Detailed Scheme

ACADEMIC YEAR 2020-2021

Dr. Ambedkar Institute of Technology
Bangalore

III - IV (2019-2023 BATCH) (175 Credits)

B.E

Department Of Information Science and Engineering

Vision

- To create **D**ynamic, **R**esourceful, **A**dept and **I**nnovative **T**echnical professionals to meet global challenges.

Mission

- To offer state-of-the-art undergraduate, postgraduate and doctoral programmes in the fields of Engineering, Technology and Management.
- To generate new knowledge by engaging faculty and students in research, development and innovation
- To provide strong theoretical foundation to the students, supported by extensive practical training to meet industry requirements.
- To install moral and ethical values with social and professional commitment.

DEPARTMENT VISION AND MISSION

Vision:

- Imparting quality technical education and preparing professionals to meet Information Technological challenges globally.

Mission:

- Prepare highly capable Information Science engineers through best practices.
- Encourage students to pursue higher education for further growth in the learning process and to promote research in the frontier areas of Information Technology.
- Educate students to take up social and professional responsibilities with ethical values for the betterment of the society.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOS)

PEO1: Graduates will have the ability to become successful computing professionals in the area of Information Science and Engineering.

PEO2: Graduates will be equipped to enhance their knowledge through core engineering and latest technological skills to promote lifelong learning.

PEO3: Graduates will be able to take up social, technical and entrepreneurial challenges in inter disciplinary and multi disciplinary fields.

PROGRAM SPECIFIC OBJECTIVES(PSOS)

PSO1:Students should be able to understand, analyze and adopt principles of programming paradigms by using latest technologies such as Cloud computing, Big data analytics, AI ,Machine Learning and IoT based applications for solving real-world problems.

PSO2: Students should be able to acquire and demonstrate the team work, professional ethics, competence and communication skills while developing software products.

PROGRAMME OUTCOMES (POs)

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

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

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

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

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

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

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

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

Dr. Ambedkar Institute of Technology, Bengaluru-560 056

SCHEME OF TEACHING AND EXAMINATION from Academic Year 2020-21

B.E INFORMATION SCIENCE AND ENGINEERING

Outcome Based Education (OBE) and Choice Based Credit System (CBCS)

III SEMESTER

Sl. No	Course and Course Code		*Course Title*	Teaching Department	Teaching Hours /Week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P					
1	BC	18MA31	Discrete Mathematics and Numerical Methods	Mathematics	2	2	--	03	50	50	100	3
2	PC	18IS31	Computer Organization and Architecture	ISE	4	0	--	03	50	50	100	4
3	PC	18IS32	Data Structures with C	ISE	4	0	--	03	50	50	100	4
4	PC	18IS33	Unix and Shell Programming	ISE	3	2	--	03	50	50	100	4
5	PC	18IS34	Digital principles and logic design	ISE	3	0	--	03	50	50	100	3
6	PC	18IS35	Software Engineering	ISE	3	0	--	03	50	50	100	3
7	PC	18ISL36	Data structure with C Lab	ISE	--		2	03	50	50	100	1
8	PC	18ISL37	Digital principles and logic design Lab	ISE	--		2	03	50	50	100	1
9	HS	18HS31/32	Constitution of India Professional Ethics and Human Rights/ / Env. Studies	Hu/Civ	1	--	--	02	50	50	100	1
10	MC	18HS33	Soft skills (MC)	Humanities	04		--	03	50	-	50	0
				TOTAL	**24**	**04**	**04**	**29**	**500**	**450**	**950**	**24**
Course prescribed to lateral entry Diploma holders admitted to III semester of Engineering programs												
11	MC	18MAD31	Advance Mathematics - I	Mathematics	02	01	--	03	50		50	0

Note: HODs are informed to accommodate one more laboratory in addition to the above courses if needed, without altering the total number of credits (TOTAL: 24).

(a)**The mandatory non – credit courses** Advance Mathematics I and II prescribed at III and IV semesters respectively, to lateral entry Diploma holders admitted to III semester of BE programs shall compulsorily be registered during respective semesters to complete all the formalities of the course and appear for SEE examination.

(b) **The mandatory non – credit courses** Advance Mathematics I and II, prescribed to lateral entrant Diploma holders admitted to III semester of BE programs, are to be completed to secure eligibility to VII semester. However, they are not considered for vertical progression from II year to III year of the programme but considered as head of passing along with credit courses of the programme to eligibility to VII semester

Note: BC: Science Course, PC: Professional Core. Hu: Humanities, MC: Mandatory Course.

HEAD DEPT. OF INFORMATION SCIENCE & ENGG

Dr. Ambedkar Institute of Technology, Bengaluru-56

SCHEME OF TEACHING AND EXAMINATION from Academic Year 2020-21

B.E INFORMATION SCIENCE AND ENGINEERING

Outcome Based Education (OBE) and Choice Based Credit System (CBCS)

IV SEMESTER

Sl. No	Course and Course code		Course Title	Teaching Department	Teaching Hours /Week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P					
1	BC	18MA41	Probability, Queuing Theory and Reliability	Mathematics	2	2	--	03	50	50	100	3
2	PC	18IS41	Microcontroller and Embedded Systems	ISE	3	0	--	03	50	50	100	3
3	PC	18IS42	Design and Analysis of Algorithm	ISE	4	0	--	03	50	50	100	4
4	PC	18IS43	Object Oriented Concepts	ISE	4	0	--	03	50	50	100	4
5	PC	18IS44	Python Programming	ISE	3	0	--	03	50	50	100	3
6	PC	18IS45	Computer Networks	ISE	3	2	--	03	50	50	100	4
7	PC	18ISL46	Object Oriented Concepts Lab	ISE	--		2	03	50	50	100	1
8	PC	18ISL47	Design and Analysis of Algorithm Lab	ISE	--		2	03	50	50	100	1
9	HS	18HS41/42	Constitution of India Professional Ethics and Human Rights/ Env. Studies	Hum/Civ	1	--	--	02	50	50	100	1
10	MC	18HS43	Employability skills (MC)	Humanities	04		--	03	50	-	50	0
				TOTAL	**24**	**04**	**04**	**29**	**500**	**450**	**950**	**24**
Course prescribed to lateral entry Diploma holders admitted to III semester of Engineering programs												
11	MC	18MAD41	Advance Mathematics - II	Mathematics	02	01	--	03	50		50	0

Note: HODs are informed to accommodate one more laboratory in addition to the above courses if needed, without altering the total number of credits (TOTAL: 24).

(a) The mandatory non – credit courses Advance Mathematics I and II prescribed at III and IV semesters respectively, to lateral entrant Diploma holders admitted to III semester of BE programs shall compulsorily be registered during respective semesters to complete all the formalities of the course and appear for SEE examination.

(b) The mandatory non – credit courses Advance Mathematics I and II, prescribed to lateral entrant Diploma holders admitted to III semester of BE programs, are to be completed to secure eligibility to VII semester. However, they are not considered for vertical progression from II year to III year of the programme but considered as head of passing along with credit courses of the programme to eligibility to VII semester.

Note: BC: Science Course, PC: Professional Core. Hu: Humanities, MC: Mandatory Course.
ENV: Environmental Studies, CIP: Constitution of India Professional Ethics and Human Rights

HEAD DEPT. OF INFORMATION SCIENCE & ENGG

III SEMESTER

For THIRD Semester B E (For CS/IS branch only)

	SUBJECT TITLE: DISCRETE MATHEMATICS & NUMERICAL METHODS		
	Subject Code : 18MA31CS/IS	**Number of Credits: 3 = 2 : 1 : 0 (L : T : P)**	**No of lecture hours per week: 04 (L=2+T=2)**
	Exam Duration: 3 Hrs	**Exam Marks: CIE +Assignment + Group Activity+ SEE = 40 + 5 +5+ 50 = 100**	**Total No. of lecture hours: 39**
	Course objectives: To enhance the student's ability to think logically, mathematically and algorithmically many basic computer related concepts and provide a coherent development in the field of Discrete Structures and Numerical methods.		

Unit No.	**Syllabus Content**	**No. of hours**	
		Theory	**Tutorial**
1	**Logics and Quantifiers:** Basic Connectives and Truth Tables, Logic equivalence – The Laws of Logic, Logical Implication – Rules of Inference.	05	02
2	**Lattices and Boolean Algebra:** Relation and ordering, partially ordered sets, Lattices as poset, properties of lattices, complete lattices, bounds of lattices, distributive lattice and complemented lattices.	06	02
3	**Groups:** Binary algebra, Semigroups and monoids, Groups, Examples and Elementary Properties, Subgroups, Homomorphisms, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem.	06	02
4	**Solutions of algebraic and transcendental equations:** Regula-Falsi method and Newton-Raphson method. **Interpolation:** Interpolation using Newton's forward and backward difference formulae. Interpolation with unequal intervals: Newton's divided difference and Lagrange's formulae.	06	02
5	**Numerical differentiation, integration and solutions of ODE'S:** Numerical differentiation using Newton's forward and backward difference formulae. Numerical Integration: Simpson's $1/3^{rd}$, $3/8^{th}$ rules and Weddle's rule. Solutions of first ODE's: Euler's modified method, Runge-Kutta fourth order method, Milne's and Adams Bashforth methods.	06	02

Course Outcomes: After the successful completion of the course the students are able to:
CO1: Demonstrate understanding of how to read and annotate an outline of a proof and able to write a logical poof of a statement.
CO2: Create rigorous mathematical arguments to logical gates and develop an algorithm.
CO3: Apply algebraic structures in codes in cryptography.
CO4: Compare the viability of different approaches to the numerical solution of problems arising in finding roots of equations, interpolation and approximation, numerical differentiation and integration, and solution of ODE's.
CO5: Develop a variety of numerical algorithms using appropriate technology/programming languages.

