrate file and directory operations. Ex: Implementation of **ls –l** command, File locking operations etc. - **(CO-6)**

	access matrix, Access control, Revocation of access rights, Capability-Based systems					
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. **Peer learning program** for 1st Years on: Dual booting & installation of UNIX, UNIX architecture, Commands and vi editor

Note: Course Teacher's Assessment (CTA) is to be based on UNIX programming.

Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
2. William Stallings: Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2013.

18UCSL405 Object Oriented Programming Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features, a prerequisite to take course on advanced features of Java Language.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Prepare** an abstract data type for the given business scenario and **write** a program to represent ADT and **use** in the given application scenario.	13	1	-
CO-2	**Write** a program to solve given problem using different reusability features like inheritance and composite objects.	2, 14	1, 16	3
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept, **solve** conflicts due to interleaved execution of threads and **write** programs.	2, 14	1	3

CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3
CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.5

Prerequisites: Knowledge of: Registration/Completion of the course Object Oriented Programming.

Suggested Platforms:

Notepad (Non IDE), IDE (JCreator, Net Beans, Eclipse etc) in Windows OS and Linux OS

All programs should:

1. Be written to realize the Object Oriented Philosophy and core Java features.
2. Be written with Java Naming & Coding conventions and well documented.
3. Handle exceptions.
4. Be tested for all possible scenarios.

Course Contents:

Minimum one exercise to cover each course outcome specified above. Maximum 8 experiments to be completed by each student independently covering all course outcomes defined for this course. Course teacher has to publish list of experiments along with individual outcome for every experiments, on the first day of the semester. Examiner may set any_problem based on the published term work during tests.

Reference Books:

1) Herbert Schildt, Java The Complete Reference 7th Edition, Tata McGraw Hill, 2007.
2) Kathy Sierra & Bert Bates, Head First Java, 2nd Edition, O'Reilly, 2009
3) Patrick Niemeyer & Daniel Leuck, Learning Java, 4th Edition, O'Reilly, 2013
4) Laura Lemay & Charles L. Perkins, Teach Yourself Java in 21 Days, 7th Edition, Sams Publishing, 2016

18UCSL406	ARM Processor Laboratory	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Understand the use of interrupts & other advanced concepts related to ARM7
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Execute** assembly level codes for a given specific problem using ARM processor.	-	2, 4	3,15
CO-2	**Execute** embedded C programs for a given specific problem using ARM processor.	-	4,14	15,16
CO-3	**Implement** programs for interfacing with real world devices such as LCD's Keyboards, DAC, ADC, Relays Motors etc.	13	4,5,16	3,12

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	1.0	2.0	2.0	-	-	-	-	-	-	1.0	3.0	2.0	1.0	1.5

Course Contents:

PART A: Conduct the following experiments to learn ALP using ARM:

- Arithmetic and logical operations
- Interrupts related operations
- Timer related applications.

PART B: Conduct interfacing experiments to learn embedded C for ARM:

- LCD- interfacing
- Stepper Motor Interfacing
- Real time sensors Interfacing
- 7-segment LED interface

Reference Books:

1. Andrew N. Sloss, ARM System Developer's guide, ELSEVIER Publications, 2016
2. William Hohl, ARM Assembly Language, CRC Press.
3. Steve Furber, ARM System-on-chip Architecture by, Pearson Education, 2012
4. James K. Peckol, Embedded Systems: A Contemporary Design Tool, 2008
5. Jonathan W. Valvano, Brookes / Cole, Embedded Microcomputer Systems, Real Time Interfacing, 1999
6. LPC 2148 USER MANUAL.

18UCSL407	Introductory Project	(0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs): This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

Course Outcomes (COs):

CO	Description of the course outcome At the end of the course student should be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Identify** the societal problems	-	-	2,6,7,9,12,13
CO-2	**Analyze** real environment **and Formulate** the problem statement.	-	-	9,12,13
CO-3	**Conduct** exhaustive literature survey	-	-	9,12,13
CO-4	**Propose** sustainable engineering solutions	-	-	7,5,12,13
CO-5	**Prepare** the report and communicate effectively through presentation.	-	-	9,10,12,13
CO-6	**Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain.	-	-	11

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	1.0	-	-	1.0	1.0	1.0	-	1.0	-	1.0	1.0	1.0	-	-	-

Guidelines for conduction:

1. Team consists of 4-5 students.
2. Students have to choose a guide among the faculty members who are teaching their semester.
3. In consultation with guide, the team should carry out their project work.
4. Final evaluation is based on seminar and report submission.
5. This requires designated committee to monitor the process of conduction

V Semester

Course Code	Course Category	Course Title	Teaching		Examination				
			L-T-P (Hrs/Week)	Credits	CIE	Theory (SEE)		Practical (SEE)	
					Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UHUC500	HU	Management, Entrepreneurship and IPR	4-0-0	4	50	100	3	-	-
18UCSC500	PC	Data Communication	4-0-0	4	50	100	3	-	-
18UCSC501	PC	Database Management Systems	4-0-0	4	50	100	3	-	-
18UCSC502	PC	Compiler Design and System Software	3-0-0	3	50	100	3	-	-
18UCSC503	PC	Software Engineering	3-0-0	3	50	100	3	--	--
18UCSL504	PC	Database Management Systems Lab	0-0-3	1.5	50	--	--	50	3
18UCSL505	PC	Compiler Design and System Software Lab	0- 0 -3	1.5	50	--	--	50	3
18UCSL506	PC	Minor Project-1	0-0 – 2	1	50	--	--	--	--
18UHUL507	HU	Soft skills/Aptitude	0-0 – 2	1	50	--	--	--	--
18UCSE508	PE	Advanced Object Oriented Programming	3-0-0	3	50	100	3	-	-
18UCSE509	PE	System Simulation and Modeling	3-0-0	3	50	100	3	-	-
18UCSE510	PE	Advanced Graph Theory	3-0-0	3	50	100	3	-	-
Total			21-0-10	26	500	600		100	

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination

L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks
The students will choose **one elective** from the elective list (designated as **PE** in the above table).

18UHUC500	Management, Enterpreuership and IPR	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- The evolution of IT managementand related aspects.
- The scope of entrepreneurship in digital firms.
- The issues and procedures related to intellectual property rights.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Study** the principles of management in a given organization. [Familiarity] [BL-2]	-	2	-
CO-2	**Describe** and **analyze** the role of staffing and the need for motivation in management.[Usage] [BL-3]	-	2	-
CO-3	**Explain** the role of entrepreneur in establishing an organization. **[Familiarity] [BL-2]**	-	2, 6	-
CO-4	**Describe** the importance and provisions of institutional support in establishing an enterprise. [Familiarity] [BL-2]	-	2, 6	-
CO-5	**Explain** the core principles, procedures and related laws and **apply** IPR for given new idea/invention. **[Usage] [BL-3]**	-	5, 8, 10	-

PO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	-	-	2.0	2.0	-	2.0	-	2.0	-	-	-	-	-	-

Prerequisites: Knowledge of humanities courses.

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Engineering and Management:** Historical Development of Engineering and Management, Management as synthesis. Planning, Forecasting and Decision Making: Nature of Planning, foundation of planning, some planning concepts, forecasting, nature of decision making, management science, tools for decision-making.	10	-	1	1	3. **Class Test/ Quiz** (CO-1) 4. **Written test**- CIE / SEE (CO-1)
2	**Organizing and staffing:** Nature of organizing, traditional organizational theory, technology and modern organization structures, staffing technical organization, authority and power; delegation, meeting and committees. Motivation: Motivation, leadership, motivating and leading technical professionals. Controlling: Process of control, financial and non-financial controls.	11	-	2	1	5. **Class test / Quiz** (CO-2) 6. **Written test**- CIE/SEE (CO-2) 7. **CTA - Case Study** on analysis of the role of staffing and the need for motivation in management.
3	**Foundations of Entrepreneurship:** Meaning of entrepreneur, functions of entrepreneur, types of entrepreneur, concept of entrepreneurship, role of entrepreneurs in economic development& barriers of entrepreneurship. Small Scale Industry: Definition, characteristics of SSI, role of SSI in economic development, advantages of SSI, steps to start an SSI,	11	-	3	2	3. **Written test** – CIE / SEE (CO-3)

	impact of liberalization, privatization, and globalization on SSI, definition of ancillary and tiny industry					
4	**Government and Institutional Support:** Nature of support from government, objectives and functions of SSI, SIDBI, DIC, single window agency, KIADB, KSSIDC, KSFC. Preparation of Project: Meaning of project identification, project report, contents and formulation, identification of business opportunities, feasibility studies, types and purpose.	**10**	**-**	**4**	**2, 3**	**1. Written test-** CIE / SEE **(CO-4)**
5	**Intellectual Property Rights:** Meaning and forms of intellectual property rights, competing rationale for protection, international conventions and security. Copyright: Meaning of copyright, content of copy right, ownership and rights, period of copyright, assignment and relinquishment of copyright, license, infringement of copy right, fair use, offenses and penalties. Patents: Concept of patent, patentable inventions, procedure for obtaining patent, rights and obligations of patent holders, infringements and remedies, offenses and penalties. Industrial Designs: Definition of design, procedure for registration, rights conferred by registration, infringements, Trademark and related issues.	**10**	**-**	**5**	**3**	**1. Written test-** CIE / SEE **(CO-5)** **2. CTA – Assignment** on Comparative Study of IPR and hands on **IPR filing procedure.**

Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	4. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 5. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Reference Books:

1. Kenneth C. Laudon and Jane P. Laudon: Management Information Systems - Managing the Digital Firm, 8th edition, Pearson Publication, 2017.
2. Making Intellectual Property Work for Business - Handbook for Chambers of Commerce and Business Associations Setting Up Intellectual Property Services by ICC and WIPO, Paperback, 2012.

18UCSC500	Data Communication	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Evolution of network and internet.
- Protocols, applications pertaining to network and internet communication.
- Layered architecture and services.
- Network performance measurement and emerging technologies.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the evolution and core operating principles of computer network in terms of architecture, functions, switching techniques and protocols. [Familiarity] [BL-2]	-	13	1
CO-2	**Explain** the fundamental concepts of analog and digital communication techniques. [Familiarity] [BL-2]	-	1	13
CO-3	**Explain** the procedure and **analyze** the communication channels for errors. [Usage] [BL-4]	-	1,2,13	-
CO-4	**Explain** the core working principles of switching techniques and their applications. [Familiarity] [BL-2]	-	1	2,13
CO-5	**Explain** the working principles and **write** simple **programs** to demonstrate peer to peer and logical link control protocols in building networked space. [Usage] [BL-3]	-	1,2,3	13
CO-6	**Explain** the core working principles and **write** simple **programs** using simulators to demonstrate the MAC protocols and their performance for effective channel utilization. [Usage] [BL-3]	-	1,2,3	13

PO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	1.8	2.0	-	-	-	-	-	-	-	-	-	1.3	-	-	-

Prerequisites: NIL

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Communication Networks, Services, applications and layered architectures:** Evolution of Network Architecture and Services: Telegraph Networks and Message Switching, Telephone Networks and Circuit Switching, and the Internet, Computer Networks and Packet Switching, examples of Protocols, Services, and Layering: HTTP, DNS, and SMTP, TCP and UDP Transport Layer Services; The OSI Reference Model: The seven layer OSI Reference Model, Unified View of Layers, Protocols, and Services **Overview of TCP/IP Architecture:** TCP/IP Architecture, TCP/IP Protocol: How the layer work together, Protocol Overview; Application Layer Protocols and TCP/IP Utilities	10	-	1	1	1. Class Test (CO-1) 2. Written test - CIE / SEE. (CO-1)
2.	**Digital Transmission Fundamentals:** Digital Representation of Information: Block-Oriented Information, Stream Information; Why Digital Communications ?: Comparison of Analog and Digital Transmission, Basic properties of Digital Transmission Systems; Digital Representation of Analog Signals: Bandwidth of Analog Signals, Sampling of an Analog Signal, Digital Transmission of Analog Signals; Characterization of Communication Channels: Frequency Domain Characterization, Time Domain Characterization;	12	3	2	1,2	1. Class test / Quiz (CO-2) 2. Written test- CIE/SEE. (CO-2)

	Fundamental Limits in Digital Transmission: The Nyquist Signaling Rate, The Shannon Channel Capacity; Line Coding ;Modems and Digital Modulation: Binary Phase Modulation, QAM and Signal Constellations, Twisted Pair, Coaxial Cable, Optical Fiber, Radio Transmission, Infrared Light; Error Detection and Correction: Error Detection, Two Dimensional Parity Checks, Internet Checksum, Polynomial Codes, Standardized Polynomial Codes, Error Detecting Capability of a Polynomial Code.					
3.	**Circuit Switching Networks:** Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Wavelength-Division Multiplexing; SONET: SONET Multiplexing, SONET Frame Structure; Transport Networks: SONET Networks, Optical Transport networks; Circuit Switches: Space Division Switches, Time Division Switches; The Telephone Network: Transmission Facilities, End to End Digital Services.	**10**	**-**	**3**	**2**	1. **Class test / Quiz (CO-3)** 2. **Written test-** CIE/SEE.**(CO-3)**
4.	**Peer-to-Peer Protocols and Data Link Layer:** Peer-to-Peer Protocols: Peer –to-Peer Protocols and Service Models; ARQ Protocols and Reliable Data Transfer Service: Stop-and-Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ; Other Peer-to-Peer Protocols; Data Link Controls: Framing; Point to Point Protocol; HDLC Data link Control; Link Sharing using Packet Multiplexers: Statistical Multiplexing, Speech Interpolation and the Multiplexing of Packetized Speech.	**10**	**3**	**4**	**2,3**	1. **Class test / Quiz (CO-4)** 2. **Written test-** CIE/SEE.**(CO-4)** 3. **CTA - Write** simple **programs** to demonstrate peer to peer and logical link control protocols in building networked space.

