II/IV B.Tech IT Proposed syllabus of admitted batch 2015-2016

UG PROGRAM – B.TECH (IT)

Wef. Admitted batch 2015-2016

DEPARTMENT OF INFORMATION TECHNOLOGY

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

(UGC AUTONOMOUS)

(Affiliated to Andhra University, Approved by AICTE & Accredited by NBA)
Sangivalasa, Bheemunipatnam Mandal, Visakhapatnam Dt. 531162. Phone : 08933-225083/84/87 Fax:226395
Website : www.anits.edu.in email:principal@anits.edu.in

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE STRUCTURE FOR BTECH (IT) UNDER AUTONOMOUS STATUS

With effect from Admitted Batch 2015

FIRST YEAR SEMESTER – I										
Code No	**Subject**	**Category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT111	English	HS	3	1	-	4	40	60	100	3
IT112	Engineering Mathematics-I	BS	3	1	-	4	40	60	100	3
IT113	Engineering Chemistry	BS	3	1	-	4	40	60	100	3
IT114	Professional Ethics & Human Values	HS	2	1	-	3	100	-	100	2
IT115	Basics of Electronics Engineering	ES	3	1	-	4	40	60	100	3
IT116	Engineering Chemistry Lab	BS	-	-	3	3	50	50	100	2
IT117	Programming with C Lab	ES	-	1	3	4	50	50	100	3
ITAC1	NCC/NSS/SPORTS				3	3				
	TOTAL		**14**	**6**	**9**	**29**	**360**	**340**	**700**	**19**

FIRST YEAR SEMESTER – II										
Code No	**Subject**	**category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT121	Engineering Mathematics-II	BS	3	1	-	4	40	60	100	3
IT122	Engineering Physics	BS	3	1	-	4	40	60	100	3
IT123	Environmental Science	BS	3	1	-	4	40	60	100	3
IT124	Engineering Drawing	ES	1	-	3	4	40	60	100	3
IT125	Elements of Electrical Engineering	ES	3	1	-	4	40	60	100	3
IT126	Engineering Physics lab	BS	-	-	3	3	50	50	100	2
IT127	Language lab	HS	-	-	3	3	50	50	100	2
IT128	Object Oriented Programming with C++ Lab	ES	-	1	3	4	50	50	100	3
IT129	Workshop	ES	-	-	3	3	50	50	100	2
ITAC2	NCC/NSS/SPORTS				3	3				
	TOTAL		**13**	**5**	**18**	**36**	**400**	**500**	**900**	**24**
FIRST SEM , SECOND SEM COMMON FOR CSE & IT DEPARTMENTS										

SECOND YEAR SEMESTER – I										
Code No	**Subject**	**Category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT211	Data Structures	PC	4	1	-	5	40	60	100	4
IT212	Digital Logic Design	ES	3	1	-	4	40	60	100	3
IT213	Discrete Mathematical Structures	BS	4	1	-	5	40	60	100	4
IT214	Computer Organization	PC	4	1	-	5	40	60	100	4
IT215	Data Communications	PC	3	1	-	4	40	60	100	3
IT216	Data Structures Lab	PC	-	-	3	3	50	50	100	2
IT217	Digital Electronics lab	ES	-	-	3	3	50	50	100	2
IT218	Python Programming Lab	PC	-	1	3	4	50	50	100	3
	TOTAL		**18**	**6**	**9**	**33**	**350**	**450**	**800**	**25**

SECOND YEAR SEMESTER – II										
Code No	**Subject**	**Category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT221	Computer Networks	PC	3	1	-	4	40	60	100	3
IT222	Information Systems Design	PC	3	1	-	4	40	60	100	3
IT223	Operating Systems	PC	4	1	-	5	40	60	100	4
IT224	Probability Statistics & Queuing Theory	BS	4	1	-	5	40	60	100	4
IT225	Computer Graphics & Multimedia	PC	3	1	-	4	40	60	100	3
IT226	Networking Lab	PC	-	-	3	3	50	50	100	2
IT227	Computer Graphics & Multimedia Lab	PC	-	-	3	3	50	50	100	2
IT228	Operating Systems (Linux) Lab	PC	-	-	3	3	50	50	100	2
TOTAL			**17**	**5**	**9**	**31**	**350**	**450**	**800**	**23**

THIRD YEAR SEMESTER – I										
Code No	**Subject**	**Category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT311	Object Oriented Programming through JAVA	PC	3	1	-	4	40	60	100	3
IT312	Database Management Systems	PC	3	1	-	4	40	60	100	3
IT313	Unix Network Programming	PC	3	1	-	4	40	60	100	3
IT314	Formal Languages Automata Theory	PC	4	1	-	5	40	60	100	4
IT315	*** Open Elective-1**	OE	3	1	-	4	40	60	100	3
IT316	Unix Network Programming Lab	PC	-	-	3	3	50	50	100	2
IT317	Java Programming Lab	PC	-	-	3	3	50	50	100	2
IT318	Database Management Systems Lab	PC	-	-	3	3	50	50	100	2
IT319	Quantitative Aptitude - 1 & Verbal Aptitude – 1	HS	4	-	-	4	100	-	100	2
TOTAL			**20**	**5**	**9**	**34**	**450**	**450**	**900**	**24**