Course Outcomes (CO) Mapping with Programme Outcomes (PO)
CO1: PO1, PO2
CO2: PO1, PO2
CO3: PO1, PO2
CO4: PO1, PO2
CO5: PO1, PO2,PO4

TEXTBOOKS:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education, 2004.
2. J. P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science", ,Tata McGraw-Hill
3. B.S. Grewal, Higher Engineering Mathematics (Latest Edition, 2016), Khanna Publishers, New Delhi

REFERENCE BOOKS/Web sources:

1. Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
2. D.S. Malik and M.K.Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004.
3. H.K.Dass and Er. RajnishVerma, Advanced Engineering Mathematics (Latest Edn, 2015), S.Chand Publisher, New Delhi.
4. Dennis G Zill, MihaelGulle, Advanced Engineering Mathematics (2ndEdn), CBS publishers.
5. N.P.Balli and Manish Goyal, A text book of Engineering Mathematics, Lakshmi PublicationsYork.

QUESTION PAPER PATTERN:

The Semester End Examination (SEE) is for 100 marks.

1. There shall be five full questions (one question for each unit) carrying 20 marks each and all are
   Compulsory.
2. There shall be internal choice in all the Units.

Note: Three assignments are evaluated for 5 marks.

Sub Title : COMPUTER ORGANIZATION AND ARCHITECTURE		
Sub Code: 18IS31	**No. of Credits:4=4: 0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 4**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

1. Presents the basic structure and operation of a digital computer.
2. To understand the basics of assembly language.
3. Implement assembly programs that accomplish basic computational and I/O operations.
4. Acquire knowledge of semiconductor memories, cache memory and virtual memory.
5. Analyze the concepts of CPU Bus structures and Pipelining.

UNIT No	Syllabus Content	No of Hours
1	**Basic Structure of Computers:** Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Pipelining and Superscalar Operation,Clock Rate, Instruction Set:CISC and RISC, Performance Measurement. Numbers, Arithmetic Operations and Characters, Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language. **T 1:Ch 1-1.1,1.2,1.3,1.4,1.6(1.6.1,1.6.2,1.6.3,1.6.4,1.6.5,1.6.7)** **Ch 2-2.1,2.2,2.3,2.4,2.5,2.6**	**12**
2	I**nput / 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. **T1: Ch – 4.1, 4.2 (4.2.1 to 4.2.5), 4.4, 4.5, 4.6, 4.7**	**10**
3	**Memory System:** Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage. **T1: Ch 5 – 5.1 to 5.7, 5.9**.	**10**
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,	**10**

	Integer Division, Floating-point Numbers and Operations. **T1 : Ch 6 – 6.1 to 6.7**	
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. **T1: Ch 7, Ch 8 – 8.1, Ch 9 – 9.1, 9.2, 9.3**	**10**

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcomes:
After the completion of the course students will be able to
CO1: Analyze functional units of a computer, its operational concepts, addressing modes, internal organization of a system through practicing with an assembly language
CO2:Analyze and design I/O devices, interrupts and I/O interfaces.
CO3: Analyze and design of memory unit including SRAM, DRAM, cache mapping techniques and basics of virtual memory.
CO4:Design basic processing unit and implement execution of complete instruction.
CO5: Implement basic and intermediate concepts of pipelining.

COs	**Mapping with POs**
CO1	PO1,PO2,PO12
CO2	PO1,PO5,PO12
CO3	PO1,PO2, PO5,PO12
CO4	PO1,PO3,PO5,PO12
CO5	PO1,PO2, PO5,PO12

TEXT BOOKS:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, 5th Edition, Tata McGraw Hill, 2002. (Listed topics only from Chapters 1, 2, 4, 5, 6, 7, 8, 9 and12).

REFERENCE BOOKS / WEBLINKS:

1. William Stallings: Computer Organization & Architecture, March ,2012.
2. NPTEL:http://nptel.ac.in/courses/106106092/
3. http://freevideolectures.com/Course/2277/Computer-Organization#

Sub Title : **DATA STRUCTURES WITH C**		
Sub Code: 18IS32	**No. of Credits:4=4: 0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 4**
Exam Duration : **3 hours**	**Exam Marks: CIE +Assignment + Group Activity + SEE =** **40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

1. To become familiar with the concept of pointers and its usage in dynamic memory allocation.
2. To study and understand the representation and implementation of linear data structures.
3. To classify and comprehend the consequences of using non linear data structures in implementing a system .
4. To identify the suitable data structure during application development
5. To gain knowledge of sorting, searching and hashing techniques .

UNIT No	Syllabus Content	No of Hours
1	**Introduction:** Data Structures, Classifications (Primitive & Non Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, Dynamically allocated arrays. **Array Operations**: Traversing, inserting, deleting, searching, and sorting. Multidimensional Arrays, Polynomials and Sparse Matrices. **Strings:** Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Examples. **T 1: Ch 1: 1.2, Ch 2: 2.2 - 2.7 T 2: Ch 1: 1.1 - 1.4, Ch 3 : 3.1 - 3.3, 3.5, 3.7 , Ch 4: 4.1 - 4.9, 4.14 R3: Ch 1: 1.4**	**12**
2	**Stacks:** Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. **Recursion** - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. **Queues:** Definition, Array Representation, Queue Operations, Circular Queues,Circular queues using Dynamic arrays, Dequeues, Priority Queues, A Mazing Problem. Multiple Stacks and Queues. Programming Examples. **T 1: Ch 3: 3.1 -3.7 T 2: Ch 6: 6.1 -6.3, 6.5, 6.7-6.10, 6.12, 6.13**	**10**

3	**Linked Lists:** Definition, Representation of linked lists in Memory, Memory allocation;Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. **Applications of Linked lists** – Polynomials, Sparse matrix representation. Programming Examples. **T 1: Ch 4: 4.1 – 4.6, 4.8 T 2: Ch 5: 5.1 – 5.10**	**10**
4	**Trees:** Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations,Threaded binary trees, Binary Search Trees – Definition,Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression , Programming Examples, AVL Trees, AVL rotations, overview of Red Black trees and Tournament Trees **T 1: Ch 5.1 –5.5, 5.7 T 2: Ch 7: 7.1 – 7.9**	**10**
5	**Graphs:** Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. Sorting and Searching: Insertion Sort, Radix sort, Address Calculation Sort. **Sorting and Searching:** Insertion Sort, Radix sort, Address Calculation Sort. **Hashing:** Hash Table organizations, Hashing Functions, Static and Dynamic Hashing. **Files and Their Organization:** Data Hierarchy, File Attributes, Text Files and Binary Files, Basic File Operations, File Organizations and Indexing. **T 1: Ch 7, Ch 8: 8.1,Ch 9: 9.1, 9.2, 9.3**	**10**

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcomes:
After the completion of the course students will be able to
CO1: Implement pointers in memory allocation , data structure functions.
CO2: Classify common data structures and implement them.
CO3: Apply appropriate algorithm for problem solving after identifying the appropriate linear data structure.
CO4: Design efficient programs by choosing the most apt non linear data structure.

COs	**Mapping with POs**
CO1	PO3,PO4,PO9,PO10
CO2	PO2,PO3,PO4,PO9,PO12

CO3	PO2,PO3,PO4,PO9,PO12
CO4	PO2,PO3,PO4,PO9,PO12

TEXT BOOKS:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press,2014.

2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

REFERENCE BOOKS / WEBLINKS:

1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014
2. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
3. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013
4. A M Tenenbaum, Data Structures using C, PHI, 1989
5. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

Sub Title : **UNIX AND SHELL PROGRAMMING**		
Sub Code: 18IS33	**No. of Credits:4= 3: 1 : 0 (L-T-P)**	**No. of Lecture Hours/Week :4**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE =** **40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

1. Understand and execute the different types of unix command related to file, protection and security.
2. Develop shell programs using command substitution, positional parameters and control structures .
3. Implementation of SED and AWK commands.
4. Develop simple programs using PERL and AWK scripts.