5.	**Medium Access Control Protocols and Local Area Networks:** The Medium Access Control Protocols: Multiple Access Communications; Random Access: ALOHA, Slotted ALOHA, CSMA, CSMA-CD; Scheduling Approaches to Medium Access Control: Reservation Systems, Polling, Token-Passing Rings; Channelization: FDMA, TDMA, CDMA.	**10**	**-**	**5**	**3**	1. **Written test**- CIE/SEE. **(CO-5)** 2. **CTA - Write** simple **programs** using simulators to demonstrate the MAC protocols and their performance for effective channel utilization.
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Alberto Leon Garcia and IndraWidjaja, "Communication Networks", "Fundamental Concepts and Key architectures", 2nd Edition, Tata McGraw-Hill, 2001
2) Behrouz A. Forouzan, "Data Communications and Networking", 5th Edition, Tata McGraw-Hill, 2007
3) William Stallings, "Data and Computer Communication", 9th Edition, Pearson Publication, 2013.

18UCSC501	Database Management Systems	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Data models and relational theories
- Database design, programming using SQL/PL-SQL, database architecture and transaction concepts.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Compare** the traditional file system and Data Base approach. [Usage][BL-3]	-	2, 13	-
CO-2	**Identify** entities, attributes, their relationships and **prepare** ER model for the given application scenario. [Usage][BL-4]	2, 3, 13	-	15
CO-3	**Write** the queries using relational algebra for the given data manipulation requirement of an RDBMS. [Usage][BL-3]	2,3,13,14	-	15
CO-4	**Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario. [Usage][BL-3]	2,3,13,14	-	15
CO-5	**Write** triggers, stored procedures and functions for the given application scenario. [Usage][BL-3]	2,3,13,14	-	15
CO-6	**Design** database in appropriate normal form for a given application scenario. [Usage][BL-3]	2,3,13	-	15
CO-7	**Explain** the strategies to deal with the issues related to transaction management and to ensure ACID properties. [Familiarity][BL-2]	-	1,13	-

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.8	3.0	-	-	-	-	-	-	-	-	-	2.7	3.0	1.0	-

Prerequisites: Knowledge of:

- Programming languages,
- Set Theory,
- File systems, and
- Abstract application development process.

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools / Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction:** Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; **Entity-Relationship Model:** Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.	10	-	1 2	1	1. **Written test** - CIE/SEE. **(CO- 1,2)**
2.	**Relational Model and Relational Algebra:** Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational	10		3	1, 2	1. **Written test**- CIE/SEE. **(CO-3)**

	Operations: JOIN and DIVISION; Examples of Queries in Relational Algebra; Relational Database Design ER- to-Relational Mapping.					
3.	**SQL**: Data Definition and Data Types; DDL statements like creation and specification od table, DCL statements for Schema change, alter, delete etc; DML statements like Insert, Delete and Update statements in SQL etc and more complex statements for Basic queries Nested sub queries, Correlated sub queries. PL/SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL.	**12**	**-**	**4**	**2**	1. Class Test/Quiz **(CO-4,5)** 2. **Written test** – CIE / SEE.**(CO-4,5)** 3. **CTA - Assignment:** writing SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario. 4. **CTA - Assignment:** writing triggers, stored procedures and functions for the given application scenario. Note: The above two activities may be linked with the associated laboratory.
4.	**Database Design – 1:** Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference rule, Equivalence of sets, Minimal set cover. Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form.	**10**	**-**	**5**	**3**	1.**Written test**- CIE / SEE. **(CO-6)** 2.**CTA - Assignment:** Desgning the database in approritae normal form for a given application **scenario. (CO-6)**

5.	**Database Design –2:** Properties of Relational Decompositions; Dependency preservation, Lossless (non additive) join properties, Problem with null values and dangling tuples. **Transaction Management:** The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Introduction to ARIES; The log; Other recovery-related structures; The write-ahead log protocol; Check pointing	10		5 6	3	1 **Written test-** CIE / SEE.**-(CO-6,7)** 2 **CTA - Assignment:** writing programs to demonstrate the strategies to deal with the issues related to transaction management

Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. Based on the time availability, course teachers may try with **programming assignments** on NoSQL, Graph Databases, file system implementation to simulate SQL and other contemporary DB technologies.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Elmasri & Navathe, "Fundamentals of Database Systems", 6th Edition, Addison-Wesley, 2012.
2. Raghu Ramakrishnan & Johannes Gehrke, "Database Management Systems", 3rd Edition, McGraw-Hill, 2003.
3. Silberschatz, Korth and Sudharshan, "Data base System Concepts", 6th Edition, Mc-Graw Hill, 2010.
4. C.J. Date, A. Kannan & S. Swamynatham, "An Introduction to Database Systems", 8th Edition, Pearson Education, 2006.

18UCSC502 Compiler Design and System Software (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This is a 3 credit course at undergraduate level enabling the students to understand structure of a compiler,representation of patterns and syntax using lexical rules and grammars respectively, working of parsers, translation schemes, code optimization and code generation, working of assemblers, loaders, linkers and macro processor.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs / PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the basic structure and working principles of phases of compiler. [Familiarity] [BL-2]	-	13	1
CO-2	**Write** a parser for the given input based on the appropriate parsing technique and **validate** the design.[Assessment] [BL-4]	13,14	1,2,3	15
CO-3	**Generate** an optimized intermediate code. [Usage] [BL-3]	-	1,2,3, 13	15
CO-4	**Explain** the working principles of run time environments that include stack allocation, heap management and garbage collection technique used in compiler. [Familiarity] [BL-2]	-	1,2,3, 13	15
CO-5	**Generate** optimized code for the given intermediate code.[Usage] [BL-3]	-	1,2,3, 13	15
CO-6	**Design** Assembler for the given language specification and **validate** the design. [Usage] [BL-3]	-	1,2,3, 13,14	15
CO-7	**Design** Macroprocessor for the given language specification and **validate** the design. [Usage] [BL-3]	-	1,2,3, 13,14	15
CO-8	**Explain** the working principles of Linkers & Loaders for the given language specification. [Familiarity] [BL-2]	-	1,2,3, 13,14	15

Note: BL- Bloom's Level

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.9	2.0	2.0	-	-	-	-	-	-	-	-	-	2.1	2.3	1.0	-

Prerequisites: Knowledge of Finite automata and formal language and any programming language.

Course Contents

Modules	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Introduction:** Different Phases of Compilers, Comparison of Compilers and Interpreters. Top-down Parsing: RDP and Predictive parsing.	7	-	1, 2	1	1. **Class Test** (Prerequisite) 2. **Quiz** (CO-2) 3. **Written test**- CIE/SEE. (CO-1,2)
2	**Bottom-up Parsing:** Simple LR, LALR, CLR, parsers ambiguous grammars.	6	2	2	1	1. **Quiz - (CO-2)** 2. **Written test**- CIE/SEE.(CO-2) 3. **CTA** – Assignment on writing a parser for the given input
3	**Intermediate Code Generation and Optimizations:** Syntax-directed translation; Syntax-directed translation schemes, Variants of syntax trees; Three-address code; Types and declarations; Translation of expressions; Type checking; Control flow; Various techniques of machine independent optimization.	5	3	3	2	1. **Class test** (CO-3) 2. **Quiz** (CO-3) 3. **Written test**- CIE/SEE.(CO-3) 4. **CTA** – Assignment on generating an optimized intermediate code. (CO-3)
4	**Run-Time Environments:** Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management; Introduction to garbage collection. **Code Generation:** Issues in the design of Code Generator; The Target language; Addresses in the target	8	-	4 5	2 3	1. **Written test**– CIE/SEE. (CO-4) 2. **CTA** – Assignment on Study of memory layout of C/ JAVA language. (CO-4)

	code; Basic blocks and Flow graphs; Optimization of basic blocks; A Simple Code Generator					
M5.	**Ancillary Code Processing Techniques:** Generic description of Assembler, Loader, Linker and Macro's. Assemblers: Basic Assembler Features & Functions and Design of assembler. Loaders and Linkers: Basic Loader Functions - Design of Loaders and Linkers Macro Processor: Design of Macro Processors.	**6**	**2**	**6,7,8**	**3**	1. **Written test** - CIE/SEE. **(CO-6, CO-7, CO-8)** 2. **Quiz** **(CO-6, CO-7, CO-8)**

Other performance ensuring measures	
Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. **Course Project (Group activity) CTA:** Project to implement a translator which translates the input from one form to other. Some sample definitions are listed below: • Conversion of an algorithm to C program • Translate some simple words in English to any other regional language. • Translation of C language to assembly language

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman,"Compilers- Principles, Techniques and Tools", 2/E, Addison-Wesley, 2007.
2) D.M.Dhamdhere, "System Programming and Operating Systems", 2nd revised edition, Tata McGraw - Hill, 2009 reprint.
3) Leland L Beck, "System Software : An Introduction to Systems Programming" 3rd Edition Pearson Education 2007
4) John J Donovan, "System Programming", Tata McGraw-Hill 2017

18UCSC503	Software Engineering	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs):

This is a 3 credit, 39 contact hours course at undergraduate level focusing on knowing the process of software system development and enables students to develop software system using engineering techniques.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Describe** the properties of various critical systems and the quality parameters. [Familiarity] [BL-2]	-	-	16
CO-2	**Describe** the different key practices of process models. [Familiarity] [BL-2]	-	3,13	
CO-3	**Identify** various system requirements and **prepare** system specification reports to solve real life problems in various domains and **develop** domain expertise. [Usage] [BL-3]	1,2,13	-	-
CO-4	**Conceptualize** the system through **design and modeling** the system architecture, components and processes with quality and standards. [Usage] [BL-3]	1,2,3,13	5	10,12
CO-5	**Develop** software system using engineering techniques, industry relevant tools and programming features/techniques. [Usage] [BL-3]	1,2,3,13,14	5,15	10,11, 12,16
CO-6	**Verify and validate** the given system using standard tools and techniques. [Usage][BL-3]	-	5,15	10
CO-7	**Manage** project in terms of risk, configuration/versions, Cost and Resources. [Usage] [BL-3]	-	9,11	10

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	3.0	1.5	-	2.0	-	-	-	2.0	1.0	1.5	1.0	2.8	3.0	2.0	1.0

Prerequisites: Knowledge of:

a. Basics of computer systems and its usage.
b. Any Computer Programming Language.