THIRD YEAR SEMESTER – II										
Code No	**Subject**	**Category**	**Periods**				**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
			Lecture	**Tutorial**	**Practical**	**Total**				
IT321	Compiler Design	PC	4	1	-	5	40	60	100	4
IT322	Design & Analysis of Algorithms	PC	4	1	-	5	40	60	100	4
IT323	Object Oriented Analysis and Design with UML	PC	3	1	-	4	40	60	100	3
IT324	Mobile computing	PC	3	1	-	4	40	60	100	3
IT325	**Professional Elective-I**	PE	4	1	-	5	40	60	100	4
IT326	Mobile computing Lab	PC	-	-	3	3	50	50	100	2
IT327	Web based open source technologies Lab	PC	-	1	3	4	50	50	100	3
IT328	Computer Aided Software Engineering tools lab	PC	-	-	3	3	50	50	100	2
IT329	Soft Skills lab	HS	-	-	3	3	100	-	100	2

IT3210	Quantitative Aptitude-2 & Verbal Aptitude – 2	HS	4	-	-	4	100	-	100	2
TOTAL			**22**	**6**	**12**	**40**	**550**	**450**	**1000**	**29**

FOURTH YEAR SEMESTER – I

Code No	Subject	Category	Periods				Sessional Marks	External Marks	Total Marks	Credits
			Lecture	Tutorial	Practical	Total				
IT411	Data Analytics	PC	3	1	-	4	40	60	100	3
IT412	Cryptography & Network Security	PC	3	1	-	4	40	60	100	3
IT413	*** Open-Elective-II**	PE	3	1	-	4	40	60	100	3
IT414	**Professional Elective – II**	SE	4	1	-	5	40	60	100	4
IT415	**Professional Elective – III**	PE	4	1	-	5	40	60	100	4
IT416	Analytics Lab	PC	-	-	3	3	50	50	100	2
IT417	Network Security Lab	PC	-	-	3	3	50	50	100	2
IT418	PROJECT I	PW	-	-	6	6	100	-	100	4
IT419	# Industrial Training	IT	-	-	-	-	-	100	100	2
TOTAL			**17**	**5**	**12**	**34**	**400**	**500**	**900**	**27**

Assessment done based on the industrial training taken after Third Year Semester - II

FOURTH YEAR SEMESTER – II

Code No	Subject	Category	Periods				Sessional Marks	External Marks	Total Marks	Credits
			Lecture	Tutorial	Practical	Total				
IT421	**Professional Elective – IV**	PE	4	1	-	5	40	60	100	4
IT422	**Professional Elective – V**	PE	4	1	-	5	40	60	100	4
IT423	PROJECT II	PW	-	-	9	9	100	100	200	8
IT424	Massive Open Online Course	OOC	-	-	-	-	100	-	-	2
TOTAL			**8**	**2**	**9**	**19**	**280**	**220**	**500**	**18**
TOTAL CREDITS										**189**

- The total no of credits required to award B Tech Degree :180. Minimum of 40 students have to opt for an elective to offer.
- Students are enabled for self-learning outside the class rooms through Academic activities , Extra/Co-Curricular activities / NSS/NCC/Sports.
- Academic activities I- Paper Presentation, Participation in Programming/coding contests.
- Academic activities II- Certificate of participation related to Skill Development Programs/Advanced Topics.
- A massive open online course (MOOC) is a free Web-based distance learning program Offered with 2 credits

Professional Elective –I	Professional Elective-II	Professional Elective-III	Professional Elective-IV	Professional Elective-V
Operation Research	Artificial Intelligence	Machine Learning	Image Processing	Soft computing
Database Administration	Database Storage Management	Advanced Data base Management System	Data Mining	Advances in Data Mining
IT Infrastructure and Management	Distributed Operating System	Ethical Hacking	Parallel Computing	Cloud Computing
Principles of Programming Language	User Interface Design	Client Server Technologies	Human Computer Interaction	E-commerce

OPEN ELECTIVES I & II		
SNO	**NAME OF THE COURSE**	**NAME OF THE DEPARTMENT OFFERING THE COURSE**
1	Essentials of Information Technology	Department of Information Technology
2	Foundations of Web Development & Design	
3	IT Infrastructure & Management	
4	Data Structures	
5	Multimedia Concepts	
6	Principles of Ethical Hacking	
7	Open Source Technologies	
8	Database Management Systems	
9	Operating Systems	
10	Software Engineering	
11	Computer Operating Systems	Department of Computer Science & Engineering
12	Cloud Computing overview	
13	Introduction to soft computing	
14	Concepts of Object Oriented Programming	
15	Database Management Systems	
16	Fundamentals of Computer Networks	
17	Web Designing	
18	Robotics	Department of Mechanical Engineering
19	Finite Element Analysis	