Unit No.	**Syllabus Content**	**No of Hours**
1	Introduction. The UNIX operating system, UNIX architecture, Features of UNIX, Command usage:- locating commands, internal and external commands, Man Browsing the manual pages ,Understanding the man documentation File system: - the file, what is in a file name?, The parent child relationship ,The HOME variable: the Home directory, Pwd: checking your current directory,Cd: changing the current directory Mkdir : making directories, Rmdir: removing current directories, Absolute pathnames Relative pathnames, Ls :listing directory contents, Unix file system. Basic file attributes: Ls –l: listing file attributes, the –d option: listing directory attributes,File ownership, File permission , Chmode: changing file permission (relative and absolute method), Directory permission, changing file ownership, chown, chgrp. The vi editor: Vi basics, three modes of vi editor ,Input mode –entering and replacing text, Saving text and quitting –the ex mode,Navigation, Editing text,Undoing last editing instructions, Repeating the last command, Searching for a pattern substitution. The shell: The shells interpretive cycle, shell offering, pattern matching. Escaping and quoting: Redirection: the three standard files,/dev/null and dev/tty: two special files,Pipe, Tee: creating a tee, Command substitution, Shell variables	**11**
2	**The process:** Process basics, Ps:process status, System process, mechanism of process creation,Internal and external commands, Process states and Zombies, Running jobs in background,Nice: job executing with low priority, Killing processes with signals, Job control , at and batch: execute later,cron:running jobs periodically,Time: timing processes **Customizing the environment:** The shells, Environmental variables, the common environmental variables,aliases,in-line command editing , The initialization scripts. **More file attributes:** File systems and Inodes, Hard links,Symbolic links and ln,	**11**

	The directory,Umask: Default file and directory permission, Modification and access times,Find: locating files, **Simple filters:** The sample database, Pr:printing files,Head: displaying the beginning of a file, Tail: displaying the end of a file,Cut: slitting a file vertically,Paste : pasting files,Sort: ordering a file, Uniq: locate repeated and non repeated lines, Tr:translating characters.	
3	**Filters using regular expression: Introduction,grep**: searching for a pattern,Basic regular expression(BRE), Extended regular expression(ERE) and egrep,**Sed:** the stream editor, Line addressing, using multiple instructions,**Context addressing**Writing selected lines to a file, Text editing,Substitution, basic regular expression **Essential shell programming:** Shell script, Read: making scripts interactive, Using command line arguments, Exit and exit status of command, Logical operator && 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 parameter,The header document , Trap :interrupting a program, Debugging shell script with set –x, Sample validation and entry scripts.	**10**
4	**Awk**-advanced filter: Simple awk filtering, Splitting a line into fields, Printf: formatting output,Variables and expressions, The comparison operators, Number processing,Variables, The –f option : storing awk programs in a file,The BEGIN and END section, Built in variables,Arrays, functions,Control flow – the if statement,Looping with for, Looping with while	**10**
5	**Perl-the master manipulator:** Perl preliminaries, The chop function, Variables and operators,The string handling functions, Specifying filenames in command line, $-: the default variable,Current line number ($.) and range operator (..), Lists and arrays, Foreach:looping through a list,Split:, join,dec2bin.pl,grep,associative arrays,Regular expressions and substitution,File handling, file tests, subroutines	**10**

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5
Note 3: Group Activity is evaluated for 5 Marks

Course Outcomes:
After the completion of the above course students will be able to

CO1: Develop simple command level codes for file, process, redirection, piping, protection and security
CO2: Demonstrate the usage of shell using shell positional parameters and command substitution.
CO3: Demonstrate different types of SED addressing and AWK filtering.
CO4: Develop PERL programs for string usage, file concept and arrays handling.

COs	**Mapping with POs**
CO1	PO1, PO2,PO3,PO4
CO2	PO1,PO2,PO3,PO4,PO5
CO3	PO1,PO2,PO3,PO4,PO5
CO4	PO1,PO2,PO3,PO4,PO5,PO7

TEXT BOOK:

UNIX –Concepts and Applications, Sumitabha Das, 4 thEdition, Tata Mc GrawHill, 2006.
(Chapters 1.2, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19)

REFERENCE BOOKS:

1. UNIX and Shell Programming, Behrouz A. Forouzan and Richard F. Gilberg, Thomson, 2005.
2. Unix & Shell Programming, M.G. Venkateshmurthy, Pearson Education, 2005.

Sub Title : DIGITAL PRINCIPLES AND LOGIC DESIGN		
Sub Code: 18IS34	**No. of Credits:3=3: 0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 3**
Exam Duration : **3 hours**	**Exam Marks: CIE +Assignment + Group Activity + SEE =** **40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 39**

Course Objectives:

1. To understand how to work with variety of digital logic gates along with their operations using truth table and logic diagram.
2. To understand and apply minimization techniques for designing optimized digital circuits along with HDL implementation.
3. To analyze and design cost effective combinational and sequential circuits for given problems.
4. To analyze and design a synchronous and asynchronous counter.
5. To analyze and design a sequential circuit.

Unit No.	**Syllabus Content**	**No. of Hours**
1	**Digital Principles:** Definitions for Digital Signals, Digital Waveforms, Digital Logic. **Digital Logic**: Overview of basic gates and universal gates, AND-OR-Invert Gates, Positive and Negative Logic. T1:1.1,1.2,1.3,2.1,2.2,2.3,2.4 **Combinational Logic Circuits:** Boolean Laws and Theorems ,Sum-of-Products Method, Truth Table to Kamaugh Map, Pairs, Quads, and Octets , , Karnaugh Simplifications for 4 variables, Don't-care Conditions, Product-of-Sum, Product-of-sums Simplification ,Simplification using Quine McClusky Method. T1: Ch 3: 3.1 to 3.9.	8
	Data-Processing Circuits: Multiplexers, Demultiplexers, 1-of-16 Decoder, Encoders, Magnitude Comparator, HDL Implementation of Data Processing Circuits. T1: Ch 4: 4.1,4.2,4.3,4.6,4.9,4.14	8
3	**Flip-Flops:** Flip-flops: RS FLIP-FLOPs , Gated FLIP-FLOPs Edge-triggered RS FLIP-FLOPs, Edge-triggered D FLIP-FLOPs, Edge-triggered JK FLIP-FLOPs, JK Master-slave FLIP-FLOPs; JK Master-slave FLIP-FLOP, Various Representations of FLIP-FLOPs, Conversion ofFLIP-FLOPs: A Synthesis Example, HDL Implementation of Flip-flops. T1: Ch 8: 8.1 to 8.8, 8.10,8.12 **Registers:** Types of Registers, Applications of Shift Registers, Register Implementation using HDL. T1: Ch 9: 9.1,9.7	8
4	**Counters:** Asynchronous Counters ,Synchronous Counters, Decade Counters ,Counter Design as a Synthesis problem, Counter Design using	8

	HDL.T1:Ch 10: 10.1,10.3,10.5,10.7,10.9	
5	**Design of Synchronous and Asynchronous Sequential Circuits:** Design of Synchronous Sequential Circuit: Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and Circuit Diagram, State Reduction Technique. T1: Ch 11: 11.1 to 11.4,11.7	7

Note 1: All units will have internal choice

Note 2: Three Assignments are evaluated for 5 marks:

Assignment – I from Units 1 and 2.

Assignment – II from Units 3 and 4

Assignment -III from Unit 5

Note 3: Group Activity is evaluated for 5 Marks

Course Outcomes:

After the completion of the course students will be able to

CO1: Comprehend the fundamental concepts and principles of digital design.

CO2: Design and analyze cost effective combinational circuits and apply concept of Minimization of Boolean functions using different methods. Implement HDL programming.

CO3: Design, analyze and implement various data processing circuits and describe behavior of various digital circuits.

CO4: Design and analyze synchronous and asynchronous counters.

CO5: Design and analyze sequential logic circuits using different models.

COs	**Mapping with POs**
CO1	PO1,PO2, PO4,PO5,PO7
CO2	PO1,PO2, PO4,PO5
CO3	PO1, PO2,PO4,PO5,
CO4	PO1,PO2,PO3, PO4,
CO5	PO1,PO2, PO4,PO5, PO7

TEXT BOOK:

Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2011.

REFERENCE BOOKS:

1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition, Tata McGraw Hill, 2005.
2. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Thomson, 2004.
3. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss: Digital Systems Principles and Applications, 10th Edition, Pearson Education, 2007.
4. R D Sudhaker Samuel, *K.S. Nandini Prasad*: Logic Design, 1st edition, Elsevier Publication, 2013.

Sub Title : SOFTWARE ENGINEERING		
SubCode:18IS35	**No. of Credits:3=3 : 0 : 0 (L-T-P)**	**No. of Lecture Hours/Week: 3**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 39**

Course objectives:

1. Knowledge of basic SW engineering methods and practices, and their appropriate application.
2. Understanding of software requirements and the SRS documents.
3. Describe System model and Object oriented concepts.
4. Understanding of software evolution and related issues of Design Patterns.
5. Understanding of approaches to verification and validation including static analysis, project management, and ensure good quality software.

Unit No.	**Syllabus**	**No of Hours**
1	**Overview:** FAQ's about software engineering, Professional and ethical responsibility.**Socio-Technical systems:** Emergent system properties; Systems engineering.**Software Processes:** Process activities; The Rational Unified Process; Agile methods, Plan-driven and agile development, XP, Scrum, Computer Aided Software Engineering. T1: Ch 1, Ch 2, Ch 3	**8**
2	**Requirements:** Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; The software requirements document. **Requirements Engineering Processes:** Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management. T1: Ch 4	**8**
3	**System modeling:** System Models: Context models; Behavioral models; Object models; Structured methods.**Software Design:** Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles.**Object-Oriented design:** Objects and Object Classes; An Object-Oriented design process; Object-oriented design using the UML. T1: Ch 5 ,Ch 6	**8**
4	Design patterns ,Implementation issues, Design evolution.**Development:** Rapid Software Development: Rapid application development. Open source development. **Software Evolution:** Program evolution dynamics; Software maintenance. T1: Ch 7 , Ch 9	**7**
5	**Verification and Validation:** Verification and Validation: Planning;	**8**

	Software inspections; Automated static analysis; Verification and formal methods. **Software Management**: Project Management; Risk management.**Project planning:** software pricing; Project scheduling; Agile Planning; Estimation techniques. T1: Ch 22, Ch 23,Ch 24	

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5
Note 3: Group Activity is evaluated for 5 Marks

Course Outcomes:

After completion of course students will be able to :

CO1: Assess professional and ethical responsibility of a software engineer.