Course Contents

Modules	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Overview:** Introduction: FAQ's about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems. **Critical Systems, Software Processes:** Critical Systems: A simple safety critical system; System dependability; Availability and reliability. Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering.	7	-	1 2	1	1. **Class Test.** (CO-1, 2) 2. **Written test**- CIE / SEE. (CO-1,2)
2	**Requirement Engineering:** Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document. Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.	5	2	3	1	1. **Class test.** (CO-3) 2. **Written test**- CIE/SEE. (CO-3) 3. **CTA- Group Activity: Course project** Preparing SRS report in IEEE standard format. (CO-3)
3	**System models:** System Models: Context models; Behavioral models; Data models; Object models;	6	3	4	2	1. **Class test.** (CO-4) 2. **Written test**- CIE/SEE. (CO-4)

	Structured methods. **Software Design and Development:** Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles. Object-Oriented design. UI Design Issues. **Rapid Software Development:** Agile methods; Extreme programming; Rapid application development. Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.					3. **CTA- Group Activity: Course project** Use of industry relevant tool for building system models. Evaluation is to be based on report submission and oral presentation. **(CO-4)**
4	**Verification and Validation:** Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. Software testing: System testing; Component testing; Test case design; Test automation. Testing Techniques: Equivalence Partitioning, Boundary Value Analysis,, Cause Effect Graphing, Test Generation from Predicates, Statement testing, Branch Testing, Condition Testing, Path Testing, Procedural Call Testing, Data Flow Testing.	**6**	**3**	**6**	**2, 3**	1. **Class test. (CO-6)** 2. **Written test-** CIE/SEE. **(CO-6)** 3. **CTA- Group Activity:Course project** Use of industry relevant tool for testing the design and the code developed. **(CO-6)** 4. **CTA- Group Activity**: Exercise on writing test script for the given system specification (Model based testing) **(CO-6)**
5	**Software Quality & Project Management:** Various Software quality parameters and associated standards and procedures, Project Management activities; Project planning; Project scheduling; Risk management. Configuration Management, Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model. Software Cost Estimation: Productivity; Estimation techniques, Project duration and staffing.	**7**	**-**	**7**	**3**	1. **Written test-** CIE/SEE.**(CO-7)** 2. **CTA- Group Activity**: Exercise on cost estimation and project management tool. Ex: GitHub **(CO-7)**

Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA. 3. **CTA- Group Activity:** Programming based on requirements & design developed and use of tools to study various testing strategies. (CO- 5) Evaluation is to be based on report submission and oral presentation

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Ian Somerville, "Software Engineering", 10/E, Person Education, 2016.
2. Roger Pressman, "Software Engineering, Practitioners approach", 7/E, McGraw-Hill, 2010.

18UCSL504	Database Management Systems Lab	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on hands on experience on creation of data models, database design, programming using SQL/PL-SQL and development of an application using any high level language.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Identify** entities, attributes, their relationships and **prepare** ER model for the given problem. [Usage][BL-3]	2, 3, 13	-	15
CO-2	**Design** database in appropriate normal form for the given problem. [Usage][BL-3]	2,3,13	-	15
CO-3	**Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario. [Usage][BL-3]	2,3,13,14	-	15
CO-4	**Write** the programs using advanced features of data base programming that includes PL/SQL, Cursors, Triggers, Stored procedures and Functions for given application scenario. [Usage][BL-3]	2,3,13,14	-	15

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	3.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Registration for / Completion of: DBMS theory course

Contents:

Part A: Multiple standalone preparatory exercises to meet the course outcomes.

Part B: Project work (Standalone application or web enabled application) based on knowledge gained from theory course and part-A hands on experience.

18UCSL505	Compiler Design and System Software Lab	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on Representation of patterns and syntax using lexical rules and grammars respectively, Implementation of parser & translation schemes, Implementation of assemblers, loaders, linkers & macro processor, Knowledge of system level APIs for implementation of IPC and system commands.

Course outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Prepare** the grammar for the given constructs and **Write** a program using compiler writing tools to implement lexical analyzer and parser.	13,14	1,2,3	15
CO-2	**Write** a program to **implement** a parser.	13,14	1,2,3	15
CO-3	**Write** a program to implement assembler functions.	13,14	1,2,3	15
CO-4	**Write** a program to **Implement** various UNIX commands using system calls.	13,14	1,2,3	15
CO-5	**Use** IPC concepts in implementing communication protocol.	13,14	1,2,3	15

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Knowledge of: Unix Operating System, Any programming language, and Finite automata and formal Languages

Contents:

1. Programs on Lex and Yacc.
2. Implementation of parser.
3. Implementation of assembler.
4. Emulation of basic commands of UNIX using system calls.
5. Application development using Inter Process Communication.

18UCSL506	Minor Project – 1	(0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs): This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

Course Outcomes (COs):

CO	Description of the course outcome At the end of the course student should be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Identify** the societal problems. [Usage][BL-3]	-	2,6,7,9,12,13	-
CO-2	**Analyze** real environment and **Formulate** the problem statement. [Usage][BL-3]	-	2,9,12,13	-
CO-3	**Conduct** exhaustive literature survey [Usage][BL-3]	-	2, 9,12,13	-
CO-4	**Propose** sustainable engineering solutions / prototypes. [Usage][BL-3]	-	3,5,7,12,13	-
CO-5	**Prepare** the report and communicate effectively through presentation. [Usage][BL-3]	-	8,9,10,12	-
CO-6	**Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain. [Usage][BL-3]	-	11	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	2.0	-	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	-	-	-

Guidelines for conduction:

1. Team consists of 4-5 students.
2. Students have to choose a guide among the faculty members who are teaching their semester.
3. In consultation with guide, the team should carry out their project work.
4. Final evaluation is based on following evidence will be looked into and accordingly rubrics will be developed, such as problem statement, design, prototype/part of implementation, use of standard tools and techniques for testing and validation procedure.
5. This requires designated committee to monitor the process of conduction

18UHUL507	Soft Skills / Aptitude	(0-0-2) 1

Contact Hours: 24

This course will be dealt at the Institute level.

This is included with an objective of improving the communication skills, proficiency in English language and aptitude ability of the student. This is a credit course and aimed to enhance the employability. Both the internal and external resource persons shall be engaged in imparting the related knowledge and shall have only CIE as the evaluation component. There shall be one test conducted at the end for 25 marks in Aptitude testing and there shall be one presentation by the student for 25 marks or any other suitable testing components. The arrangement for CIE evaluation is to be done by the department and maintain the relevant documents.

18UCSE508	Advanced Object Oriented Programming	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs):

This course focuses on core and advanced Java language features that are part of JDK 8 and above.

Course Outcomes (COs):

Course Outcomes	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Build** graphical user interface using JavaFX for a given problem [Usage][BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-2	**Develop** applications that involve parallel programming abilities using concurrent utility feature [Usage][BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-3	**Write** programs to solve a given problem using generics and collection Frameworks [Usage][BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-4	**Use** Java networking features to **write** applications that involve client / server interactions [Usage][BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-5	**Develop** an application that use appropriate driver classes to connect databases and **perform** database operations required as per problem specification [Usage] [BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-6	**Develop** web-based applications using J2EE features like Servlets and JSP [Usage] [BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-7	**Write** program using lambda expressions to solve given problem scenario. [Usage] [BL-3]	13,14,15	-	1, 2,3, 5, 9,16
CO-8	**Write** program using stream APIs to solve given problem scenario. [Usage] [BL-3]	13,14,15	-	1, 2,3, 5, 9,16

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	1.0	1.0	-	1.0	-	-	-	1.0	-	-	-	3.0	3.0	3.0	1.0

Prerequisites: Knowledge of:

- Basic features of Java
- Object oriented programming paradigm, its concepts and practices

Course Contents

MODULE	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**GUI programming with JavaFX:** Introducing JavaFX GUI programming – JavaFX basic concepts, JavaFX application skeleton, the Application thread; Exploring JavaFX controls – ToggleButton, RadioButton, CheckBox, ListView, TextField, TreeView, disabling a control; Introducing JavaFX menus – menu basics, overview of MenuBar, Menu and MenuItem, create main menu, add mnemonics and accelerators to menu items, use RadioMenuItem and CheckMenuItem, create ContextMenu and Toolbar	5	2	1	1	1. **Written test-** CIE / SEE **(CO-1)** 2. **CTA – Quiz (CO-1)** 3. **CTA – Group Activity:** Develop a standalone GUI application to solve the given problem **(CO-1)**

2	**Concurrent Utilities:** concurrent API packages, using synchronization objects, phaser, using an executer, TimeUnit enumeration, concurrent collections, locks, atomic operations, parallel programming via Fork/Join framework **Generics and Collections Framework:** Generics - what are generics, a simple generics example, a generic class with two type parameters, general form of general class, bounded types, using wildcard arguments, creating a generic method; Collections Framework – collections overview, the collection interfaces, the collection classes, accessing a collection via an iterator, for-each alternative to iterators, spliterators, storing user-defined classes in collections	4 5	- -	2 3	1 1,2	1. **Written test**- CIE/SEE **(CO- 2 & 3)** 2. **CTA –Quiz (CO- 2 & 3)** 3. **CTA – Assignment** on writing programs using generics and Collections **(CO-2)**
3	**Networking and RMI:** networking basics, the networking classes and interfaces, InetAddress class, TCP/IP client sockets, HttpURLConnection class, TCP/IP server sockets, Datagrams; Remote Method Invocation (RMI) – A simple client/server application using RMI **J2EE Databases:** the concept of JDBC, JDBC driver types, a brief overview of JDBC process, database connection, Statement objects, ResultSet class, ResultSetMetaData class	4 3	- -	4 5	2 2	1. **Written test**- CIE/SEE **(CO-4&5)** 2. **CTA – Quiz (CO-4&5)** 3. **CTA – Group Activity:** Develop a standalone application to solve the given problem using Database and Networking Features. **(CO-5)**
4	**Java Servlets:** Introduction,benefits of Java servlets, a simple java servlet, anatomy of a Java servlet, deployment descriptor, reading data from a client, reading/writing HTTP request/response headers; working with cookies, tracking sessions	3	2	6	2,3	1. **Written test**- CIE/SEE **(CO-6)** 2. **CTA – Quiz (CO-6)** 3. **CTA – Group Activity:** Develop a web application using HTML, JSP/ Servlets, Sessions and Cookies to solve a given problem.

	Java ServerPages: Introduction,JSP tags, Request string, User sessions, cookies, session objects	3	1	6	3	(CO-6)
5	**Extended features of Java:** Lambda Expressions – introducing lambda expressions, block lambda expressions, passing lambda expressions as arguments, lambda expressions and exceptions, lambda expressions and variable capture, method references; The Stream API – stream basics, reduction operations, using parallel streams, mapping, collecting, iterators and streams	7	-	7	3	1. **CTA – Quiz**(CO-7) 2. **Written test-** CIE/SEE(CO-7) 3. **CTA – Programming assignment** on Extended features of Java (CO-7)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Java: The Complete Reference, 10th edition, Herbert Schildt, McGraw-Hill, 2017
2) J2EE: The Complete Reference, Jim Keogh, McGraw-Hill, 2011

18UCSE509	System Simulation and Modeling	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To introduce students to basic simulation methods and tools for modelling and simulation of continuous, discrete and combined systems.
- The ability to analyze a system and to make use of the information to simulate various systems to improve the performance.
- Analytical methods (Markov Models and Queuing Networks) and simulation techniques (Monte Carlo Techniques and Event Driven Simulation) applied in performance modelling of communication systems and networks.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Conceptualize** the discrete system simulation with the aid of real time examples. [Usage] [BL-3]	-	5,13	1,2,3,15
CO-2	**Write** a program to simulate a given scenario. [Usage] [BL-3]	-	5,13,14	1,2,3,15
CO-3	**Apply** different statistical models available in simulation and their usage in specific applications. [Usage] [BL-3]	-	5,13,14	1,2,3,15
CO-4	**Design** the queuing systems and **evaluate** the performance. [Usage] [BL-3]	-	5,13	1,2,3,15
CO-5	**Identify** the distribution of data to adhere to fitness test and also analyze the corresponding simulation. [Usage] [BL-3]	-	5,13	1,2,3,15

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	1.0	1.0	-	2.0	-	-	-	-	-	-	-	2.0	2.0	1.0	-

Prerequisites: Knowledge of Probability and Statistics

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction:** When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study. Simulation examples: Simulation of queuing systems; Simulation of inventory systems; other examples of simulation. General Principles	7	-	1	1	1. **Class Test** (CO-1) 2. **Written test**- CIE / SEE. (CO-1) 3. **CTA - Case study** on examples of simulation (CO-1)
2.	**Concepts in Discrete-Event Simulation:** The Event-Scheduling, Time-Advance Algorithm, World Views, Manual simulation Using Event scheduling; List processing.	7	-	2	1, 2	1. **Class test / Quiz** (CO-2) 2. **Written test**- CIE/SEE.(CO-2) 3. **CTA-Writing programs** to demonstrate event scheduling (CO-2)
3.	**Statistical Models In Simulation:** Review of technology and concepts; Useful statistical models; discrete distributions; Continuous distributions; Poisson process; Empirical distributions.	8	-	3	2	1. **Class test / Quiz** (CO-3) 2. **Written test**– CIE/SEE. (CO-3) 3. **CTA**- Application oriented problems using distribution

						techniques (CO-3)
4.	**Queuing Models:** Characteristics of queuing systems; Queuing notation; Long-run measures of performance of queuing systems; Steady state behavior of M/G/1 queue; Networks of queues.	7	-	4	2,3	1. Class Test/Quiz (CO-4) 2. **Written test**- CIE/SEE. (CO-4) 3. CTA - Application oriented problems using Queuing models.(CO-4)
5.	**Input Modeling:** Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; selecting input models without data; Multivariate and Time-Series input models. **Estimation of absolute performance:** Types of simulations with respect to output analysis; stochastic nature of output data; Measures of performance and their estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations.	10	-	5	3	1. **Written test**- CIE/SEE.-(CO-5) 2. CTA-Case study(CO-5)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Banks, John S. Carson II, Barry L. Nelson- DavidM. Nicol, "Discrete-Event System Simulation", Jerry 5/E, Pearson Education, 2013.
2) J. A. Sokolowski, C.M. Banks, "Principles of Modeling and Simulation: A multidisciplinary Approach", John Wiley & Sons Publications, edited 2011.
3) Sheldon M. Ross, "Simulation", 4/E, Elsevier, 2006.