20	Computer Aided Design	
21	Operations Research	
22	Rain water Harvesting	Department of Civil Engineering
23	Earth Quake Resistant Design of Structures	
24	Disaster management and Mitigation	
25	Environmental Impact Analysis	
26	Urban planning and sustainable Development	
27	Industrial Safety & Hazards Management	Department of Chemical Engineering
28	Theoretical Biology	
29	Fuel Cell Technology	
30	Design of Experiments	
31	Food Processing Technology	
32	Corrosion Engineering	
33	Computational Tools for Engineering	
34	Modeling and Optimization	
35	Renewable Energy Technologies	Department of Electrical & Electronics Engineering
36	Fundamentals of Electric Power Utilization	
37	Nano Technology & Engineering Applications	Department of Physics
38	Principles and Applications of NDT Methods	
39	Concepts of Physics for Civil Engineers	
40	Introduction to embedded system Design	Department of Electronics and Communication Engineering
41	Introduction to VLSI Design	
42	Electronic Design with Integrated Circuits	
43	Digital Electronics	
44	Applications of Fields and Waves	
45	Special Topics – Electronics	
46	Introduction to Image Processing /Computer vision	
47	Applied Electronics	

IT211 **DATA STRUCTURES** **CREDITS: 4**

Instruction: 4 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE :
Following this course students will be able to:

- Assess how the choice of data structures impacts the performance of programs.
- Choose the appropriate data structure and algorithm design method for a specified application.
- Solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions.

COURSE OUTCOMES:

1	Analyze, evaluate and choose appropriate abstract data types and algorithms to solve particular problems
2	Compare and contrast the benefits of dynamic and static data structures implementations
3	Design and implement abstract data types such as linked list, stack, queue and tree by Using C as the programming language using static or dynamic implementations
4	Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	3	3	3	3	3	2	3				2	3
CO-2	3	2	3	3	3	2	2				2	3
CO-3	3	3	3	3	3	2	0		3		3	3
CO-4	3	3	3	3	3	3	3		3		3	3

SYLLABUS

Unit-1: Introduction (12 Periods)

Introduction to data structures, arrays and structures. Dynamic Memory Management, Abstract Data Type (ADT). Introduction to Time and Space complexity and their tradeoffs.

List : Definition and examples- Primitive Operations- Representation using array and Linked List. Types of Linked Lists and implementation: single, double and circular. The array and linked list advantages, disadvantages and applications.

Unit-2:Stacks and Queues (12 Periods)

The Stack ADT: Definition, Primitive Operations and representation. Stack ADT implementation using array and linked list. Applications of Stacks: Prefix, infix and Postfix notations, conversion between infix, prefix and postfix, postfix evaluation using stacks. Recursion: definition and examples (ex: Towers of Hanoi Problem, other examples).

Queue ADT: Definition, Primitive operations and Representation. Queue ADT implementation using array and linked list. Types of Queue: Circular Queue, Priority Queue, De-queue Operations and implementation using array and linked list. The queues advantages, disadvantages, and applications.

Unit - 3: Sorting and Searching (12 Periods)

Sorting: General background, selection sort, bubble sort, insertion sort, shell sort, radix sort, quick sort and merge Sort.

Searching: General background, linear search, binary search and Interpolation search.

Introduction to Hashing, Hash Function, Hashing techniques, Collision Resolution Methods: Open Addressing, Chaining.

Unit-4: Trees (12 Periods)

Trees: Introduction, Terminology, Binary trees: Terminology, Representation. Binary tree implementation using array and linked list. Tree Traversal Techniques, applications and threaded binary trees.

Types: Heap, Binary Search Tree, AVL Tree, B-Tree of order m, introduction to Red-Black tree.

Unit-5: Graphs (16 periods)

Graphs: Introduction- terminology, Representation of graphs-linked list and adjacency matrix, Representation in C, Implementation of graphs using arrays and linked list, Graph traversals-Breadth-First Search, Depth-First Search. Spanning Trees: Introduction and terminology, Minimum Spanning Tree algorithms: Prims and Krushkals. Applications of Graphs: Dijkstra's & Warshall's Algorithm.

TEXT BOOKS:

1. Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structure, computer science Press.

REFERENCE BOOKS:

1. Y. Langsam, M. Augenstin and A. Tannenbaum, “Data Structures using C” Pearson Education, 2nd Edition, 1995.
2. Richard F, Gilberg , Forouzan, Cengage ,”Data Structures”, 2/e, 2005.

IT212 **DIGITAL LOGIC DESIGN** **CREDITS: 3**
(COMMON FOR CSE & IT)

Instruction: 3 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

- To provide knowledge and understanding of Boolean algebra and digital concepts.
- To provide the knowledge of analyzing and designing of combinational and sequential logic networks.
- HDL in this course provides the ability to synthesize the designs in Verilog HDL or VHDL.

COURSE OUTCOMES:

After the completion of the course, the student would be able to

CO 1: Analyze and synthesize logic circuits by applying the knowledge of number systems, codes ,Boolean algebra and digital logic circuits to solve typical problems on the same.

CO 2: Minimize the given Switching function in SOP and POS forms using K-Map & Design of different types of combinational logic circuits using various logic gates.

CO3: Design and analyze synchronous sequential logic circuits including registers & counters using gates & flip-flops.

CO 4: Design combinational logic circuits using different types of PLDs, namely, PROM, PLA and PAL.