CO2: Design and develop software system, component, or process to meet desired needs within realistic constraints

CO3: Identify and develop system models to design the software system.

CO4: Recognize and apply the techniques, modern engineering tools necessary for engineering practice

CO5: Demonstrate the knowledge of verification and validation to ensure good quality software

Cos	**Mapping with POs**
CO1	PO1,PO6,PO8,PO12
CO2	PO1,PO2,PO3,PO12
CO3	PO1,PO2,PO3,PO5
CO4	PO1,PO2,PO3,PO10
CO5	PO1,,PO6,PO10,PO11,PO12

Text Books:

1.Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012. (Listed topics only from Chapters 1,2,3,4, 5,6, 7, 8, 9,10,11, 22, 23 and 24)

Reference Books:

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India
3. http://agilemanifesto.org/
4. http://www.jamesshore.com/Agile-Book

Sub Title : DATA STRUCTURES WITH C LAB		
Sub Code:: 18ISL36	**No. of Credits:1= 0:0:1 (L-T-P)**	**No. of lecture hours/week : 2**
Exam Duration : 3 hours	**Exam Marks: CIE + SEE = 50 + 50 = 100**	

Course Objectives:

1. To understand design and implement the concept of stack using recursive techniques.
2. To Implement the application of stacks in converting an expression from infix to postfix notation and evaluate postfix expressions.
3. Design common data structures and implement linear queue, circular queue, priority queue
4. To understand the importance of implementing data structures like stacks using linked list, queues using linked list, doubly linked lists and circular linked list.
5. To traverse a non linear data structure like a Binary Search Tree.

I. LIST OF PROGRAMS

1 Design develop and implement menu driven C program to perform following set of operations on Stack of integers (using array of maximum size MAX)

i)) Push   ii) Pop   iii) Display iv) Exit
The program should print appropriate messages for stack overflow, stack underflow, and stack empty.

2 Design , develop and implement a program in C to convert and print a given valid parenthesized or parenthesize free infix expression to postfix expression. The expression consists of single character operands and the binary operators + (plus), - (minus), * (multiply) , / (divide), % (mod) and ^ (power) .

3 Design , develop and implement a program in C to evaluate a valid suffix/postfix expression using stack. Assume that the suffix/postfix expression is read as a single line consisting of positive single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide), % (mod) and ^ (power) .

4 Design develop and implement menu driven C program to perform following set of operations on queue of integers using an array.
i) Insert   ii) Delete   iii) Display   iv) Exit
The program should print appropriate messages for queue overflow, queue underflow, and queue empty.

5 Design develop and implement menu driven C program to perform following set of operations on circular queue of integers using an array.
i) Insert   ii) Delete   iii) Display   iv) Exit
The program should print appropriate messages for circular queue overflow, circular queue

underflow, and circular queue empty.

6 Design, Develop and Implement a menu driven program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: *USN, Name, Branch, Sem, PhNo*

i.Create a SLL of N Students Data by using *front insertion*.

i.Display the status of SLL and count the number of nodes in it

i.Perform Insertion at End of SLL

v.Perform Deletion at End of SLL

v.Exit

The program should print appropriate messages for dynamic stack overflow, underflow and empty.

7 Design, Develop and Implement a menu driven program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: *USN, Name, Branch, Sem, PhNo*

i.Create a SLL of N Students Data by using *front insertion*.

i.Display the status of SLL and count the number of nodes in it

i.Perform Insertion at End of SLL

v.Perform Deletion at front end of SLL

v.Exit

The program should print appropriate messages for dynamic queue overflow, underflow and empty

8 Design, Develop and Implement a menu driven Program in C for the following operations on

Doubly Linked List (DLL) of Employee Data with the fields: *SSN, Name, Dept, Designation,*

Sal, PhNo

i.Create a DLL of N Employees Data by using *end insertion*.

i.Display the status of DLL and count the number of nodes in it

i.Perform Insertion and Deletion at End of DLL

v.Perform Insertion and Deletion at Front of DLL

v.Exit

9 Design, Develop and Implement a menu driven Program in C for the following operations on

Binary Search Tree (BST) of Integers .

a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2

b. Traverse the BST in Inorder, Preorder and Post Order

c. Search the BST for a given element (KEY) and report the appropriate message

d. Exit

10 Design, Develop and Implement a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes.

Represent and Evaluate a Polynomial:
$P(x,y,z) = 6 x^{2} y^{2} z - 4 y z^{5} + 3 x^{3} y z + 2 x y^{5} z - 2 x y z^{3}$

Note: Programs 2, 3, 6, 8, 9, 10 to be conducted with support of Virtual Lab .

Weblink:

https://cse01-iiith.vlabs.ac.in /
https://ds1-iiith.vlabs.ac.in/data-structures-1/

II. OPEN ENDED QUESTIONS

Design and implement a solution to the following in C.

1. Design, Develop and Implement a menu driven Program in C for the following array operations.

i. Creating an array of N Integer Elements
ii. Display of array Elements with Suitable Headings
iii. Inserting an Element (ELEM) at a given valid Position (POS)
iv. Deleting an Element at a given valid Position (POS)
v. Exit.

2. Design, Develop and Implement a Program in C for the following operations on Strings.

i. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP)
ii. Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in
iii. STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR

3. Design, Develop and Implement a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes

i. Represent a Polynomial P(x,y,z)
ii. Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z)
iii. Display the polynomial P (x,y,z)

NOTE:

1. Student is permitted to submit open ended solution to any other open ended question apart from the list above . But it has to be approved by the staff in charge.
2. In the examination each student picks one question from a lot of all 10 questions

Course Outcomes:
After the completion of the above course students will be able to
CO1: Design and develop stack, an application providing solution to convert infix to postfix expression using stack and also design a solution to evaluate postfix expression.
CO2: Implement queues like linear queue, circular queue .
CO3: Design and develop solution to implement the following : singly linked list, stacks using linked list, queues using linked list, doubly linked list and circular linked list.
CO4: Design the solution to traverse a Non linear data structure like a Binary Search Tree.

COs	**Mapping with POs**
CO1	PO2,PO3,PO4,PO7,PO9
CO2	PO3,PO4,PO7,PO9
CO3	PO2,PO3,PO4,PO7,PO9
CO4	PO3,PO4,PO7,PO9

Sub Title : DIGITAL PRINCIPLES AND LOGIC DESIGN LAB		
Sub Code:18ISL37	**No. of Credits:1=0 : 0 : 1 (L-T-P)**	**No. of lecture hours/week : 2**
Exam Duration : 3 hours	**Exam Marks: CIE + SEE = 50 + 50 = 100**	

Course Objectives:

1. Acquire basic skills and confidence to design, analyze, and implement circuits involving various digital logic gates.
2. To design, implement and analyze combinational logic circuits.
3. To design, implement and analyze sequential logic circuits.
4. Realization of one, two bit magnitude comparator.
5. Simulation and analysis of various logic circuits using VERILOG/VHDL.

I. LIST OF EXPERIMENTS

PART –A

1. Realize full adder using 3-to-8 decoder IC and 4 input NAND gates.
2. Given any 4-variable logic expression simplify using multiplexer IC.
3. Design and implement mod-n (n<8) synchronous up counter using J-K Flip-Flop ICs
4. Design and implement a ring counter using 4-bit shift register.
5. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9). Display the count value on 7 segment LED display using BCD to 7 segment code converters IC.

PART- B

1. Write VHDL/Verilog code to realize all the logic gates.
2. Given a Boolean expression , simplify it using K-Map .Write Verilog/VHDL code to realize simplified boolean expression.
3. Write the Verilog/VHDL code for a full adder. Simulate and verify it’s working.
4. Write VHDL code for full subtractor. Simulate and verify its working.
5. Write the Verilog /VHDL code for an 8:1 multiplexer. Simulate and verify its working.
6. Write Verilog /VHDL code for two bit magnitude comparator.
7. Write the Verilog/VHDL code for D Flip-Flop with positive-edge triggering. Simulate and verify it’s working.
8. Write the Verilog/VHDL code for JK flip flop with negative triggering .Simulate and verify it’s working.
9. Write the Verilog/VHDL code for switched tail counter. Simulate and verify it’s working.
10. Write the Verilog/VHDL code for mod-8 up counter. Simulate and verify it’s working.

II. OPEN ENDED QUESTIONs

Students can build their own applications using the digital components either hardware or simulation tools (software tools)

Course Outcomes:

After completion of the course the students will be able to:

CO1: Simplify Boolean expressions and implement optimal Logic circuits.

CO2: Design and realize combinational circuits.

CO3: Design and realize sequential circuits used for variety of applications.

CO4: Apply minimization techniques to design and implement optimized digital circuits.

CO5: Develop HDL programs for combinational and sequential circuits.