18UCSE510	Advanced Graph Theory	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Isomorphism, Connected and Disconnected Graphs.
- Spanning trees and Cutsets.
- Planarity of Graphs.
- Chromatic Number and Polynomial.
- Directed Graphs.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Determine** whether the two graphs are isomorphic or not. [Usage] [BL-3]	-	1,2	-
CO-2	**Prove** the properties of trees, cutsets and **determine** the spanning trees of a graph. [Usage] [BL-3]	-	2,13	1
CO-3	**Determine** the planarity and dual of a graph [Usage] [BL-3]	-	2,13	1
CO-4	**Determine** the chromatic polynomial and chromatic number of a graph. [Usage] [BL-3]	-	2,13	1
CO-5	**Explain** the principles of directed graphs and **represent** the digraphs in different forms. [Usage] [BL-3]	-	2,13	1

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	2.0	-	-	-	-	-	-	-	-	-	-	2.0	-	-	-

Prerequisites: Knowledge of Discrete Mathematical Structures.

Course Contents

Modules	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction to Graphs:** Definition, Finite and Infinite Graphs, Incidence and Degree, Isolated vertex, Pendant Vertex, and Null graph. **Paths and Circuits:** Isomorphism, Subgraphs, Walks, Paths, and Circuits, Connected and Disconnected Graphs, Operations on Graphs, Euler graphs, Hamiltonian Paths and Circuits	8	-	1	1	1. **Written test-** CIE / SEE. (CO-1) 2. **CTA - Assignments** on performing operations on graphs and determining Hamiltonian paths and circuits. (CO-1)
2.	**Trees and Fundamental Circuits:** Trees, Fundamental properties of trees, Pendant vertices in a tree, Distance and Centres in a tree, Rooted and Binary trees, Spanning Trees, Finding all spanning trees in a graph, Spanning trees in a weighted graph, The matrix – tree theorem. **Cutsets:** Introduction, Properties, All cutsets in a graph, Fundamental circuits and cutsets, The Chinese Postman problem.	8	-	2	1 2	1. **Written test-** CIE/SEE.(CO-2) 2. **CTA - Implementation** of an algorithm to determine the spanning trees of a graph. (CO-2)
3.	**Planar and Dual Graphs:** Introduction to Planar graphs, Kuratowski's two graphs, Different representations of a Planar graph, Detection of Planarity, Geometric Dual.	8	-	3	2	1. **Quiz** (CO-3) 2. **Written test**– CIE/SEE (CO -3)
4.	**Coloring:** Chromatic number, Chromatic Partitioning,	7	-	4	3	1. **Quiz** (CO-4)

	Chromatic Polynomial, Matchings, Coverings, The Four Color Problem, Brooks Theorem.					**2. Written test** - CIE/SEE. **(CO-4)**
5.	**Directed Graphs:** Definition, Types of Digraphs, Binary Relations, Directed Paths and Connectedness, Euler Digraphs, Trees with Directed Edges, Fundamental Circuits in Digraphs, Matrices A,B, and C of Digraphs, Adjacency matrix of Digraph, Random graphs.	**8**	-	**5**	**3**	**1. Written test**- CIE/SEE. **(CO-5)** **2. CTA - Implementation** of algorithms to determine the different representations of directed graphs.
	Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…					**1.** Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. **2. Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Narasingh Deo"Graph Theory with Applications to Engineering and Computer Science", Dover Publications Inc 2016.
2. Ralph P. Grimaldi "Discrete and Combinatorial Mathematics", 5th Edition, Pearson Education. 2006.
3. Kenneth H. Rosen "Discrete Mathematics and its Applications", 7th Edition, McGraw Hill, 2012.

VI Semester

Course Code	Course Category	Course Title	Teaching		Examination				
			L-T-P (Hrs/Week)	Credits	CIE	Theory (SEE)		Practical (SEE)	
					Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UCSC600	PC	Computer Networks	**4-0-0**	4	50	100	3	-	-
18UCSC601	PC	Object Oriented Modeling and Design	**4-0-0**	4	50	100	3	-	-
18UCSL602	PC	Computer Networks Lab	**0-0-3**	1.5	50	-	-	50	3
18UCSL603	PC	Industry Oriented Programming Practices Lab	**0-0-3**	1.5	50	-	-	50	3
18UCSL604	PC	Minor Project-2	**0-0-4**	2	50	-	-	50	3
18UHUL605	HU	Soft skills/Aptitude	**0-0-2**	1	50	-	-	-	-
18UCSE606	PE	Unix Systems Programming	**3-0-0**	3	50	100	3	-	-
18UCSE607	PE	Digital Image Processing	**3-0-0**	3	50	100	3	-	-
18UCSE608	PE	Principles of Programming	**3-0-0**	3	50	100	3	-	-
18UCSE609	PE	Data Mining	**3-0-0**	3	50	100	3	-	-
18UCSE610	PE	Advanced Data Structures and Algorithms	**3-0-0**	3	50	100	3	-	-
18UCSE611	PE	Pattern Recognition	**3-0-0**	3	50	100	3	-	-
18UCSE612	PE	Embedded Systems	**3-0-0**	3	50	100	3	-	-
18UXXO6XX	OE	Open Elective - 1	**3-0-0**	3	50	100	3	-	-
Total			**17 - 0 -12**	**23**	**450**	**500**		**150**	

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination
L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks

The students will choose **two electives** from the elective list (designated as **PE** in the above table). Further, they choose **one open elective** offered by other department (designated as **OE** in the above table).

18UCSC600	Computer Networks	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Various network services and switching networks.
- Protocol design, implementation and performance issues.
- Various network management issues and possible remedies.
- Virtual networks for security issues in Internet Protocol (IP).

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the communication services, protocols and algorithms of packet-switching networks. [Familiarity] [BL-2]	-	1,2,13	-
CO-2	**Explain** the working of TCP/IP layered model of communication and **analyze** the traffic management. [Usage] [BL-3]	4,5	1,2,13	14
CO-3	**Explain** the issues of internet routing protocols, VPN and overlay networks. [Familiarity] [BL-2]	-	1,2,13	-
CO-4	**Explain** the working of ATM layered model of communication. [Familiarity] [BL-2]	-	1,2,13	-
CO-5	**Explain** the need of network management services, associated security issues and **use** modern tools to **perform** network management. [Familiarity] [BL-2]	4,5	1,2,13	-

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	-	3.0	3.0	-	-	-	-	-	-	-	2.0	1.0	-	-

Prerequisites: Knowledge of:

a. Data Communication
b. Operating System.

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools / Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Packet-Switching Networks:** Network services and internal network operations, Packet network topology, Datagrams and virtual circuits, Routing in packet networks, Shortest-path routing, ATM networks.	10	-	1	1	1. **Class Test** (CO-1) 2. **Written test-** CIE / SEE. (CO-1)
2.	**Traffic management at the packet level**: Traffic management at the flow level, Traffic management at the flow-aggregate level. **TCP / IP:** TCP / IP architecture, The Internet protocol, User datagram protocol, Transmission control protocol.	12	-	2	1, 2	1. **Class Test** (CO-2) 2. **Written test-** CIE/SEE (CO-2) 3. **CTA –** Study on Industry relevant tools to perform traffic management. 4. **CTA –** Write C program using RFCs to implement standard protocol using TCP / IP.
3.	**Internet routing protocols**, Multicast routing, DHCP, NAT and Mobile IP. **VPNs, Tunneling, Overlay Networks:** Virtual Private Networks, Multiprotocol Label switching, Overlay networks.	10	2	3	2	1. **Written test-** CIE/SEE (CO-3)

4.	**ATM Networks:** Introduction to ATM networks, BISDN reference model, ATM layer, ATM adaptation layer, ATM signaling, PNNI routing, Classical IP over ATM.	10	-	4	2,3	1. **Class Test** (CO-4) 2. **Written test** - CIE/SEE (CO-4)
5.	**Network Management, Security:** Network management overview, SNMP, Structure of Management information, MIB, Remote network monitoring, Overview of Security and cryptographic algorithms.	10	-	5	3	1. **Written test**- CIE/SEE (CO-5) 2. **CTA** – Study on Industry relevant tools to perform network management. (CO-5)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Alberto Leon-Garcia and IndraWidjaja- Communication Networks' "Fundamental Concepts and Key architectures", 2/E, Tata McGraw-Hill, 2005.
2) Nader F. Mir, "Computer and Communication Networks", 2/E, Pearson Education, 2009.
3) Behrouz A. Forouzan, "Data Communications and Networking", 10/E, Tata McGraw-Hill, 2006.
4) William Stallings, "Data and Computer Communication", 8/E, Pearson Education, 2013.

18UCSC601	Object Oriented Modeling and Design	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs):

This is a 4 credit, 52 contact hours course at undergraduate level focusing on the process of object oriented system modeling, design, patterns and tools used in the industry to enable them to construct software system using various standards and techniques.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial 3	Moderate 2	Low 1
CO-1	**Apply** fundamental Object Oriented concepts in solving problems. [Usage][BL-3]	13,14	1,2,3,15	-
CO-2	**Analyze** problem scenario and **identify** classes/ Objects, their properties and associations. [Usage][BL-3/4]	13	1,2,3,5,15	-
CO-3	**Analyze** problem scenario and **model** the system using UML diagrams. [Usage][BL-3/4]	13	1,2,3,5,15	-
CO-4	**Evaluate** the quality of Object Oriented system in-terms of Cohesion, coupling, sufficiency, completeness and primitiveness. [Usage][BL-5]	-	16	15
CO-5	**Implement** Object Oriented model in any Object Oriented language. [Usage][BL-3]	13,14	1,2,3	15
CO-6	**Identify** and **apply** the appropriate patterns in solving problems. [Usage][BL-3]	-	13,16	1
CO-7	**Propose** the appropriate strategies to incorporate standard quality parameters in the design of a system. [Usage][BL-3]	-	13,16	1

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.7	2.0	2.0	-	2.0	-	-	-	-	-	-	-	2.7	3.0	1.6	2.0

Prerequisites: Knowledge of:

a. Any object oriented Programming Language.
b. Software Engineering.