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	2								3		3	
CO-2	3								3		3	
CO-3	3								3		3	
CO-4	3								3		3.	

UNIT I Binary Systems, Boolean Algebra and Logic Gates (10 Periods)

Digital Systems, Binary Numbers, Number Systems, Base Conversion Methods, Complements, Signed Binary Numbers, Binary Codes, Binary Logic.

Basic Definitions, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra. Boolean Functions, Canonical and Standard Forms, Different Logic Operations, Digital Logic Gates.

UNIT II Gate-Level Minimization (4 Periods)

The Map Method, Minimal Functions and their properties, Don't-Care Conditions, Tabulation Method, NAND and NOR Implementation, Other Two- Level Implementations, Verilog Hardware Description Language (Verilog HDL).

Combinational Logic Design: **(6 Periods)**

Combinational Circuits, Analysis Procedure, Design Procedure, Design of adders, subtractors, adder-subtractor circuit, BCD adder circuit, applications of adders, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, Demultiplexers, Verilog HDL For Combinational Circuits.

UNIT III Sequential Logic Circuits (5 Periods)

Sequential Circuits, Latches, Flip-Flops, Analysis of Clocked Sequential Circuits, Flip-Flop Conversions, Verilog HDL for Sequential Circuits.

Registers and Counters (6 Periods)

Registers, Shift Registers, Ripple Counters, Synchronous Counters, Johnson and Ring counters, Verilog HDL for Registers and Counters.

UNIT IV Synchronous Sequential Logic (4 Periods)

Basic Design Steps, Serial Adder Example, State Reduction & Assignment Problem.

Fundamentals of Asynchronous Sequential Logic (5 Periods)

Introduction, Analysis Procedure, Design Procedure, circuits with latches, Races and Hazards.

UNIT-V: Programmable Logic Devices (8 Periods)

Programmable Logic Devices : PROM, PLA, PAL, realization of switching functions using PROM, PLA and PAL; comparison of PROM, PLA and PAL, Programming tables of PROM, PLA and PAL, Sequential Programmable Devices.

Text Books :

1. M. Morris Mano, Digital Design, Pearson Education, Inc., 2008, 4th Edition.

Reference Books:

1. Zvi Kohavi, Switching and Finite Automata Theory, Tata McGraw-Hill, 1978, 2nd Edition.
2. Frederick, Introduction to Switching Theory and Logical Design,2011 & J. Hill and Gerald R. Peterson, John Wiley and Sons, 2011, 3rd Edition.
3. William I. Fletcher, An Engineering Approach to Digital Design, PHI, 2008.

IT213 DISCRETE MATHEMATICAL STRUCTURES CREDITS: 4
(COMMON FOR CSE & IT)

Instruction: 4 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

- The knowledge of Mathematics is necessary for a better understanding of almost all the Engineering and Science subjects. Here our intention is to make the students acquainted with the concept of basic topics from Mathematics, which they need to pursue their Engineering degree in different disciplines.

COURSE OUTCOMES:

1. To understand set theory, relations , mathematical logic , mathematical reasoning and to study about the validity of the arguments.
2. Be able to apply basic counting techniques to solve combinatorial problems.
3. To understand Recurrence Relation, Generating functions and solving problems involving recurrence equations.
4. To familiarize the different types of binary relations and related algorithms on transitive closure.
5. To familiarize with the applications of graphs , trees and algorithms on minimal spanning tress.

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	2								2		2	3
CO-2	3	2										
CO-3	2											
CO-4	2											
CO-5	2		2									

SYLLABUS

UNIT-I : MATHEMATICAL LOGIC (15 Periods)

Sets-Operations on sets-relations-functions-Fundamentals of Logic- Logical inferences-Methods of proof of an implication-First Order logic and Other methods Proof -Rules of inference for quantified Propositions –Mathematical Induction.

UNIT II : ELEMENTARY COMBINATORICS (08 Periods)

Basics of Counting- Combinations and Permutations-Their Enumeration with and without repetition-Binomial coefficients-Binomial and Multinomial Theorems-The Principle of Inclusion-Exclusion.

UNIT III : RECURRENCE RELATIONS (08 Periods)

Generating Functions of Sequences-Calculating their Coefficients-Recurrence relations-Solving recurrence relations-Method of characteristic Roots- Non-homogeneous Recurrence relations and their solutions.

UNIT IV : RELATIONS AND DIGRAPHS (09 Periods)

Relations and directed Graphs - Special Properties of Binary relations- Equivalence Relations-Ordering Relations-Lattices and Enumeration- Operations on relations-Paths and Closures-Directed Graphs and Adjacency matrices .

UNIT V : GRAPHS (20 Periods)

Introduction to Graphs – types of Graphs – Graphs basic terminology and special types of simple graphs – representation of graphs and graph isomorphism – Euler paths and circuits- Hamilton paths and circuits – Planar graphs – Euler's formula.

Introduction to Trees and their properties – Spanning Trees – Depth First Search , Breadth First Search – Minimum Spanning Trees – Kruskal's Algorithm and Prim's Algorithm.