COs	**Mapping with POs**
CO1	PO1,PO2, PO4,PO5 ,PO7
CO2	PO1,PO2, PO4,PO5
CO3	PO1, PO2,PO4,PO5
CO4	PO1,PO2,PO3, PO4
CO5	PO1,PO2, PO4,PO5 , PO7

NOTE:

1. STUDENT IS PERMITED TO SUBMIT OPEN ENDED SOLUTION TO ANY OTHER OPEN ENDED QUESTION APART FROM THE LIST ABOVE . BUT IT HAS TO BE APPROVED BY THE STAFF IN CHARGE.

2. IN THE EXAMINATION EACH STUDENT PICKS ONE QUESTION FROM PART A AND ONE QUESTION FROM PART B

	CONSTITUTION OF INDIA & PROFESSIONAL ETHICS		
	Subject Code: 18HS31	No. of Credits: 1	No of lecture hours per week: 2 Hrs
	Exam Duration: 2 Hrs	Exam Marks: 50	Total No. of lecture hours: 16 hrs per week

Total Text Objectives:

Constitution of India and professional Ethics taught to the students to enable them an insight into the constitution and enlighten them about the fundamental rights, Protection of rights through enforcement of writs. Functions of judiciary, knowledge and importance of election in democratic country procedure of elections to the post of president, members of parliament and state legislative assembly and the function of Parliament, legislative Assembly and their powers. Professional ethics- The duties and basic responsibilities of engineers towards society and their employer, apply of new ideas and protection of those ideas.

Course Objective

1. The objective of part-I of the syllabus is to know how the Constitution of India was framed and the basic structure of Indian Constitution and the rights given by the Constitution to their Citizens.

2. The objective of part-II.. To inculcate knowledge on election commission in the democratic system. The functions of judiciary in upholding of the Indian Constitution

3. The objective of part-III. Is enlightening them about the system of government adopted in both Central and State and Method of election of representative of Parliament and State Assembly their powers and function and also enlighten them about the term democracy

4. The Objective of part IV. To create awareness on Professional ethics and Human Values

5. The objective of part-V. To inculcate knowledge and exposure on Safety and Risk, Risk Benefit Analysis and have an idea about the Collective Bargaining, Confidentiality, Professional, Employee, Intellectual Property Rights

UNIT NO	SYLLABUS CONTENTS	NO of hours
1	INTRODUCTION Framing of India Constitution, Salient Features of the Constitution, Basic Structure, Preamble of the Constitution, Fundamental Rights- Article(12-35),Art 32 &226, Restrictions under Constitution Fundamental Duties Art (51A).	8
2	Directive Principles Of State Policy,Election commission, Judiciary system-Role of Supreme Court of India and High Court of State	3
3	UNION GOVERNMENT:Executive - President, Prime Minister, council of ministers,Legislature- Parliament, Lok-Sabha , Rajya-Sabha STATE GOVERNMENT:Executive – Governor, Chief Minister, Council of Ministers.Legislative- Legislative Assembly , Legislative Council	5
	HUMAN VALUES& PROFESSIONAL ETHICS:	5

4	Morals, values and Ethics – Integrity – Work ethic – Service learning – Civic virtue – Respect for others – Living peacefully – Caring – Sharing – Honesty – Courage – Valuing time – Cooperation – Commitment – Empathy – Self-confidence.	
5	SAFETY, RESPONSIBILITIES AND RIGHTS : Safety and Risk – Assessment of Safety and Risk – Risk Benefit Analysis and Reducing Risk – Respect for Authority – Collective Bargaining – Confidentiality – Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) Piracy and plagiarism.	4

Course Outcomes

1. Students come to know that who are all take part in the framing of Indian Constitution and how it was framed, what it contained. The Rights they can enjoy as a citizen of India. Case law gives detailed knowledge to the students about their fundamental rights. Procedure of enforcement of fundamental rights.

2. . Student gets the knowledge about procedure of conducting of election by the election commission, its duties and powers along with powers and functions of judiciary.

3. Student comes to know regarding system of parliamentary form of government .and how the representative of Central and State Government are elected their powers and functions

4. Student comes to know the correct meaning of ethics and their ethical duties and responsibilities and using of honesty in their profession, and the decision making ability.

5. Student gets the knowledge of using the intellectual propert rights and its protection and its application in their profession.

TEXT BOOKS:

1. Introduction To The Constitution Of India By Durga Das Basu. (Students Edition) Prentice – Hall Eee, 19th/20th Edn., 2001.

2. Engineering Ethics By Charles E. Haries, Michael.S.Pritchard And Michael J. Robins Thompson Asia, 2003-08-05

REFERENCE BOOKS:

1 An Introduction to Constitution of India by M.V.Pylee, Vikas Publishing, 2002.

2 Constitution of India by B S Raman

3 Engineering Ethics by M.Govindarajan, S. Natarajan, V.S.Sendilkumar, Prentice – Hall of India Private Ltd, New Delhi, 2004.

4 Constitution of India and Professional Ethics—K R Phaneesh

5 Introduction to the Constitution of India—Brij Kishore Sharma

	SOFT SKILLS		
	Subject Code: 18HS33	**Mandatory Course**	**No of lecture hours per week: 2 Hrs**
	Exam Duration:	**Exam marks: NIL**	**Total No. of lecture hours: 26 hrs per semester**

COURSE OBJECTIVES:

1. The lessons under unit 1 is to help students to recognize oneself as an individual being aware of one's knowledge, personality, environment and lifestyle, through different base and measures like SWOC analysis, personal developments.

2. The lessons under this unit make students to understand the importance of soft skills and hard skills and how to think critically and also learn the importance of creative thinking.

3. The lessons under this unit make students to understand the importance of attitude, time, and also learn to manage them.

4. The objective of this unit is to help a student to learn the art of goal setting and being motivated in achieving his goal in spite of all the hardships with different problems along with teamwork, and leadership qualities.

5. The unit 5 aims at teaching the students about importance of handwriting, punctuation marks, spellings, which helps them, improve in writing skills.

UNIT NO	SYLLABUS CONTENT	HRS /COS
1	a. **SWOC ANALYSIS** b. **SELF – AWARENESS AND PERSONAL DEVELOPMENT** c. **ETTIQUTTE AND MANNERS**	**5** **CO1**
2	d. **INTRODUCTION TO SOFT SKILLS AND HARD SKILLS** e. **CREATIVITY** f. **CRITICAL THINKING**	**5** **CO2**
3	g. **ATTITUDE** h. **ADAPTABILITY** i. **TIME MANAGEMENT/ PRIORITY** j. **STRESS MANAGEMENT**	**5** **CO3**
4	k. **GOAL SETTING** l. **MOTIVATION** m. **TEAM WORK** n. **PROBLEM SOLVING** o. **LEADERSHIP**	**5** **CO4**

5	p. **GUIDELINES FOR HANDWRITING** q. **PUNCTUATION MARKS** r. **SPELLING** s. **VISUAL NARRATIVES(Ramayana, Mahabharata, Short Stories, films and advertisements)**	**6** **CO5**

COURSE OUTCOME:

1. After the completion of this unit students will know about their self assessments and personality that how to handle various situations in a positive way.

2. At the end of this unit student will be able to see how important it is to consider things carefully and from different angles, something one sees, hears, experiences or reads in order to understand it fully. The student will also have understood the importance of soft skills and thinking creatively.

3. After the completion of this unit students will have learnt about attitude and adaptability and also how to deal with problems and stress in the present world.

4. At the end of this unit students will have learnt some simple ways of planning to achieve his dream and also feel enthusiastic about doing something individually and as well in team.

5. After the completion of this unit students writing skills will have improved.

REFERENCE:

● English for Job Seekers (Language and Soft Skills for the Aspiring) by Geetha Rajeevan, C.L.N. Prakash at al) Cambridge University Press pvt,Ltd.

● New International Business English by Leo Jones and Richard Alexander. Cambridge University Press pvt,Ltd

● Business Benchmark by Norman Whitby. Cambridge University Press pvt,Ltd

● Grammar practice Activities (practical guide for teachers) Cambridge University Press pvt,Ltd

IV SEM

For FOURTH Semester B E (For CS/IS branch only)

	SUBJECT TITLE: PROBABILITY, STATISTICS & QUEUEING THEORY		
	Subject Code : 18MA41CS/IS	**Number of Credits: 3 = 2 : 1 : 0 (L : T : P)**	**No of lecture hours per week: 04 (L=2+T=2)**
	Exam Duration: 3 Hrs	**Exam Marks: CIE +Assignment + Group Activity + SEE =** **40 + 5 +5+ 50 =100**	**Total No. of lecture hours: 39**

Course objectives: To develop analytical capability and to impart knowledge in Statistical methods and Queuing theory and their applications in Engineering and Technology, so as to enable them to apply the same for solving real world problems.