Course Contents

UNITS	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Review:** Object Oriented Concepts and principles. **Introduction, modeling concepts, class modeling:** Object Orientation, developments themes; Evidence for usefulness of developments; modeling history. Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.	10	-	1	1	1. **Class Test.** (CO-1) 2. **Written test**- CIE / SEE.(CO-1)
2	**Advanced class modeling, state modeling:** Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips. State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips. **Advanced state modeling, interaction modeling:** Advance state modeling: nested state diagrams; nested states; signal generalization; concurrency; a sample state	7	5	2	1	1. **Class test. (CO-2 &3)** 2. **Written test**- CIE/SEE.(CO-2, 3) 3. **CTA- Group Activity/ pair programming**: **Course project** Building reusable objects based on specification and writing test scripts based on OO model. (CO-2,3)

	model; relation of class and state models; practical tips. Interaction modeling: use case models; sequence models; activity models. Use case relationships; procedural sequence models; special constructs for activity models. **Evaluation**: OO system quality in terms of Cohesion, coupling, sufficiency, completeness and primitiveness. **Implementation** : OO design in appropriate language			3 4 5	2	
3	**Patterns–Part 1:** Introduction; layers, pipes and filters, blackboard.Distributed systems: broker; interactive systems: mvc, presentation-abstraction-control.	6	4	6	2	1. **Class test.** (CO-6) 2. **Written test**- CIE/SEE. (CO-6) 3. **CTA- Group Activity: Course project** on building system to demonstrate MVC/PAC (CO-6)
4	**Patterns–Part2:** Adaptable systems: microkernel; reflection. Structural decomposition: whole - part; organization of work: master - slave; access control: proxy. Others: Command Processor, View Handler, Forward Receiver, Client-Dispatcher-Server and publish Subscriber.	7	3	6	3	1. **Class test.** (CO-6) 2. **Written test**- CIE/SEE. (CO-6) 3. **CTA- Group Activity: Course project** on building system to demonstrate whole and part relationship (CO-6)
5	**Quality:** Functionality and architecture; architecture and quality attributes; system quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Relationship of tactics to architectural patterns; Styles	10	-	7	3	1. **Written test**- CIE/SEE.(CO-7)

Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…	**CTA:**Lecture /Training by **Industry expert** onUML tools and OO based Testing.(CO- 2 & 3) **CTA- Group Activity:** use of tools to prepare UML diagrams (CO- 2 & 3) Evaluation is to be based on report submission and oral presentation

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Michael Blaha, James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2/E, Pearson Education, 2007.
2) Frank Buschmann, ReginеMeunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture", A System of Patterns Volume 1, John Wiley and Sons, 2006.
3) Len Bass, Paul Clements, Rick Kazman, "Software Architecture in Practice", 2/E, Pearson Education, 2003.
4) Grady Boochetai, "Object-Oriented Analysis and Design with Applications", 3/E, Pearson Education, 2007.
5) Ali Bahrami, "Object oriented systems development", McGrawHill, 1999.
6) Mary Shaw and David Garlan, "Software Architecture Perspectives on an Emerging Discipline", Prentice-Hall of India, 2007.

18UCSL602	Computer Networks Lab	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

1. Configuration of networking devices.
2. Troubleshooting IPv4 and IPv6 Addressing
3. DHCP and DNS Servers

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Configure** the Cisco networking devices like routers, switches, and hubs etc using commands. [Usage] [BL-3]	13,14	1,2,3	5,15
CO-2	**Simulate** different topologies/network using Cisco Packet Tracer application. [Usage] [BL-3]	13,14	1,2,3	5,15
CO-3	**Manage** IP addresses and troubleshooting [Usage] [BL-3]	13,14	1,2,3	5,15
CO-4	**Manage** applications like Web, Email, DHCP, DNS and FTP [Usage] [BL-3]	13,14	1,2,3	5,15
CO-5	**Study** on Industry relevant tools to perform traffic management. [Usage] [BL-3]	13,14	1,2,3	5,15
CO-6	**Write** C program using RFCs to implement standard protocol using TCP / IP. [Usage] [BL-3]	13,14	1,2,3	5,15

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	1.0	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Sl. No.	Term Work

T1	**Building a Simple Network:** Part 1: Set Up the Network Topology (Ethernet only) Part 2: Configure PC Hosts Part 3: Configure and Verify Basic Switch Settings **Learning Outcomes:[CO-1,2]**
T2	**Connecting a Wired and Wireless** Part 1: Connect to the Cloud Part 2: Connect Router0 Part 3: Connect Remaining Devices Part 4: Verify Connections Part 5: Examine the Physical Topology **Learning Outcomes:[CO-1,2]**
T3	**Troubleshooting IPv4 and IPv6 Addressing** Part 1: Troubleshoot First Issue Part 2: Troubleshoot Second Issue Part 3: Troubleshoot Third Issue **Learning Outcomes:[CO-2,3]**
T4	**Configuring IPv6 Addresses on Network Devices** Part 1: Set Up Topology and Configure Basic Router and Switch Settings Part 2: Configure IPv6 Addresses Manually Part 3: Verify End-to-End Connectivity **Learning Outcomes:[CO-2,3]**
T5	**Designing and Implementing a Subnetted IPv4 Addressing Scheme** Part 1: Design a Network Subnetting Scheme Part 2: Configure the Devices Part 3: Test and Troubleshoot the Network **Learning Outcomes:[CO-1,2,3]**
T6	**Web and Email** Part 1: Configure and Verify Web Services

	Part 2: Configure and Verify Email Services **Learning Outcomes:[CO-3,4]**
T7	**DHCP and DNS Servers** Part 1: Configure Static IPv4 Addressing Part 2: Configure and Verify DNS Records **Learning Outcomes:[CO-3,4]**
T8	**FTP Servers** Part 1: Configure FTP Services on Servers Part 2: Upload a File to the FTP Server Part 3: Download a File from the FTP Server **Learning Outcomes:[CO-3,4]**
T9	**Troubleshooting Connectivity Issues** The objective of this Packet Tracer activity is to troubleshoot and resolve connectivity issues, if possible. Otherwise, the issues should be clearly documented and so they can be escalated. **Learning Outcomes:[CO-1,2,3,4]**
T10	**Study** on Industry relevant tools to perform traffic management. **Learning Outcomes:[CO-5]**
T11	**Write** C program using RFCs to implement standard protocol using TCP / IP. **Learning Outcomes:[CO-6]**

18UCSL603	Industry Oriented Programming Practices Lab	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The study of the various programming practices through demonstration,
- The implementation in different programming paradigm/languages in terms of principles and benefits it offers in the system design and development.
- The coding guidelines encompassing all aspects of code development to enable them to be a professional software developer.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	Explain the various styles, standards of different programming paradigm and Write simple programs. [Usage] [BL-3]	-	1,2,3,13,14, 15	5
CO-2	Write simple programs to explain the code quality and assess the quality of the given code. [Usage] [BL-3]	-	1,2,3,13,14, 15	5
CO-3	Illustrate the need for parallelization of serial parograms and its impact on performance [Usage] [BL-3]	-	1,2,3,13,14, 15	5
CO-4	Write simple scripts for given system administration. [Usage] [BL-3]	-	1,2,3,13,14, 15	5
CO-5	Generate the technical reports and effectively communicate through presentation slides and tools. [Usage] [BL-3]	10	-	-

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	1.0	-	-	-	-	3.0	-	-	2.0	2.0	2.0	-

Prerequisites: Knowledge of any computer programming and Software Engineering is essential. Introductory level exposure to different domains like OS, Network, DBMS, and Web Technology is desirable.

Contents:

Naming convention & consistency; comments; Consistent indentations; Spaces; Structure and its impact on readability and efficiency; Organization of: Programs, Function, File and folder. Guidelines for portability and performance; Separation of code and data; Documentation; Writing reusable codes; Programming Paradigm specific style and practices (Object orientation V/S procedural etc.).;Writing code with quality in terms of: Robustness, Maintainability, Testability, Adaptability, Availability, Usability, sufficiency, completeness, Primitiveness, Cohesiveness, and Coupling; Communications, Documentations, Proposals/ technical writing, Sharing of information in a group, Conduction of meeting/ review, versioning.

Note:

1. This course is to be conducted in the **laboratory** by demonstrating various programming/industry practices.
2. **Adjunct faculty** from industry may be used to give industry relevance to the course coverage.
3. Students are expected to **write programs / do the course work** using knowledge gained and prepare **reports** based on the course work assigned by a course teacher.
4. Students are expected to give presentation on a chosen topic approved by the course teacher. **Internal evaluation** is to be based on the **continuous evaluation** of each activity of the **course work, report** preparation and **presentation**.
5. Final evaluation (SEE) is to be based on oral exams based on implementation.

Reference Books:

1. Brain W. Kernighan, Rob Pike- The Practice of Programming, Pearson education, 2008.
2. Knowledge repository created by various industries available on the internet

18UCSL604	Minor Project – 2	(0-0-4) 2

Contact Hours:48

Course Learning Objectives (CLOs): Though the specific objectives of this course depend on the Project chosen, below are the generic objectives of this course:

Understand the domain, Analyze through Modeling and Implementation through state of the art technology available. To know Software Engineering Principles: Modeling, Estimation, Design standards and architectural issues through use of Standards etc. Also, write modular programs and handle exceptions to provide reliable solutions, to test and verify the programs for different scenarios.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Identify** and **formulate** the problem.	11,12	1,2,8	6,7
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	11,12, 13	2,3,5,8,16	6,7
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	11,12,14	3,4,5,8,16	-
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap.	11,12,15	5,8,16	-
CO-5	**Prepare** the report and **communicate** effectively through presentation.	10,11	8,9	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	2.0	2.0	1.0	1.0	2.0	2.0	3.0	3.0	3.0	3.0	3.0	3.0	2.0

Prerequisites: Knowledge of

a) Software Engineering concepts
b) Any Programming Language

Guidelines for the Conduction of Project Work:

A project may belong to any of the following categories:

1. **Learning Centric:** Here the output of the project activity is enhancement of the student's knowledge. Here the student chooses a work with the intension to gain the knowledge.
2. **Application**: Traditional Software Engineering project, with appreciable complexity. Scenario of demonstration had to be made clear and Completeness is important.
3. **Product Base:** The final output is a novel one which may be the assembly of several existing subsystems. Completeness is important till the user manual.
4. **Research Oriented**: A research problem of student's interest. Achievement would be a publication in the IEEE/ACM international/national conferences.

General Instructions to Students:

- Students are expected to perform extensive literature survey, identify problem statements and prepare synopsis in consultation with project guide/supervisor. Students are expected to submit Synopsis- Initial (Registration Phase-1) approved by project guide, to the project coordinator as per the schedule notified. A copy is to be maintained with students and the guide. This registration/ Initial synopsis contains the description of the project concept created and acts as a base line for design and Implementation of the system.
- Notification/schedules and evaluation procedures will be sent to all students in the Google groups created in the department.
- Batch size is of maximum 4 students. Mixing of divisions is not permitted unless it is the sponsored/research project and request is initiated by project supervisor/guide.
- Students may approach the faculty members of CSE department for choosing them as project guide/supervisor.
- A teacher can guide/supervise maximum of 2 UG project groups. However in special case, a DUGC (Project Coordinating Team, **PCT**, working on the behalf of DUGC) may assign additional project groups to a given teacher.
- The department will financially support presentations of publication of paper only for reputed conference publications.
- DUGC/Project coordinating team (**PCT**) may be consulted for any other/ missing information.