TEXT BOOKS:

1) Joe L. Mott, Abraham Kandel & T. P. Baker, "Discrete Mathematics for computer scientists & Mathematicians" Prentice Hall of India Ltd, New Delhi.

REFERENCE BOOKS:

1) Keneth. H. Rosen, "Discrete mathematics and its applications", Tata McGraw- Hill Publishing Company, New Delhi

2) Richard Johnsonbaug ,"Discrete mathematics" , Pearson Education, New Delhi.

IT214 **COMPUTER ORGANIZATION** **CREDITS: 4**

Instruction: 4 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVES:

- clearly differentiate between Computer Organization and Computer Architecture
- identify and describe the functions of all the basic components making up a computer system
- Present, as clearly and completely as possible, the characteristics of modern-day computer systems, highlighting on the CPU Organization & Operation, Number systems, Operating Systems Memory Systems, Logic Circuits Design and I/O and Interfacing.
- Engage into contrast discussions based on the two CPU design philosophies i.e the Complex Instruction Set Computers (CISC) and Reduced Instruction Set Computers (RISC) systems

COURSE OUTCOMES:

1. Understand Register transfer language, computer instructions and solve problems using micro operations
2. Analyze micro program control to implement micro program instructions
3. Understand central processing unit, stack organization and to evaluate stack operations
4. Review peripheral devices , types of memories and analyze how mapping is done between various memories.

MAPPING OF COS AND POS

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3	3	2	1	-	-	-	-	-	-	-	3
CO-2	3	2	2	3	-	-	-	-	-	-	-	3
CO-3	3	2	2	2	-	-	-	-	-	-	-	2
CO-4	3	3	2	1	-	-	-	-	-	-	-	2

SYLLABUS

UNIT-I **(14 Periods)**

Register transfer and micro operations :

Register Transfer Language, Bus and Memory Transfers, Arithmetic, Logic and Shift Micro operations, Arithmetic Logic Shift Unit

Computer Arithmetic:

Introduction, Addition and Subtraction, Booth Multiplication Algorithm, Decimal Arithmetic Unit.

UNIT-II : **(12 Periods)**

Basic Computer Organization:

Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt, Complete Computer Description.

UNIT-III : **(12 Periods)**

Control Design:

Hardwired & Micro Programmed (Control Unit), Control Memory, Address Sequencing, Conditional and Unconditional Branching, Micro program Example.

UNIT-IV : **(12 Periods)**

Central Processing Unit:

Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes with numerical examples, Data Transfer and Manipulation, Program Control, Program Interrupt, Types of interrupts, CISC Characteristics, RISC Characteristics..

UNIT-V :

(14 Periods)

Input-Output Organization:

Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access.

Memory Organization:

Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

TEXT BOOKS:

1. M.MorrisMano ,Computer System Architecture, Third Edition, Pearson Education Inc., 2003

REFERENCE BOOKS:

1. John D. Carpinelli, Computer Systems Organization and Architecture,Pearson Education Inc., 2003.
2. William Stallings, Computer Organization and Architecture,5th Edition,2000.

IT215 **DATA COMMUNICATIONS** **CREDITS: 3**

Instruction: 3 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVES:

- Introduce students to the evolution of computer networks and the concepts data communication;
- Introduce students the general principles of network design and compare the different network topologies
- Introduce students to the digital and analogue representations and channels;
- Describe the mechanism and techniques of encoding;
- Introduce students to the general principles of circuit and packet switching;
- Introduce students to the wireless Local Area Networks;
- Provide students with in-depth knowledge of data link layer fundamental such as error detection, correction and flow control techniques; multiple access control techniques

COURSE OUTCOMES:

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

1	Understand the basic concepts of Data Communications and different models
2	Understand and analyses the characteristics of signals propagated through different transmission media
3	Apply signal encoding techniques, error detection, correction techniques and learn interfacing
4	Distinguish various Multiplexing techniques and learn various modems like ADSL, xDSL.
5	Illustrate various Data link control protocols namely flow control, error control and HDLC

MAPPING OF COS AND POS

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

SYLLABUS

UNIT I **(12 periods)**

Data Communication overview: A communication model, Data communications, Data Communication networking- *Introduction to WAN, LAN, wireless Networks, MAN* , an example configuration **Data Transmission**: Concepts and Terminology-*Transmission terminology, Frequency, spectrum and Bandwidth* Analog and Digital Data Transmission- *Introduction to Analog and Digital Data, Analog and Digital Signals, Analog and Digital Transmission*, Transmission Impairments-*Attenuation, Delay Distortion, Noise*, channel Capacity-*Nyquist Bandwidth,Shannon Capacity Formula,The expression Eb/N0* **Transmission media**: guided transmission media-*Twisted pair, coaxial cable, Optical fiber*, Wireless transmission – *Antennas, terrestrial microwave, satellite microwave, Broadcast Radio, Infrared* Wireless Propagation-*Ground wave propagation, sky wave propagation, Line- of-sight Propagation*, Line-of-sight Transmission- *free space loss, Atmospheric Absorption, Multipath, Refraction*