Unit No.	**Syllabus Content**	**No. of hours**	
		Theory	**Tutorial**
1	**Probability distributions:** Recap of Random Variables.Discrete probability distributions- Binomial, Poisson and Geometric distributions; Continuous probability distributions-Exponential, Normal and Weibull distributions.	06	02
2	**Two dimensional Random variables:** Joint probability mass function, Marginal probability function, conditional probability function, Joint density function, marginal density function, conditional probability density function, covariance, correlation coefficient.	06	02
3	**Statistical techniques:** Curve fitting by method of least squares: $y = ax+b$, $y = ax^2+bx+c$ and $y= ab^x$, Correlation– Karl Pearson's coefficient of correlation, Regression analysis – lines of regression (without proof)- problems.	05	02
4	**Random Process:** Classification of random process, description of random process, stationary random process – first order, second order and Strict-sense stationary processes, Autocorrelation and Cross-correlation functions, Ergodic process.	06	02
5	**Queuing Theory:** Basic characteristics of Queuing models-Transient and steady states, Kendall's notation of a Queuing system, Steady state probabilities for Poisson Queue systems, Markov process, Poisson process, birth and death process, Queuing models: Model I- $M/M/1/\infty$/FIFO and Model II- $M/M/s/\infty$/FIFO.	06	02

Course Outcomes: After the successful completion of the course the students are able to:

CO1: Understand of basic rules of random variables and moments of random variables.
CO2: Create probability functions of transformation of random variables and use these techniques to generate data from various distributions.
CO3: Develop probabilities in joint probability distributions and derive the marginal and conditional distributions of bivariate random variables.
CO4: Apply the concepts of probability theory to discrete time Markov chain and establish the Markovian queuing models.
CO5: Implement a variety of statistical techniques to solve problems of industry standard statistical software.

Course Outcomes (CO) Mapping with Programme Outcomes (PO)
CO1: PO1, PO2
CO2: PO1, PO2
CO3: PO1, PO2
CO4: PO1, PO4
CO5: PO1, PO2

TEXTBOOKS:

1. Kishore S. Trivedi, Probabilty and Statistics with Reliability, Queuing and Computer Science.
2. S D Sharma, Operation research, Tata Mc-Grill.
3. Sundaran Pillai, Probabililty, Statistics and Queuing theory PHI.

REFERENCE BOOKS/Web sources:

1. S.C.Gupta and B.K.Kapur, Fundamentals of Mathematical Statistics.
2. Robert B Cooper, Introduction to queuing theory, 2nd Edition, North Holland.
3. Ivo Adan and Jacques Resing, Queueing Systems, Lecture notes, Netherlands (2015).
4. Arnold O. Allen, Probability, Statistics and Queing theory with computer Science Applications, Academic Press, INC. New York.

QUESTION PAPER PATTERN:

The Semester End Examination (SEE) is for 100 marks.

1. There shall be five full questions (one question for each unit) carrying 20 marks each and all are Compulsory.
2. There shall be internal choice in all the Units

Note: Three assignments are evaluated for 5 marks.

Sub Title : MICROCONTROLLER AND EMBEDDED SYSTEMS		
Sub Code: 18IS41	**No. of Credits:3=3 : 0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 3**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 39**

Course Objectives:

1. Differentiate between microprocessors and microcontrollers.
2. Explain the architecture of ARM processor with its instruction set.
3. Identify the applicability of the embedded system
4. Comprehend the real time operating system used for the embedded system

Unit No..	Syllabus Content	No of Hours
1	**Microprocessors versus Microcontrollers, ARM Embedded Systems**: The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software, ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table , Core Extensions T1:Ch 1 - 1.1 to 1.4, Ch 2 - 2.1 to 2.5	7
2	**Introduction to Arm Instruction Set:** Data Processing Instructions, Branch Instructions, Load-Store Instructions, Software Interrupt Instruction, Program Status Register Instructions, Loading Constants,ARMv5E Extensions, Conditional Execution T1: Ch 3-3.1-3.8	8
3	**Embedded System Components:** Embedded Vs General computing system, Classification of Embedded systems, Major applications and purpose of ES. Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (onboard and external types), Embedded firmware, Other system components. **T2: All the Topics from Chapter1 and Chapter2**	8
4	**Embedded System Design Concepts:** Characteristics and Quality Attributes of Embedded Systems, Operational and non-operational quality attributes, Embedded Systems-Application and Domain specific, Hardware Software Co-Design and Program Modeling, embedded firmware design	**8**

	and development T2: Chapter-3, Chapter-4, Chapter-7 (Sections 7.1, 7.2 only), Chapter-9 (Sections 9.1, 9.2, 9.3.1, 9.3.2 only)	
5	**RTOS and IDE for Embedded System Design:** Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread preemption, Preemptive Task scheduling techniques, Task Communication, Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS, Integration and testing of Embedded hardware and firmware, Embedded system Development Environment – Block diagram (excluding Keil), Disassembler/decompiler, simulator, emulator and debugging techniques T2: Chapter-10 (Sections 10.1, 10.2, 10.3, 10.5.2 , 10.7, 10.8.1.1, 10.8.1.2, 10.8.2.2, 10.10 only), Chapter 12, Chapter-13 (block diagram before 13.1, 13.3, 13.4, 13.5, 13.6 only)	8

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5
Note 3: Group Activity is evaluated for 5 Marks

Course Outcomes:

After the completion of the above course students will be able to

CO1: Apply the knowledge gained for Programming ARM for different applications.
CO2: Interface external devices and I/O with ARM microcontroller.
CO3: Interpret the basic hardware components and their selection method based on the characteristics and attributes of an embedded system.
CO4: Develop the hardware /software co-design and firmware design approaches.
CO5: Demonstrate the need of real time operating system for embedded system applications

Cos	**Mapping with POs**
CO1	PO1,PO2,PO3,PO5,PO11
CO2	PO1,PO2,PO3,PO5,PO9,PO12
CO3	PO2,PO4,PO5,PO6,PO12
CO5	PO1,PO2,PO5,PO6,PO12

TEXT BOOKS:

1. Andrew N Sloss, Dominic Symes and Chris Wright, ARM system developers guide, Elsevier, Morgan Kaufman publishers, 2008.

2. Shibu K V, “Introduction to Embedded Systems”, Tata McGraw Hill Education, Private Limited, 2nd Edition.

REFERENCE BOOKS:

1. The Insider’s Guide to the ARM7 Based Microcontrollers, Hitex Ltd.,1st edition, 2005
2. Steve Furber, ARM System-on-Chip Architecture, Second Edition, Pearson, 2015
3. Raj Kamal, Embedded System, Tata McGraw-Hill Publishers, 2nd Edition, 2008
4. Ragunandan, An Introduction to ARM System Design, Cengage Publication

Sub Title : DESIGN AND ANALYSIS OF ALGORITHMS		
Sub Code:18IS42	**No. of Credits:4= 4 :0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 4**
Exam Duration : **3 hours**	**Exam Marks: CIE +Assignment + Group Activity + SEE =** **40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

1. To introduce the concept of an algorithm and understand the techniques for its analysis.
2. To represent the algorithmic time efficiency using different asymptotic notations.
3. Explore the various algorithm design techniques, the process of its design and analysis.
4. To solve problems using appropriate design techniques.
5. Understand concepts of space-time tradeoffs.

UNIT No	**Syllabus Content**	**No of Hours**
1	**Introduction:** What is an Algorithm? Algorithm specification ,Analysis framework Performance analysis : Space complexity , Time complexity Asymptotic Notations and Basic efficiency classes: Informal introduction , O- notation, Ω-notation , Ɵ- notation,Basic efficiency classes, Important problem types: Sorting searching string processing, graph problems, combinatorial problems , Mathematical Analysis of Non-Recursive and Recursive Algorithms **Brute Force** : Introduction, Bubble Sort, sequential search T2: Ch 1 : 1.1,1.2, 1.3 ; T1: Ch 2 : 2.1, 2.2,2.3,2.4 T1: Ch 3: 3.1,3.2	**10**
2	**Divide and conquer:** General Method, Binary search , Recurrence equation for DAC, Finding Minimum and maximum Merge Sort, Quick Sort **Decrease-and-conquer :** Introduction, Depth First Search, Breadth First Search, Topological Sorting. T2: Ch 3: 3.1,3.2,3.3,3.4,3.5; T1: Ch 5: 5.2, 5.3	**11**
3	**Greedy method:** The General Method, Knapsack Problem, Job Sequencing with Deadlines) Minimum cost spanning trees : Prim's Algorithm, Kruskal's Algorithm, Single Source Shortest Paths: Dijikstras Algorithms, Huffman trees **Transform and Conquer: Heaps and** Heap sort T2: Ch 4: 4.1,4.2,4.4**; T1: Ch 9 : 9.1,9.2,9.3,9.4** T1: Ch 6 : 6.4	**10**
4	**Dynamic Programming: computing binomial coefficient,** Warshall's and Floyds algorithms, Knapsack problem Travelling Salesperson problem **Backtracking**: N-Queens problem, sum of Subset Problem T1: Ch 8 : 8.1,8.2,8.4 T2: Ch 5.9 T1: Ch 12: 12.1	**11**

5	**Branch-and-Bound:** Assignment Problem,Knapsack problem, Traveling Salesman n Problem **Space and Time Tradeoffs**: Sorting by Counting, Horspool's algorithm NP-Complete and NP Hard problems: Basic concepts , non deterministic algorithms , P,NP, NP-Complete and NP-Hard classes T1:Ch 11: 11.2 T1:Ch 7.1,7.2 T2: Ch 11: 11.1.	**10**

Note 1: All units will have internal choice

Note 2: Three Assignments are evaluated for 5 marks:

Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcomes:

After the completion of the above course students will be able to

CO1: Determine time efficiency of recursive and non- recursive algorithms.