Evaluation Procedures:

a) CIE Marks for the project is to be awarded by project guide/supervisor and SEE marks are to be awarded by examiners (appointed by DUGC) and PCT based on the overall guidelines and project classifications/rubrics by looking into Software Engineering aspects & usefulness w.r.t research/innovation/technology/industry trends through formal interactions and presentations. However, each team is expected to give a formal complete presentation of their work at the end of each phase (1 to 4) to project coordinating team (PCT).

b) PCT evaluates the work and suggest the corrections and observation. All project teams are expected to incorporate these changes in their work. These observations will be made available to Project guides and SEE examiners, which will help them to evaluate and award marks during assessment process.
c) Project guides should keep track of all interactions they do with project team members on weekly basis.
d) All Projects are evaluated and individual students are awarded a grade based on the grading criteria set.
e) Individuals' grade/marks is decided based on both CIE and SEE marks/grade.
f) A project is considered for possible award of S grade; if and only if its 'research outputs' / 'product innovation outputs' results in to a publication of a paper. In a special case, innovative or sponsored business applications with focus on recent technological trends/ Industry trends catering for societal needs may be considered. If contents of paper/research output are not at satisfactory level, then, the assessment results in to appropriate lower grades.
g) A project is considered for possible award of A grade; if and only if it demonstrate product development skills in core system or systems level applications using all aspects of software engineering product development phases like: requirements, design, implementation, testing including standards like: use of design/architectural patterns, coding standards, use of tools for design/testing, programming practices, documentation and reporting etc..,. In a special case, innovative or sponsored business applications with focus on recent technological trends / Industry trends catering for societal needs may be considered.
h) If project work contribution/ output is not at satisfactory level, then the assessment results @ appropriate lower grades. All grades/marks are awarded based on individual contributions evaluated from software engineering perspectives **specified in the rubrics**.
i) Expected important features:
    - Report preparation using Latex.
    - Online plagiarism check report is to be enclosed in the report.
    - Use of IEEE standard. Ex: reference listing and use of PPTs for presentation etc…
    - Use of software tools. Ex: for Design, version control, UI design, Testing etc…
    - Conduction of workshop/ training on technology/domain to students and Preparation of training material/manual (.doc & .pdf).
    - Publication of paper based on outcome of the project.
    - Submission of proposal to KSCST (Govt. of Karnataka) or other agencies for funding.
    - Any other features suggested by guides/coordinators from time to time.

j) Marks Weightage and Various parameters for project evaluation for both CIE and SEE level @ 6th Semester.

k)

Sl.No.	Parameter for Assessment	Marks (%)
1	Requirements Analysis (SRS): Abstract and Detailed.	20
2	Design Specification; Use of: UML diagrams, architecture diagram, ER diagram, Patterns etc… Proper cohesiveness and coupling of various components in the system design.	35
4	Use of Tools and standards.	5
5	Implementation: Code documentation, style, robustness, maintainability, Testability, Usability (User Experience) etc…	10
6	Testing: for every scenario of all use cases identified.	10
7	Final Oral Presentation (viva-voce) (IEEE Standards for slides, oral presentation techniques etc…)	10
8	Project Reports- Final and Intermediate if any: preparation using LATEX and plagiarism check	10

Note:

1. Sufficient and completeness of each parameter is to be seen while awarding marks for individual students.
2. Marks for individual students in a given project team may vary based on individuals 'learning outcomes.

- All project teams are expected to participate in the project exhibition arranged at department level. Project teams are expected to share their project experience to all their juniors and motivate them to take-up challenging work as their project work. During project exhibition, Top 2 projects from the batch will be awarded with a certificate of appreciation at the end of academic year.

18UHUL605	Soft Skills / Aptitude	(0-0-2) 1

Contact Hours: 24

This course will be dealt at the Institute level.

This is included with an objective of improving the communication skills, proficiency in English language and aptitude ability of the student. This is a credit course and aimed to enhance the employability. Both the internal and external resource persons shall be engaged in imparting the related knowledge and shall have only CIE as the evaluation component. There shall be one test conducted at the end for 25 marks in Aptitude testing and there shall be one presentation by the student for 25 marks or any other suitable testing components. The arrangement for CIE evaluation is to be done by the department and maintain the relevant documents.

18UCSE606	UNIX Systems Programming	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs):

This course facilitates the students to get familiarity with system calls, UNIX kernel structure and use of standards like ANSI and POSIX in programming.

Course Outcomes (COs):

CO	Description of the Course Outcomes: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Describe** the ANSI and POSIX standards used in UNIX operating system and programming **[Familiarity] [BL-2]**	-	1	13
CO-2	**Explain** the UNIX file types and **demonstrate** the use of UNIX file APIs in programming **[Usage] [BL-3]**	13,14	1,2,3	15
CO-3	**Identify** the process management activities of UNIX and **write** programs that make use of processes and their environment **[Usage] [BL-3]**	13,14	1,2,3	15
CO-4	**Describe** the use of signals in UNIX and **illustrate** the use of signals in programs **[Familiarity] [BL-2]**	13,14	1,2,3	15
CO-5	**Explain** the need of daemons in UNIX and **identify** the use of daemons in UNIX OS **[Familiarity] [BL-2]**	13,14	1,2,3	15
CO-6	**Explain** inter process communication mechanisms of UNIX and **write** programs to demonstrate IPCs for client-server interactions **[Usage] [BL-3]**	13,14	1,2,3	15

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Knowledge of:

- UNIX operating system and its commands
- Operating Systems Fundamentals
- Basics of Networking

Course Contents

Modules	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Duration in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**ANSI and POSIX Standards:** UNIX and ANSI Standards – ANSI C standard, POSIX standards, POSIX environment, POSIX feature test macros, limits checking at compile time and run time; UNIX and POSIX APIs – POSIX APIs, UNIX and POSIX development environment, API common characteristics	4	-	1	1	1. CTA – Quiz (CO-1) 2. Written test- CIE / SEE (CO-1)
	UNIX Files: file types, UNIX and POSIX file systems, UNIX and POSIX file attributes, inodes in UNIX system V, UNIX kernel support for files, directory files, hard and symbolic links	3	-	2	1	1. CTA - Quiz (CO-2) 2. Written test- CIE / SEE (CO-2)
2	**UNIX File APIs:** General file APIs, open, read, write, close, fcntl, lseek, link, unlink, stat, fstat, lstat, access, chmod, fchmod, chown, fchown, lchown, utime, file and record locking, directory file APIs, device file APIs, FIFO file APIs, symbolic link file APIs	4	3	2	1	1. CTA –Quiz (CO-2) 2. Written test- CIE/SEE (CO-2) 3. CTA – Programming assignments that make use of UNIX file APIs. (CO-2)
3	**Environment of a UNIX Process:** Introduction, main function, process termination, command line arguments, environment list, memory layout of a C program, alloca function, environment variables, setjmp and lonjmp functions,	3	1	3	2	1. CTA – Quiz (CO-3) 2. Written test- CIE/SEE (CO-3) 3. CTA –Programming assignments that make use of

	getrlimit and setrlimit functions **Process Control:** Introduction, process identifiers, fork function, vfork function, exit functions, wait and waitpid functions, race conditions, exec functions, changing user IDs and group IDs, system function	3	2	3	2	environment of UNIX process, fork() / vfork(), wait(), and exec() functions/ system calls (CO-3)
4	**Process Relationships:** Introduction, terminal logins, network logins, process groups, sessions, controlling terminal, job control **Signals and Daemon Processes:** Signals – UNIX kernel support for signals, signal, signal mask, sigaction, sigsetjmp and siglongjmp APIs, kill, alarm, interval timers; Daemon Processes – introduction, daemon characteristics, coding rules, error logging, client-server model	3 4	- 2	3 4, 5	2 3	1. CTA – Quiz (CO-3) 2. **Written test-** CIE/SEE (CO-3) 1. CTA – Quiz (CO-4, 5) 2. **Written test-** CIE/SEE (CO-4, 5) 3. **CTA** - Programming assignments that demonstrate the signal handling capabilities (CO-4)
5	**Interprocess Communication:** Introduction, pipes, message queues, UNIX APIs for message queues, client-server example for message queue, sockets, socket APIs, client-server example for socket	5	2	6	3	1. CTA – Quiz (CO-6) 2. **Written test-** CIE/SEE (CO-6) 3. **CTA** – Programming activity to setup/ simulate client-server communication using pipes, message queues and sockets (CO-6)
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. UNIX System programming using C++, Terrence Chan, Prentice Hall India, 2015
2. Advanced Programming in the UNIX environment, W. Richard Stevens, Pearson Education/ PHI, 2005

18UCSE607	Digital Image Processing	(3-0-0) 3

Contact Hours: 39

Course Learning Objective (CLOs): This course focuses on the following learning objectives:

- To learn fundamental theories and techniques of digital image processing.
- To acquire the skill necessary to explore advanced topics of digital image processing.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course the student will be able to:	Mapping to POs/PSOs		
		Substantial	Moderate	Low
CO-1	**Describe** the principles of Digital Image Processing. [Familiarity] [BL-2]	-	1,2	3,5,13,14
CO-2	**Demonstrate** the image enhancement techniques that include primitives image sensing and acquisition techniques, image formation, image representation & relationship between the pixels. [Usage] [BL-3]	13,14	1,2,3	15
CO-3	**Explain** the basic principles of mathematical morphology & **write** program to **extract** the characteristic features of image using morphological operations. [Usage] [BL-3]	13,14	1,2,3	15
CO-4	**Apply** segmentation techniques for a given application scenario. [Usage] [BL-3]	13,14	1,2,3	15
CO-5	**Explain** and **implement** the core princples of image representation techniques. [Usage] [BL-3]	13,14	1,2,3	15

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	1.8	1.8	-	1.0	-	-	-	-	-	-	-	2.6	2.6	1.0	-

Prerequisites: Knowledge of: Basics of statistics, Linear Algebra.

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools / Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1	**Digital Image Fundamentals** - Introduction, Applications, Fundamental Steps in Digital Image Processing, Elements of visual perception, Image sensing and acquisition, Image Sampling and Quantization, Basic relationships between pixels.	8	-	1, 2	1	1. **Written test**- CIE / SEE. **(CO-1 & 2)**
2	**Intensity Transformations and Spatial Filtering** - Basic Intensity Transformation Functions, Histogram Processing; Fundamentals of Spatial Filtering., Smoothing and Sharpening Spatial filters.	7	1	2	1,2	1. **Class Test. (CO-1 & 2)** 2. **Written test**- CIE/SEE. **(CO-1& 2)** 3. **CTA – Individual / Group Activity: Implement** an algorithm for a given scenario, using available tools / platforms **(CO-1&2)**
3	**Morphological Image Processing**- Erosion and Dilation, Opening and Closing, Hit or Miss Transforms, Basic Morphological Algorithms, GrayScale Morphology.	7	1	3, 6	2	1. **Class Test. (CO-3)** 2. **Written test**- CIE/SEE. **(CO-3)** 3. **CTA – Individual / Group Activity: Implement** an algorithm for a given scenario, using available tools / platforms **(CO-6)**
4	**Image Segmentation**- Point, Line, and Edge Detection, Thresholding, Region-Based Segmentation, Segmentation Using Morphological Watersheds.	7	1	4	2, 3	1. **Class test. (CO-4)** 2. **Written test**- CIE/SEE. **(CO-4)** 3. **CTA – Individual / Group Activity:**

						Implement an algorithm for a given scenario, using available tools / platforms **(CO-4)**
5	**Representation and Description-** Image Representation, Boundary and Regional Descriptors	7	-	5	3	1. **Written test-** CIE/SEE. **(CO-5)** 2. **CTA - Group Activity: Course project** Implementation and testing the design and the code developed. **(CO-6)**
Other performance ensuring measures **Compulsory:** Implementation based assignments. **Others:** Presentation outside college, Certification, any engineering solutions for societal problems, Participation in relevant conference / training program/workshops, **CTA :** Implementation Based Assignment : 80 % (8 Marks) + Others: 20 % (2 Marks) = Total : 100 % (10 Marks)						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Rafel C Gonzalez and Richard E Woods, "Digital Image Processing", 3/E, Pearson Education, 2016.
2. Milan Sonka, Vaclav Hlavac and Roger Boyle, "Image Processing, Analysis and Machine Vision", 2/E, Thomoson Learning, Brooks/Cole, 2001.
3. Anil K Jain, "Fundamentals of Digital Image Processing", Prentice-Hall of India Pvt. Ltd., 1997.
4. B.Chanda, DDuttaMajumder, "Digital Image Processing and Analysis", Prentice-Hall, India, 2002.