UNIT II **(10 periods)**

Signal Encoding Techniques: Digital Data Digital signals(*Nonreturn to Zero(NRZ), multilevel Binary, Biphase, Modulation rate*), Digital Data Analog Signals (*Amplitude shift keying, frequency shift keying, Phase Shift keying ,Quadrature Amplitude Modulation*), Analog Data Digital Signals(*Pulse code Modulation ,Delta Modulation*), Analog Data Analog Signals(Amplitude *Modulation, Angle Modulation)*

UNIT III **(6 periods)**

Digital Data communication Techniques: Asynchronous and synchronous Transmission-*Asynchronous Transmission, synchronous transmission*, Types of Errors, Error Detection-*parity check, CRC*, Error correction-*Block Code Principles*, Line configuration-*Topology, Full Duplex and Half Duplex*, Interfacing – *V.24/EIA-232-F, ISDN Physical Interface*

UNIT IV **(10 periods)**

Multiplexing: Frequency Division Multiplexing -*Characteristics, Analog carrier systems, wave length-Division Multiplexing*, Synchronous Time Division Multiplexing- *Characteristics, TDM link control, Digital carrier systems, SONET/SDH* , Statistical Time Division Multiplexing-*characteristics, performance, cable modem*, Asymmetric digital subscriber line-*ADSL Design, Discrete Multitone*, xDSL-*HDSL,SDSL,VDSL*, modems

UNIT V **(10 periods)**

Data Link Control : Flow Control-*stop and wait flow control, sliding window flow control,* Error Control- *stop-and-wait ARQ, selective- Reject ARQ* , High Level Data Link Control (HDLC) – *Basic Characteristics, Frame Structure, operation* **,** Architecture of computer network, layered approach,X.25, Frame relay, ATM.

Basic hardware: *RJ- 45,Network interface card, rack, cable standard-Category 5,6,and 7, cross connection, straight connection cable coding standards.*

TEXT BOOKS :

1. William Stallings ,"Data & Computer Communication", Pearson Education, 7th edition

REFERENCE BOOKS:

1. Forouzan, "Data communication and networking", TATA McGraw, 4th edition
2. Gupta Prakash C.,"Data communication", PHI Learning
3. Tomasi, "Introduction to Data Communication & Networking, Pearson Education
4. A.S Tanenbum, "Computer Network",Pearson Education

IT216 **DATA STRUCTURES LAB** **CREDITS: 2**

Practical: 3 Periods /week Sessional Marks: 50
End- Exam : 3 Hours End-Exam-Marks:50

COURSE OBJECTIVE :

Following this course students will be able to:

- Assess how the choice of data structures impacts the performance of programs.
- Choose the appropriate data structure and algorithm design method for a specified application.
- Solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions.

COURSE OUTCOMES:

1 Implement linear data structures such as stacks, queues, linked lists and apply on real time problem like conversions & evaluations of expressions.
2 Implement non linear data structures such as Trees and Graphs and apply on real time problem like finding shortest path.
3 Implement different sorting and searching techniques.

MAPPING OF COS AND POS

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3	3	3	3	3	2	2		3	2	3	3
CO-2	3	3	3	3	3	2	2		3	2	3	3
CO-3	3	3	3	3	3	2	2		3	2	3	3

List of Programs:

1) Programs to implement the following using an array.
a) Stack b) Queue

2) Programs to implement the following using a singly linked list.
a) Stack b) Queue

3) Program to do the following
a) Infix to postfix conversion. b) Evaluation of postfix expression.

4) Programs to implement the following data structures.
a) Circular Queue b) Priority Queue

5) Implement primitive operations of de-queue (double ended queue) using a doubly linked list and an array.

6) Program to perform the following operations:
a) Insert an element into a binary search tree.
b) Delete an element from a binary search tree.
c) Search for a key element in a binary search tree.

7) Program that use non-recursive functions to traverse the given binary tree in
a) Preorder b) In-order c) Post-order.

8) Program to implement bfs and dfs for a given graph.

9)) Program to implement the following sorting methods:
a) Merge sort b) Quick sort c) Insertion Sort d) Selection Sort

10)) Program to implement the following searching methods:
a) Linear Search b) Binary search

11) Program to store **k** keys into an array of size n at the location computed using a Hash function, loc = key % n, where k<=n and k takes values from [1 to m], m>n, where m is size of the hash table.

12) Write a C program to handle the collisions using the following collision resolution Technique
a) Linear probing b) Quadratic probing c)Separate Chaining

REFERENCE BOOKS

- Y. Langsam, M. Augenstin and A. Tannenbaum, "Data Structures using C" Pearson Education, 2nd Edition, 1995.

- Richard F, Gilberg ,Forouzan, Cengage,Data Structures, 2/e, 2005.
- Data Structures using C,2/2, ISRD Group.

IT217 **DIGITAL ELECTRONICS LAB** **CREDITS: 2**
(COMMON FOR CSE & IT)

Practical: 3 Periods /week Sessional Marks: 50
End- Exam : 3 Hours End-Exam-Marks:50

LIST OF LABORATORY EXPERIMENTS

CYCLE-I:

1. Study of passive, active components & Integrated Circuits.
2. To study the regulation characteristics of given Integrated Circuits.
3. To verify the adder operation & subtractor operation using Operational amplifiers.
4. To verify the truth tables of given Logic Gates.