CO2: Apply, analyze algorithms and solve problems using various algorithm design techniques.

CO3: Design and analyze algorithms to solve the optimization problems.

CO4: Design and analyze algorithms associated with space–time tradeoffs .

COs	Mapping with POs
CO1	PO1,PO2,PO9
CO2	PO2,PO3,PO4,PO9
CO3	PO2,PO3,PO4,PO9
CO4	PO2,PO3,PO4,PO9

TEXT BOOKS:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Second Edition, Pearson Education, 2009.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran: Computer Algorithms/ C++, 2nd Edition, University press, 2014

REFERENCE BOOKS / WEBLINKS:

1. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein: Introduction to Algorithms, 2nd Edition, PHI, 2006.
2. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education)

Sub Title : OBJECT ORIENTED CONCEPTS		
Sub Code: 18IS43	**No. of Credits:4=4: 0 : 0 (L-T-P)**	**No.of Lecture Hours/Week: 4**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

1. To understand the object oriented concepts.
2. To understand the concepts of java.
3. To understand the concept of inheritance and exception handling.
4. To understand the concept of event handling and threads.
5. To design and write a applet and swing programs.

UNIT No	Syllabus Content	No of Hours
1	**Introduction to Java**: Java and Java applications; Java Development Kit (JDK); Java is interpreted, Byte Code, JVM; Object-oriented programming; Simple Java programs. Data types and other tokens: Boolean variables, int, long, char, operators, arrays, white spaces, literals, assigning values; Creating and destroying objects; Access specifiers.Operators and Expressions: Arithmetic Operators, Bitwise operators, Relational operators, The Assignment Operator, The.? Operator; Operator Precedence; Logical expression; Type casting; Strings Control Statements: Selection statements, iteration statements, Jump Statements.**T1:Ch1,2,3,4,5**	11
2	**Classes, Inheritance, Exceptions:** Classes: Classes fundamentals; Declaring objects; Constructors, this keyword, garbage collection. Inheritance: inheritance basics, using super, creating multi-level hierarchy, method overriding. Exception handling: Exception handling in Java. **Text book 2: T1 :Ch 6 , Ch 7, Ch 8 , Ch10**	10
3	**Packages and Interfaces, Multi-Threaded Programming:** Packages, Access Protection, Importing Packages, Interfaces. Multi Threaded Programming: What are threads? How to make the classes threadable ; Extending threads; Implementing runnable; Synchronization; Changing state of the thread;read-write problem, producer consumer problems.**T1 : Ch 9 , Ch 11**	10

4	**Event Handling:** Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model**;** Adapter classes; Inner classes**.** **Introducing the AWT: Working with Windows, Graphics, and Text:** Introduction the AWT: Working with Windows, Graphics and Text AWT Classes, Window Fundamentals, Working with Frame Windows, Introducing Graphics, Working with Color **T1:Ch 22 , Ch 23**	10
5	**Swings:** Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing Applet; Jlabel and ImageIcon; JTextField;The Swing Buttons; JTabbedpane; JScrollPane; JList; JComboBox; JTable. **T1: Ch 29, Ch 30**	11

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcomes:

After the completion of the course students will be able to

CO1: Develop JAVA programs using OOPs principles.

CO2: Develop computer programs to solve real world problems in Java.

CO3: Develop simple GUI interfaces for a computer program to interact with users, and to comprehend the event-based GUI handling principles using Applets and swings.

CO4: Develop the procedure to store and retrieve data using AWT

CO5:Build the simple swings module using Jlist, Jcombobox as GUI

COs	**Mapping with POs**
CO1	PO1,PO2,PO3,PO4,PO5
CO2	PO1, PO5,PO6, PO9,PO10,
CO3	PO1, PO2,PO2,PO5,PO11
CO4	PO1,PO2,PO3, PO4,PO5
CO5	PO3,PO4,,PO5, PO9,PO11

TEXT BOOK:

1. Herbert Schildt: Java - The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.

(Chapters 1, 2, 3, 4, 5, 6, 8, 10, 11, 21, 22, 29, 30, 31)

REFERENCE BOOKS:

1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806

2. Herbert Schildt, The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2003.

3. Stanley B.Lippmann, JoseeLajore, C++ Primer, 4th Edition, Pearson Education, 2005.

4. RajkumarBuyya,SThamarasiselvi, xingchenchu, Object oriented Programming with java, Tata McGraw Hill education private limited.

5. Richard A Johnson, Introduction to Java Programming and OOAD, CENGAGE Learning.E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies

Sub Title : PYTHON PROGRAMMING		
Sub Code: 18IS44	**No. of Credits:3=3 : 0 : 0 (L-T-P)**	**No. of lecture hours/week : 3**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 39**

Course Objectives:

1. Understanding the syntax and semantics of the Python language.
2. To create Functions in Python.
3. To handle Files & Regular expressions in Python.
4. To apply Object Oriented Programming concepts in Python.

5 To create Threaded and Networking applications in Python .

UNIT No	Syllabus Content	No of Hours
1	Introduction to Python, Writing Our First Python Program, Datatypes in Python, Operators in Python, Input and Output, Control Statements T1: **Ch 1, Ch 2, Ch 3, Ch 4,Ch 5 , Ch 6**	**7**
2	Arrays in Python, Strings and Characters, Functions, Lists and Tuples, Dictionaries T1: **Ch 7,Ch 8,Ch 9, Ch 10, Ch 11**	**7**
3	Introduction to OOPS, Classes and Objects, Inheritance and Polymorphism, Exceptions T1:C**h12,Ch13,Ch14,Ch16**	**7**
4	Files in Python, Regular Expressions in Python, Data Structures in Python, Date and Time T1: C**h 17, Ch 18 ,Ch 19, Ch 20**	**9**
5	Threads, Graphical User Interface, Networking in Python, Python's Database Connectivity T1: C**h 20 ,Ch 21, Ch 22, Ch 23**	**9**

Note 1: All units will have internal choice

Note 2: Three Assignments are evaluated for 5 marks:

Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcomes:

After the completion of the above course students will be able to

CO1: Demonstrate the understanding and usage of core python scripting elements python constructs, data types.

CO2 : Demonstrate the understanding and usage of functions ,lists, tuples and dictionaries.

CO3: Demonstrate the understanding and usage of modules, packages and regular expressions.

CO4: Demonstrate usage of object oriented features such as Inheritance, Polymorphism, operator overloading.

CO5:Apply the knowledge of python and use the language scripting elements and constructs to develop threaded and networking applications

TEXT BOOK:

1. Core Python Programming: Dr.R.Nageshwara Rao,Dreadm Tech Press 2018

REFERENCE BOOKS:

1. Think Python, Allen Downey, Green Tea Press.
2. Learning Python, Mark Lutz, Orielly.

COs	**Mapping with POs**
CO1	PO1,PO2,PO4
CO2	PO1,PO2,PO4
CO3	PO1,PO2,PO4,PO5,PO8
CO4	PO1,PO2,PO4,PO5,PO8
CO5	PO1,PO2,PO4,PO5,PO8

Sub Title : COMPUTER NETWORKS		
Sub Code: 18IS45	**No of Credits : 3:1:0**	**No. of lecture hours/week : 4**
Exam Duration : 3 hours	**Exam Marks: CIE +Assignment + Group Activity + SEE = 40 + 5 +5+ 50 =100**	**Total No. of Contact Hours : 52**

Course Objectives:

- To understand basic concepts,topologies and OSI/TCP layers
- Understand the working of different protocols.
- To understand the working of various Network layer Routing algorithms & Transport layer services
- To understand usage of application layer like DNS, Remote login, E-mail, FTPetc.

Unit No.	**Syllabus**	**No of Hours**
1	**Data Communications :Introduction to** Data Communications; Network Models;; Layered tasks; The OSI Model and the layers in the OSI model; TCP / IP Protocol Suite. **T1: Ch 1, Ch 2, Ch 3** **Digital & Analog Transmission: Data** signals; Digital Transmission; Analog Transmission **T1: Ch:4 , Ch 5, Ch 8**	**10**
2	**Data Link Layer** **Error detection and correction** : Introduction to error detection / correction; Block coding; Linear block codes; Cyclic codes, Checksum. **Data Link control:** Framing; Flow and Error control; Protocols; Noiseless channels; Noisy channels; HDLC; Point-to-point Protocol - framing, transition phases. **Multiple Access:** Random Access; Controlled Access; Channelization. **T1: Ch 10, Ch 11,Ch 12**	**10**
3	**Network Layer** **Logical Addressing** IPv4 addresses, IPv6 addresses, Internet Protocol, Delivery, forwarding and Routing **T1: Ch 19, Ch 20,Ch 22**	**12**
4	**Transport Layer** **Process to process Delivery**: UDP, TCP, SCTP, Congestion control and Quality of Service **T1: Ch 23, Ch 24**	**10**
5	**Application Layer, Network Management:** **Domain Name System (DNS**): Name Space, Domain name space,	10

	Distribution of name space , DNS in internet, Resolution,DNS messages, Types of record. Remote Login,**E-mail**: Architecture, user agent, Message Transfer Agent(SMTP),Message Access Agent: POP and IMAP. FTP **World Wide Web and HTTP**: Architecture, web documents, HTTP: HTTP transaction, Network Management: SNMP. **T1: Ch 25, Ch 26, Ch 27, Ch 28**	

Note 1: All units will have internal choice
Note 2: Three Assignments are evaluated for 5 marks:
Assignment – I from Units 1 and 2.
Assignment – II from Units 3 and 4
Assignment -III from Unit 5

Course Outcome:

After completion of course students will be able to :

CO1: Analyze and formulate components of computer networks.
CO2:Design and develop protocols for transmission at lower layers.
CO3: Identify and develop routing algorithms for network layer.
CO4: Recognize and apply technology for transport layer services.
CO5: Demonstrate the knowledge of Computer networks for different applications.