18UCSE608	Principles of Programming	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Explain the value of declaration models, especially with respect to programming-in-the-large.
- Identify and describe the properties of a variable such as its associated address, value, scope, persistence, and size.
- Demonstrate different forms of binding, visibility, scoping, and lifetime management.
- Justify the philosophy of object-oriented design and the concepts of encapsulation, abstraction, inheritance, and polymorphism.
- Evaluate the trade offs between the different paradigms, considering such issues as space efficiency, time efficiency (of both the computer and the programmer), safety, and power of expression.
- Design, code, test, and debug programs using the functional paradigm.
- Outline the strengths and weaknesses of the logic programming paradigm.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the notations to describe syntax and semantics of programming languages. [Familarity] [BL-2]	-	-	1
CO-2	**Explain** the principles of imperative programming languages such as binding, scope, control structures, subprograms and parameter passing methods and **write** simple programs to demonstrate these for a given application. [Usage] [BL-3]	13,14	1,2,3	15
CO-3	**Use** principles of object oriented programming features in writing the programs for the given problem scenario. [Usage] [BL-3]	13,14	1,2,3	15
CO-4	**Write** the formal syntax for a specification of functional programming languages. [Familarity] [BL-2]	13,14	1,2,3	15
CO-5	**Write** programs in the Prolog language for given problem scenario. [Usage] [BL-3]	13,14	1,2,3	15

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Knowledge of

a. Programming language (any one).

b. Mathematics

Course Contents

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction**: Toward higher-level languages, Programming paradigms, Language implementation: Bridging the gap, Expression notations, and Abstract syntax trees. **Types: Data Representation**: Elementary data types – Data objects, Variables and Constants, Data types, Declarations, Type checking and type conversion. Numeric data types, Enumerations, Booleans, Characters. Structured data types - Structured data objects and data types, Specification of data structure types, Implementation of data structure types, Vectors and arrays, Records.	8	-	1	1	1. **Written test**- CIE / SEE. (CO-1)
2.	**Imperative Programming:** Basic Statements, structured sequence control, handling special cases in loops, programming with variants, proof rules for partial correctness. Procedure activations – Simple call-return subprograms, Parameter passing methods, Scope rules for names, Nested scopes in the source text, Activation records, Lexical scope: Procedures as in C.	8	-	2	1	1. **Written test**- CIE/SEE.(CO-2) 2. **CTA - Implementation** of the concept of parameter passing techniques in imperative programming languages. (CO-2)
3.	**Object-Oriented Programming:** Object-oriented design, Encapsulation and information-hiding, Separation of behavior and implementation, Classes and subclasses,	7	-	3	2	1. **Written test**– CIE/SEE (CO -3) 2. **CTA - Implementation** of the concept of polymorphism and inheritance of

	Inheritance (overriding, dynamic dispatch), Polymorphism (subtype polymorphism vs. inheritance).					object oriented programming. (CO -3)
4.	**Functional Programming**: Lamda Calculus, Elements of functional programming – A Little language of expressions, Types: values and operations, Approaches to expression evaluation, Lexical scope, Type checking. Functional programming in a typed language -Exploring a list, Function declaration by cases, Function as first-class values, ML: Implicit types, Data types, Exception handling in ML. Functional programming with lists - Scheme, a dialect of lisp, The structure of lists, List manipulation.	9	-	4	2, 3	1. **Written test**– CIE/SEE (CO -4) 2. **CTA - Implementation** of the concept of Functional programming language. (CO-4)
5.	**Logic Programming:** Predicate Logic: FOL, Computing with relations, Introduction to prolog, Data structures in prolog, Programming techniques, Control in prolog.	7	-	5	3	1. **Written test**- CIE/SEE. (CO-5) 2. **CTA - Implementation** of the concept of Logic programming language. (CO-5)
	Other Performance Ensuring Measures Implementation based assignments, Survey and Presentation					1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Ravi Sethi, Programming languages, concepts & constructs Addison Wesley 2/E, 2009
2. Terrence W.Pratt, Programming languages Design and Implementation Pearson Education, 4/E, 2009.
3. Robert W Sebesta, Concept of Programming language Pearson Education, 11/E, Pearson Education, 2019

18UCSE609	Data Mining	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs):

This course focuses on the following learning perspectives:

- Compare and contrast different conceptions of data mining as evidenced in both research and application.
- Explain the role of finding associations in commercial market basket data.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Describe how to extend a relational system to find patterns using association rules.
- Evaluate methodological issues underlying the effective application of data mining.
- Identify and characterize sources of noise, redundancy, and outliers in presented data.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs supported by appropriate evidences shown in Performance Ensuring Measures-PEM		
		Substantial 3	Moderate 2	Low 1
CO-1	**Analyze** the given scenario and **perform** data pre-processing using appropriate methods. [Usage] [BL-3]	13,14	1,2,3	15
CO-2	**Predict** group membership for data instances using Hunt's algorithm, Rule based and Nearest Neighbor classification techniques. [Usage] [BL-3]	13,14	1,2,3	15
CO-3	**Analyse** the data using Apriori and Non-Apriori based algorithms in order to determine **patterns** from the large data sets. [Usage] [BL-3]	13,14	1,2,3	15
CO-4	**Explain** and **apply** the partitional and heirarchical clustering techniques for a given data set. [Usage] [BL-3]	13,14	1,2,3	15
CO-5	**Illustrate** the statistical, proximity based and clustering based techniques to detect the outliers in the data set. [Familarity] [BL-2]	13,14	1,2,3	15

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Knowledge of Statistics at introductory level.

Course Contents:

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools: Class Test, Written Test, Quiz,Programming based assignments, Group activity/ Course Project, Peer Learning & Training from Industry experts.
		Theory	Practice			
1.	**Introduction and Basic Concepts:** What is Data Mining? Motivating Challenges, The Origins of Data Mining, Data Mining Tasks. **Data:** Types of Data, Data Quality, Data pre-processing, Measures of Similarity and Dissimilarity.	8	-	CO-1	1	1. **Class Test** (CO-1) 2. **Written test**- CIE / SEE. **(CO-1)**
2.	**Classification:** Preliminaries, General approach to solving a classification problem, Decision tree induction, Model over fitting, Evaluation of the performance of a classifier, Rule based Classification, Nearest Neighbour classifiers.	8	-	CO-2	1	1. **Class test** (CO-2) 2. **Written test**- CIE/SEE.(CO-2) 3. **CTA** - Implementation of classification algorithms using Industry standard tools/programming (CO-2)
3.	**Association Analysis:** Problem Definition, Frequent Itemset Generation, Rule Generation, Compact Representation of Frequent Itemsets, FP tree Growth algorithm, Evaluation of Association Patterns.	8	-	CO-3	2	1. **Class Test/Quiz** (CO-3) 2. **Written test**– CIE/SEE (CO -3) 3. **CTA** – Implementation of association algorithms using Industry standard tools/programming (CO-3)

4.	**Cluster Analysis:** Overview, K-means algorithm, Agglomerative Hierarchical Clustering, DBSCAN, Prototype-based Clustering.	8	-	CO-4	2, 3	1. **Class Test/Quiz** (CO-4) 2. **Written test** - CIE/SEE. (CO-4)
5.	**Anamoly Detection:** Preliminaries, Statistical approaches – Parametric and Non Parametric models, Proximity based approaches – Distance and Density based techniques, Clustering based techniques.	7	-	CO-5	3	1. **Written test**- CIE/SEE.-(CO-5)
Other performance ensuring measures: Collection of Data Sets, Analysis of Data sets in terms of number of attributes, type of attributes etc. Industry standard data mining tools : Weka, Rapid miner etc.						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics / technological trends. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson Education, 3rd Edition, 2014.
2. Data Mining – Concepts and Techniques - Jiawei Han &MichelineKamber, Morgan Kaufmann Publishers, Elsevier, 3rd Edition, 2011.
3. Data Mining Techniques – Arun K Pujari, Universities Press, 3rd Edition, 2013.
4. Insight into Data Mining, K. P. Soman, S. Diwakar, V. Ajay, PHI, 2008.

18UCSE610	Advanced Data Structures and Algorithms	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Asymptotic and Amortized Analyses
- Linear sorting algorithms
- Advanced data structures such as Heaps, B-trees, Red-Black trees etc.
- String matching algorithms

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs supported by appropriate evidences shown in Performance Ensuring Measures-PEM		
		Substantial 3	Moderate 2	Low 1
CO-1	**Analyze** the performance of the given algorithm using asymptotic notations and amortized techniques. [Usage] [BL-3]	-	1,2	-
CO-2	**Explain** the working and assumptions of linear sorting methods and **apply** them to solve a given problem. [Usage] [BL-3]	-	2,13	1
CO-3	**Compare** the working of string matching algorithms and **use** them appropriately in developing applications. [Familarity] [BL-2]	-	2,13	1
CO-4	**Build** and **perform** the operations on heap structures. [Usage] [BL-3]	-	2,13	1
CO-5	**Build** and **perform** the operations on search structures. [Usage] [BL-3]	-	2,13	1
CO-6	**Use** the hash tables for the implementation of dictionary operations. [Familarity] [BL-2]	-	2,13	1
CO-7	**Choose** the appropriate data structure and **use** relevant algorithms to solve problems in different domains including project management [Familarity] [BL-2]	-	2,13	1,11

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.3	2.0	-	-	-	-	-	-	-	-	1.0	-	2.0	-	-	-

Prerequisites: Knowledge of

a. Programming language (any).
b. Data Structures.
c. Algorithms.

Course Contents:

UNIT	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools: Class Test, Written Test, Quiz,Programming based assignments, Group activity/ Course Project, Peer Learning & Training from Industry experts.
		Theory	Practice			
1.	**Introduction to Complexity Analysis:** Review of Asymptotic notations and their properties, Amortized analysis – Aggregate, Accounting and Potential methods. **Linear Sorts:** Counting Sort, Bucket Sort, and Radix Sorting with Analysis for all algorithms.	8	-	CO-1 CO-2	1	1. **Written test**- CIE / SEE. **(CO-1,2)**
2.	**String Matching:** Naive algorithm; Rabin-Karp algorithm; String matching with Finite automata, KMP algorithm, Boyer-Moore algorithm.	8	-	CO-3	2	1. **Written test**- CIE/SEE.**(CO-2)** 2. **CTA - Implementation** of a string matching algorithm. **(CO-2)**
3.	**Heap Structures:** Binomial heaps, Fibonacci heaps. **Search Structures:** 2-3 trees, 2-3-4 trees, B-trees, B^+ trees, Red-black trees.	8	-	CO-4 CO-5	2	1. **Quiz (CO-3)** 2. **Written test**– CIE/SEE **(CO -3)** 3. **CTA** – Implementation of algorithms of B^+ tree operations
4.	**Hashing:** Direct Address Tables, Hash Tables, Collision Resolution by Chaining – Analysis, Hash Functions – Properties, Division and Multiplication methods, Universal Hashing, Open Addressing – Linear and Quadratic Probing, Double hashing.	8	-	CO-6	3	1. **Quiz (CO-4)** 2. **Written test** - CIE/SEE. **(CO-4)**

5.	**Applications:** Huffman coding, Garbage collection and compaction, Min-Cut Max-Flow algorithm, Activity networks.	7	-	CO-7	3	1. **Written test**- CIE/SEE.-**(CO-5)** 2. **CTA - Assignment** on Implementation of Huffman coding algorithm, Min-Cut Max Flow algorithm.
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics. 2. **Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1. Thomas H.Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein-"Introduction to Algorithms", 3/E, PHI 2009.
2. E. Horowitz, S.Sahni and Dinesh Mehta- "Fundamentals of Data structures in C++", Galgotia, 2006.
3. Anany Levitin, "Introduction to the Design and analysis of algorithms", Pearson Education 3/E 2011

18UCSE611	Pattern Recognition	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Fundamentals of pattern recognition system
- Feature extraction and pattern classification algorithms.
- Unsupervised classification or clustering techniques
- Applications of pattern classification algorithm for a pattern recognition problem

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs supported by appropriate evidences shown in Performance Ensuring Measures-PEM		
		Substantial 3	Moderate 2	Low 1
CO-1	**Explain** the basic structure and inherent issues of the pattern recognition problems. [Familarity] [BL-2]	-	1, 13	2, 3, 4
CO-2	**Apply** feature types and classification techniques like Bayesian classifier and its derivatives in solving problems. [Usage] [BL-3]	-	1, 13	2, 3, 4
CO-3	**Compute** the probability density using parametric, non-parametric and linear discriminant functions. [Usage] [BL-3]	-	1, 13	2, 3, 4
CO-4	**Distinguish** supervised learning methods from the unsupervised ones and **apply** learning methods to the classifier design. [Usage] [BL-3]	-	1, 13	2, 3, 4
CO-5	**Use** non metric methods to classify the models that can be described by logical rules. [Usage] [BL-3]	-	1, 13	2, 3, 4
CO-6	**Apply** a suitable clustering method to solve a given problem. [Usage] [BL-3]	-	1, 13	2,3, 4

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	1.0	1.0	1.0	2.0	-	-	-	-	-	-	-	2.0	3.0	3.0	-