CYCLE-II:

1. Verification of truth tables of Logic gates using IC's.
2. Design a combinational circuit for Code Converters using IC's.
3. Design a combinational circuit for Adders & Subtractors (HA & FA) using IC's.
4. Design a sequential circuit for Flip-flop and verify its characteristics using IC's.
5. Design a bi-directional Universal Shift Register Using IC74LS194.
6. Design of Counters using IC74LS73.

CYCLE-III: (Simulation using VHDL)

1. Write a program for verification of Basic Gates.
2. Write a program for Adder & Subtractor.
3. Write a program for flip flops.
4. Write a program for MUX & DEMUX.
5. Write a Program for Shift Registers.

*** NOTE: FOUR Experiments from each cycle should be done compulsorily.**

IT218 **PYTHON PROGRAMMING LAB** **CREDITS: 3**

Practical: 3 Periods & 1 Tut /week Sessional Marks: 50
End- Exam : 3 Hours End-Exam-Marks:50

COURSE OBJECTIVE:

The student will be able to:

- Describe the basic elements of the Python language and the Python interpreter
- Analyze and demonstrate the use of lists, tuples, dictionaries in Python.
- Write classes to demonstrate the ideas of encapsulation, inheritance, interfaces and object oriented program design.
- Explain and demonstrate methods of error handling and Python exceptions.
- Write to and read from files using intermediate file I/O operations in a Python program.
- Solve problems that have origins in a variety of disciplines including math, science, the Internet and business.

COURSE OUTCOMES:

At the end of the course students should:
CO 1 : Understand and use the syntaxes of python in problem solving
CO 2 : Apply python data structures to solve real world problems
CO 3 : Implement object oriented concepts in python programming
CO 4 :Demonstrate File I/O and exception handling

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	3	3	3	3	3						1	
CO-2	3	3	2	2	2						2	
CO-3	3	3	3	3	3						1	
CO-4	3	3	2	2	2						1	

List of the experiments to be done on the following topics

1. Introduction: Introduction to Python programming language, using the interpreter, running scripts, variables, assignments, comments, operators and expressions. Introduction to basic data types including strings, integers, lists and tuples.

2. Control Flow: Conditional expressions, if statement, for statement and while statement, break and continue statements.

3. Functions, Methods and Modules: Introduction to built-in functions, methods and modules. Introduction to standard library modules like sys, os, time and random. Reading command-line arguments. Introduction to writing user-defined functions and organizing code into modules.

4. Data structures: Detailed overview of four major data structures of Python including list, tuple, set and dictionary, including list slicing, sorting lists, list comprehensions.

5. Working with Files: Introduction to reading and writing files, text and binary mode. Writing parsers for simple text formats.

6. Classes and Exceptions : Introductions to classes, object creation and class inheritance and overriding methods. Introduction to exception handling.

7. Advanced Topics

Introduction to some advanced topics in Python.

- Downloading things from web
- Web programming
- Data visualization
- Building simple games using pygame

LIST OF EXPERIMENTS

LEVEL 1: FUNDAMENTAL PROGRAMMING

1. Installation of Python - using python interpreter and printing HELLO WORLD message

2. program that accept the user's first and last name and print them in reverse order with a space between them

3. To display the first and last colors from the following list.
   color_list = ["Red","Green","White" ,"Black"]

4. To count the number of characters (character frequency) in a string . Sample String : google.com' Expected Result : {'o': 3, 'g': 2, '.': 1, 'e': 1, 'l': 1, 'm': 1, 'c': 1}

5. To convert temperatures to and from celsius, fahrenheit.
[Formula : c/5 = f-32/9 [where c = temperature in celsius and f = temperature in fahrenheit] . *Expected Output* : 60°C is 140 in Fahrenheit 45°F is 7 in Celsius

6. To get a list, sorted in increasing order by the last element in each tuple from a given list of non-empty tuples.
Sample List : [(2, 5), (1, 2), (4, 4), (2, 3), (2, 1)]
Expected Result : [(2, 1), (1, 2), (2, 3), (4, 4), (2, 5)]

7. Write a function translate() that will translate text into “rovarspraket” (Swedish secret language) That is double every consonant and place an occurence of “o” in between . For example translate(“this is fun”) should return the string “tothohisos isos fofunon”

8. program that prints each item and its corresponding type from the following list.
Sample List : datalist = [1452, 11.23, 1+2j, True, 'w3resource', (0, -1), [5, 12], {"class":'V', "section":'A'}]

9. Write a Python function that takes a list and returns a new list with unique elements of the first list. *Sample List :* [1,2,3,3,3,3,4,5] *Unique List :* [1, 2, 3, 4, 5]

10. programs for the following:
    a. Defining and Accessing a Dictionary:
    b. Updating Dictionary:
    c. Deleting Dictionary Elements:
    d. Defining and Accessing,updating ,deleting Tuples.