COs	**Mapping with POs**
CO1	PO1,PO2, PO5, PO7,PO12
CO2	PO1,PO2,PO3,PO9,PO12
CO3	PO1,PO2,PO5,PO7,PO12
CO4	PO1,PO2,PO3,PO4,PO12
CO5	PO1,PO2,PO3,PO11,PO12

Text Books:

1.Behrouz A. Forouzan: Data Communications and Networking, 4th Edition, Tata McGraw-Hill, 2006.

Reference Books:

1. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
2. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
3. Wayne Tomasi: Introduction to Data Communications and Networking, Pearson Education, 2005.

Sub Title : OBJECT ORIENTED CONCEPTS LAB		
Sub Code:: 18ISL46	**No. of Credits:1= 0:0:1 (L-T-P)**	**No. of Lecture Hours/Week: 2**
Exam Duration : 3 hours	**Exam Marks: CIE + SEE = 50 + 50 = 100**	

Course Objectives:

1. Design programs using classes and objects for java
2. Build programs for automatic initialization of objects and destroy objects that are no longer required through constructors and destructors.
3. Construct applications to provide flexible options for the creation of new definitions for some of the operators.
4. Specifying mechanism of deriving a new class from older classes through inheritance.
5. Implement methods to select appropriate member function during run time.

I. LIST OF PROGRAMS

1 Create a class called Studentwith the following details as variables within it.

(i) USN
(ii) Name
(iii) Branch
(iv) Phone

Write a Java/c++ program to create nStudent objects and print the USN, Name, Branch, and Phoneof these objects with suitable headings.

2 Design

a . Java/c++ Program to demonstrate Constructor Overloadingand method overloading
b. Develop a Java/c++ Program to implementInner class and demonstrate itsAccess Protections.

3 Write a Java/c++ program to implement the Stack using arrays. Write Push(), Pop(), and Display() methods to demonstrate its working.

4 Design a superclass called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories.

5 Write a Java class called Customer to store their name and date_of_birth. The date_of_birth format should be dd/mm/yyyy. Write methods to read customer data as and display as using StringTokenizer class considering the delimiter character as “/”.

6 Write a Java/c++ program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero.

7 Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.

8 Design a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.

9 Develop JAVA programs which demonstrates utilities of Linked List Class
10 Develop JAVA Applet programs which handles Key Board Event

Note: In the examination each student picks one question from a lot of all 10 questions.

II. OPEN ENDED QUESTIONS

Identify the different objects in the following environments and implement different suitable operations

1. Banking System
2. Library
3. Automobile Industry
4. Home Appliances
5. User Interface design etc

NOTE:

1. STUDENT IS PERMITED TO SUBMIT OPEN ENDED SOLUTION TO ANY OTHER OPEN ENDED QUESTION APART FROM THE LIST ABOVE . BUT IT HAS TO BE APPROVED BY APROVED BY THE STAFF IN CHARGE.

2.IN THE EXAMINATION EACH STUDENT PICKS ONE QUESTION FROM A LOT OF ALL 10 QUESTIONS

Course Outcomes:

After completing the course the students are able to:

CO1:Design programs using classes and objects using JAVA/C++

CO2: Develop programs for automatic initialization of objects and destroy objects that are no longer required.

CO3: Develop applications to provide flexible options for the creation of new definitions for some of the operators.

CO4:Specify mechanism of deriving a new class from older classes through inheritance .

CO5: Design a program using Templates & Exception Handling.

COs	**Mapping with POs**
CO1	PO2,PO3,PO5
CO2	PO2,PO3,PO6,PO9
CO3	PO2,PO3,PO5,PO6
CO4	PO2,PO3,PO5,PO9
CO5	PO2,PO3,PO5,PO6

Sub Title : DESIGN AND ANALYSIS OF ALGORITHMS LAB		
Sub Code:18ISL47	**No. of Credits:1 =0 : 0 : 1 (L-T-P)**	**No. of Lecture Hours/Week: 2**
Exam Duration : 3 hours	**Exam Marks: CIE + SEE = 50 + 50 = 100**	

Course Objectives:

1. To introduce various algorithm design techniques.
2. To design algorithms with specific technique and implement these algorithms using the appropriate technique.
3. Enhance written and oral communication skills among students.
4. To enhance the skill to debug programs

I. LIST OF PROGRAMS

Implement the following using C/C++:

1 Design and implement an algorithm to Sort a given set of elements using DAC merge sort method and determine the time required to sort the elements. Repeat the experiment for different values of n and analyze the time complexity.

2 Print all the nodes reachable from a given starting node in a digraph using BFS method.

3 Obtain the topological ordering of vertices in a given graph using DFS method/ Source removal method

4 From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

5 Apply Prim’s algorithm to undirected graph and obtain minimum cost Spanning Tree.

6 Design and implement Heap Sort algorithm to arrange elements in desired order

7 Design and implement an algorithm to solve 0/1 Knapsack problem using dynamic programming.

8 Design and Implement Floyd’s algorithm for the All-Pairs- Shortest-Paths problem.

9 Design and implement an algorithm to solve N-Queen's problem using Back Tracking.

10 Design and implement Horspool’s algorithm.

.

Note: In the examination each student picks one question from the lot of all 10 questions.

II. OPEN ENDED QUESTIONS

Develop / Simulate Following Game Applications:

1. Knapsack
2. Spanning Trees
3. Sum of Subset
4. Travelling Sales Person etc.

NOTE:

1. STUDENT IS PERMITED TO SUBMIT OPEN ENDED SOLUTION TO ANY OTHER OPEN ENDED QUESTION APART FROM THE LIST ABOVE. BUT IT HAS TO BE APPROVED BY THE STAFF IN CHARGE.

2. IN THE EXAMINATION EACH STUDENT PICKS ONE QUESTION FROM A LOT OF ALL 10 QUESTIONS

Course Outcomes:

After the completion of course the student will be able to :

CO1: Design algorithms using different design techniques.

CO2: Implement the algorithms using C/C++.

CO3: Analyze the time complexity of algorithms.

CO4: Design key algorithmic paradigms to solve optimization problems.

COs	**Mapping with POs**
CO1	PO2,PO3,PO4,PO9
CO2	PO2,PO3,PO4,PO9
CO3	PO2,PO3,PO4,PO9
CO4	PO2,PO3,PO4,PO9

	EMPLOYABILITY SKILLS		
	Subject Code: 18HS43	**Mandatory Course**	**No of lecture hours per week: 2 Hrs**
	Exam Duration:	**Exam marks: NIL**	**Total No. of lecture hours: 26 hrs per semester**

COURSE OBJECTIVE:

1. According to the present day requirement, this unit is designed on job seeking topics, adopting a task based approach with activities and worksheets.
2. In this unit students will be taught to prepare project report effectively which is integral part of their academics by making use of referencing skills. And also they will be taught how to write an abstract and will be familiarized with research paper guidelines.
3. The unit 3 aims at preparing the student to face interviews confidently.
4. Under the topics like synonyms, antonyms and homophones students' vocabulary will be enhanced.
5. The lessons under this unit help students' to use idioms and phrases, decoding the analogies and the correct usage of collective nouns

UNIT NO	**SYLLABUS CONTENT**	**HRS/COS**
1	a. Employability Skills b. PRESENTATION SKILLS c. RESUME WRITING d. COVERING LETTER e. E-MAIL WRITING	**8** **CO1**
2	a. PROJECT REPORT b. REFERENCING SKILLS c. ABSTRACT WRITING d. RESERCH PAPER GUIDELINE AND FORMAT	**2** **CO2**
3	e. INTERVIEW SKILLS	**8** **CO3**
4	f. SYNONYMS g. ANTONYMS h. HOMOPHONES	**4** **CO4**
5	i. IDIOMS & PHRASES j. ANALOGY k. COLLECTIVE NOUNS	**4** **CO5**

COURSE OUTCOME:

1. After the completion of this unit students will have learnt to make presentations both in formal and informal situations. And also will have learnt the art of resume writing.
2. After the completion of this unit student will have learnt how to do a project report using referencing skills. And also they will have learnt how to write abstract and will have been familiarized with research paper guidelines.
3. This unit will have helped student to communicate with various skills required for job interviews.
4. After the completion of this unit the students will have learnt the strategies of vocabulary.
5. After the completion of this unit student will have learnt to use idioms and phrases in everyday conversation.

REFERENCE:

- English Skills for Technical Students by British Council, Orient Black Swan.
- A course in Grammar and Composition by Geetha Nagaraj, Cambridge University Press India Pvt. Ltd.
- Communication Skills for Professionals by Nira Konar, PHI learning Pvt. Ltd.
- Enhancing English and Employability Skills by State Board of Technical Education.