Prerequisites: Knowledge of Statistics, Linear Algebra and programming

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction:** What is pattern recognition?,Pattern Recognition System; The Design Cycle; Learning and Adaptation. Clustering vs. Classification; Applications; **Features:** Feature vectors - Feature spaces - Problem of feature identification Feature selection and feature extraction.	8	-	1	1	1. **Written test**- CIE / SEE. (CO-1)
2.	**Bayesian Decision Theory:** Introduction, Bayesian Decision Theory; Continuous Features, Minimum error rate classifiers, discriminant functions, and decision surfaces; The normal density; Discriminant functions for the normal density. **Maximum-Likelihood and Bayesian Parameter Estimation:** Introduction; maximum-likelihood estimation; Bayesian Estimation; Bayesian parameter estimation: Gaussian Case, general theory; Hidden Markov Models.	8	-	2	1 2	1. Class test / Quiz (CO-2) 2. **Written test**- CIE/SEE.(CO-2) 3. CTA – **Individual / Group Activity: Implement** Baysein classifier algorithm for a given scenario, using available tools / platforms and programming language. (CO-2)
3.	**Non-Parametric Techniques:** Introduction; Density Estimation; Parzen windows; K Nearest- Neighbor Estimation; The Nearest- Neighbor Rule; Metrics and	8	-	CO3	IA2	1. Class Test/Quiz (CO-3) 2. **Written test**– CIE/SEE.(CO-3) 3. CTA – **Individual / Group**

	Nearest-Neighbor Classification. **Linear Discriminant Functions:** Introduction; Linear Discriminant Functions and Decision Surfaces; Generalized Linear Discriminant Functions; The Two-Category Linearly Separable case; Minimizing the Perception Criterion Functions; Relaxation Procedures; Non-separable Behavior; Minimum Squared-Error procedures; The Ho-Kashyap procedures.					**Activity: Implement** K nearest neighbor algorithm for a given scenario, using available tools / platforms and programming language. **(CO-3)**
4.	**Support Vector Machines and Kernel based method**: Introduction, obtaining the optimal hyperplane SVM formulation with slack variables; nonlinear SVM classifiers Kernel Functions for nonlinear SVMs; Mercer and positive definite Kernels. Support Vector Regression and ε-insensitive Loss function, examples of SVM learning.	**8**	**-**	**4**	**3**	1. **Written test**- CIE/SEE. **(CO-4)** 2. **CTA- Quiz/ Class test (CO-4)** 3. **CTA – Individual / Group Activity: Implement** support vector machine for a given scenario, using available tools / platforms and programming language. **(CO-4)**
5.	**Non-Metric Methods:** Introduction; Decision Trees; CART; Recognition with Strings; Grammatical Methods. **Unsupervised Learning and Clustering:** Introduction; Mixture Densities and Identifiably; Maximum-Likelihood Estimates; Application to Normal Mixtures; Unsupervised Bayesian Learning; Data Description and Clustering; Criterion Functions for Clustering.	**7**	**-**	**5** **6**	**3**	1. **Written test**- CIE/SEE. **(CO-5, 6)** 2. **CTA:- Quiz/ Class test (CO-5, 6)** 3. **CTA – Individual / Group Activity: Implement** any clustering algorithm for a given scenario, using available tools / platforms and programming language. **(CO-5,6)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						1. Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics. 2. **Quiz** is to be conducted based on the Webinar and is to be considered

	as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Richard O. Duda, Peter E. Hart, and David G.Stork "Pattern Classification", 2/E, Wiley-Interscience, 2012.
2) Earl Gose, Richard Johnsonbaugh, Steve Jost "Pattern Recognition and Image Analysis" - Pearson Education, 2007.
3) V Susheela Devi, M NarsimhaMurthy, Pattern Recognition (An Introduction), , Universities Press, 2011.

18UCSE612	Embedded Systems	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Discuss the major components that constitute an embedded system.
- Implement small programs to solve well-defined problems on an embedded platform.
- Develop familiarity with tools used to develop in an embedded environment.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs and PSOs		
		Substantial	Moderate	Low
CO-1	**Explain** the design considerations in selecting components for embedded systems, both hardware and software. [Familarity] [BL-2]	-	13,14,15,16	1,2,3,4,5
CO-2	**Design** a system using industry relevant automation tools for the given application scenario. [Usage] [BL-3]	13,14,15,16	1,2,3,4,5	-
CO-3	**Explain** the fundamentals of RTOS and its usage in firmware development for Embedded Systems. [Familarity] [BL-2]	-	13,14,15,16	1,2,3,4,5
CO-4	**Implement** basic kernel services of VxWorks and MicroC/OS-II [Usage] [BL-3]	13,14,15,16	1,2,3,4,5	-
CO-5	**Explain** the basic philosophy and features of RISC V Processor instruction set. [Familarity] [BL-2]	-	13,14,15,16	1,2,3,4,5

Note: BL- Bloom's Level

POs/ PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.4	1.4	1.4	1.4	1.4	-	-	-	-	-	-	-	2.4	2.4	2.4	2.4

Prerequisites: Knowledge of:

1. Computer Architecture
2. Microcontroller

Course Contents

Module	Contents	Performance Ensuring Measures- PEM [Hours/ Course Outcomes/ Scope of Internal Assessments/ Assessment Tools/Evidences]				
		Durations in Hours		Course Outcomes	IAs	Assessment Tools
		Theory	Practice			
1.	**Introduction:** Core of the Embedded System, Memory, Sensors and Actuators, Communication Interface, Embedded Firmware, Other System Components.Characteristics and Quality Attributes of Embedded Systems. **Hardware Software Co-Design and Program Modelling:** Fundamental Issues in Hardware Software Co-Design, Computational Models in Embedded Design, Introduction to Unified Modelling Language, Hardware Software Trade-offs.	9	-	1	1	1. **Class Test** (CO-1) 2. Case study 3. **Written test**- CIE / SEE. (CO-1)
2.	**Embedded Hardware Design and Development** :EDA Tools, How to Use EDA Tool, Schematic Design – Place wire, Bus , port, junction, creating part numbers, Design Rules check, Bill of materials, Net list creation , PCB Layout Design – Building blocks.	7	-	2	1,2	1. **Class test / Quiz** (CO-2) 2. **Written test**- CIE/SEE.(CO-2) 3. **CTA**-Lab Assignments.(CO-2)
3.	**Real-Time Operating System (RTOS) based Embedded System Design**: Operating System Basics, Types of OS, Tasks, Process and Threads, Multiprocessing and Multitasking, Task Scheduling,	7	-	3	2	1. Class Test/Quiz-(CO-3) 2. Lab Assignments-Multithreaded applications (CO-3) 3. **Written test**– CIE/SEE. (CO-3)

	Threads, Processes and Scheduling: Putting them altogether, Task Communication, Task Synchronization, Device Drivers, How to Choose an RTOS					
4.	An introduction to Embedded System Design with VxWorks and MicroC OS II RTOS	**8**	-	**4**	**2,3**	**1.** Class Test/Quiz **(CO-4)** **2.** Programming assignments like Task creation and management. **(CO-4)** **3. Written test**- CIE/SEE. **(CO-4)**
5.	**RISC V:** Philosophy of RISC V. Introduction to instruction set architecture manual, volume 1. [Chapter 1 and 2]	**8**	-	**5**	**3**	**1. Written test**- CIE/SEE. **(CO-5)** **2. CTA**-Case study**(CO-5)**
Other performance ensuring measures Like Industrial Visits, Course Projects, Implementation based assignments. Survey & Presentation, Certification, Conducting workshops/training programs, paper presentation, hobby projects, any engineering solutions for societal problems, Participation in relevant conference/training program/workshops etc…						**1.** Minimum two **Online Webinars** from **Alumni** connecting **industry to class room** on related topics. **2. Quiz** is to be conducted based on the Webinar and is to be considered as part of CTA.

Note: CTA: Course Teacher's Assessment, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. Procedure for conduction of IAs and SEE will be notified by the office of the Dean academic program and is common to all courses and programs.

Reference Books:

1) Shibu K V, "Introduction to Embedded Systems", Tata McGraw Hill Education Private Limited, 2012
2) James K Peckol, "Embedded Systems – A contemporary Design Tool", John Weily, 2008
3) The RISC-V Instruction Set Manual, Volume I: User Level ISA, Document Version 2.2

18UXXO6XX	Open Elective	(3-0-0) 3

Contact Hours: 39

Note: One of the electives from the pool mentioned above will be floated as an open elective. This elective is open for the students of CSE department and other departments. However, if the CSE students do not opt the open elective offered by the CSE department, then the CSE students will have to opt an elective offered by other departments.

VII Semester

Course Code	Course Category	Course Title	Teaching		Examination				
			L-T-P (Hrs/Week)	Credits	CIE	Theory (SEE)		Practical (SEE)	
					Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UCSC700	PC	Artificial Intelligence and Machine Learning	**4-0-0**	4	50	100	3	-	-
18UCSC701	PC	Advanced Computer Architecture	**4-0-0**	4	50	100	3	-	-
18UCSL702	PC	Artificial Intelligence and Machine Learning Lab	**0-0-2**	1	50	--	--	50	3
18UCSL703	PC	Major Project Phase-1	**0- 0 -4**	2	50	--	--	50	3
18UCSL704	PC	Internship	**4weeks**	2	50	--	--	50	3
18UCSE705	PE	Computer Graphics	**3-0-0**	3	50	100	3	-	-
18UCSE706	PE	Software Testing	**3-0-0**	3	50	100	3	-	-
18UCSE707	PE	Web Technology	**3-0-0**	3	50	100	3	-	-
18UCSE708	PE	Ad-hoc Networks							
18UCSE709	PE	Operations Research	**3-0-0**	3	50	100	3	-	-
18UCSE710	PE	Multicore Architecture and Programming	**3-0-0**	3	50	100	3	-	-
18UXXO7XX	OE	Open Elective – 2	**3-0-0**	3	50	100	3	-	-
Total			**14 - 0 - 6**	**19**	**350**	**400**	-	**150**	-

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination
L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks
The students will choose **one elective** from the elective list (designated as **PE** in the above table). Further, they choose **one open elective** offered by other department (designated as **OE** in the above table).

VIII Semester

Course Code	Course Category	Course Title	Teaching		Examination				
			L-T-P (Hrs/Week)	Credits	CIE	Theory (SEE)		Practical (SEE)	
					Max. Marks	*Max. Marks	Duration in Hrs.	Max. Marks	Duration In Hrs.
18UCSC800	PC	Distributed Systems and Applications	**4 - 0 - 0**	4	50	100	3	-	-
18UCSL801	PC	Technical Seminar/ Independent study	**0 - 0 - 2**	1	50	-	-	-	-
18UCSL802	PC	Major Project Phase-2	**0-0 -14**	7	50	-	-	50	3
18UCSE803	PE	Cryptography and Network Security	**3-0-0**	3	50	100	3	-	-
18UCSE804	PE	Cloud Computing	**3-0-0**	3	50	100	3	-	-
18UCSE805	PE	Network Management	**3-0-0**	3	50	100	3	-	-
18UCSE806	PE	Mobile Applications Development	**3-0-0**	3	50	100	3	-	-
18UCSE807	PE	Ontology and Semantic Web	**3-0-0**	3	50	100	3	-	-
18UCSE808	PE	Data Science	**3-0-0**	3	50	100	3	-	-
18UXXE/O8XX	PE/OE	Program/Open Elective	**3-0-0**	3	50	100	3	-	-
Total			**10– 0 – 16**	**18**	**250**	**300**	--	**50**	--

Note: BS- Basic Science, PC- Program Core, HU- Humanity Science, CIE- Continuous Internal Examination, SEE- Semester End Examination
L- Lecture, T-Tutorials, P-Practicals. *SEE for theory is conducted for 100 marks and is reduced to 50 marks

The students will choose **one elective** from the elective list (designated as **PE** in the above table). Further, they choose either **one open elective**

offered by other department (designated as **OE** in the above table) or one elective from the elective list (designated as **PE** in the above table). The open electives are offered by the department for other branch students. These open electives are offered in VI, VII, and VIII semesters, one per each semester. The following are the open electives offered by the department (this list is subject to change depending on the market need and requirement). The CSE students opt one open elective per semester of VI, VII, and VIII semesters offered by the other departments.

Open Electives:

1. Introduction to Data Structures and Algorithms
2. Programming in Java/Python
3. Database Management Systems

The End