11. To demonstrate the use of built-in string method

12. To demonstrate the use of lists

LEVEL 2 : CLASSES AND I/O

13. To implement classes concept in python
14. To implement inheritance in a Banking system
15. To implement polymorphism
16. Python Programs on Exception Handling
    a. Write a python program to handle Number format error
    b. Write a python program to handle IOError
17. Write a python program to perform the following file operations.
    a. Create, open & close a file:
    b. write content on to a file
    c. Read content form the file

d. Random access operation on files using tell & seek functions
e. other file operations using the Module ‘os’

LEVEL 3 : PROBLEM SOLVING

18. Cryptography :

In cryptography, a *Caesar cipher* is a very simple encryption techniques in which each letter in the plain text is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 3, A would be replaced by D, B would become E, and so on. The method is named after Julius Caesar, who used it to communicate with his generals. *ROT-13* ("rotate by 13 places") is a widely used example of a Caesar cipher where the shift is 13. In Python, the key for ROT-13 may be represented by means of the following dictionary:

key = {'a':'n', 'b':'o', 'c':'p', 'd':'q', 'e':'r', 'f':'s', 'g':'t', 'h':'u',

a. 'i':'v', 'j':'w', 'k':'x', 'l':'y', 'm':'z', 'n':'a', 'o':'b', 'p':'c',
b. 'q':'d', 'r':'e', 's':'f', 't':'g', 'u':'h', 'v':'i', 'w':'j', 'x':'k',
c. 'y':'l', 'z':'m', 'A':'N', 'B':'O', 'C':'P', 'D':'Q', 'E':'R', 'F':'S',
d. 'G':'T', 'H':'U', 'I':'V', 'J':'W', 'K':'X', 'L':'Y', 'M':'Z', 'N':'A',
e. 'O':'B', 'P':'C', 'Q':'D', 'R':'E', 'S':'F', 'T':'G', 'U':'H', 'V':'I',
f. 'W':'J', 'X':'K', 'Y':'L', 'Z':'M'}

Your task in this exercise is to implement an encoder/decoder of ROT-13. Once you're done, you will be able to read the following secret message:

Pnrfne pvcure? V zhpu cersre Pnrfne fnynq!

Note that since English has 26 characters, your ROT-13 program will be able to both encode and decode texts written in English.

19. Speech synthesis:

The *International Civil Aviation Organization (ICAO) alphabet* assigns code words to the letters of the English alphabet acrophonically (Alfa for A, Bravo for B, etc.) so that critical combinations of letters (and numbers) can be pronounced and understood by those who transmit and receive voice messages by radio or telephone regardless of their native language, especially when the safety of navigation or persons is essential. Here is a Python dictionary covering one version of the ICAO alphabet:

d = {'a':'alfa', 'b':'bravo', 'c':'charlie', 'd':'delta', 'e':'echo', 'f':'foxtrot',
'g':'golf', 'h':'hotel', 'i':'india', 'j':'juliett', 'k':'kilo', 'l':'lima',
'm':'mike', 'n':'november', 'o':'oscar', 'p':'papa', 'q':'quebec', 'r':'romeo',
's':'sierra', 't':'tango', 'u':'uniform', 'v':'victor', 'w':'whiskey',
'x':'x-ray', 'y':'yankee', 'z':'zulu'}

Your task in this exercise is to write a procedure `speak_ICAO()` able to translate any text (i.e. any string) into *spoken*ICAO words. You need to import at least two libraries: `os` and `time`. On a mac, you have access to the system TTS (Text-To-Speech)

as follows: `os.system('say ' + msg)`, where `msg` is the string to be spoken. (Under UNIX/Linux and Windows, something similar might exist.) Apart from the text to be spoken, your procedure also needs to accept two additional parameters: a float indicating the length of the pause between each spoken ICAO word, and a float indicating the length of the pause between each word spoken

20. Cows and bulls game :

Create a program that will play the "cows and bulls" game with the user. The game works like this:
Randomly generate a 4-digit number. Ask the user to guess a 4-digit number. For every digit that the user guessed correctly *in the correct place*, they have a "cow". For every digit the user guessed correctly *in the wrong place* is a "bull." Every time the user makes a guess, tell them how many "cows" and "bulls" they have. Once the user guesses the correct number, the game is over. Keep track of the number of guesses the user makes throughout the game and tell the user at the end.
Say the number generated by the computer is 1038. An example interaction could look like this:
Welcome to the Cows and Bulls Game!
Enter a number:
>>> 1234
2 cows, 0 bull
>>> 1256
1 cow, 0 bull
...

21. Chip defect

k defects are randomly distributed amongst *n* integrated-circuit chips produced by a factory (any number of defects may be found on a chip and each defect is independent of the other defects). Let p(*k*,*n*) represent the probability that there is a chip with at least 3 defects. For instance p(3,7) ≈ 0.0204081633.

Find p(20 000, 1 000 000) and give your answer rounded to 10 decimal places in the form 0.abcdefghij

REFERENCE BOOKS

- *Swaroop C H* ,A Byte of Python, http://python.swaroopch.com/
- *David Beazley* , Python Cookbook, 3rd edition, O'Reilly Media http://chimera.labs.oreilly.com/books/1230000000393/
- *Mark Pilgrim* ,Dive Into Python 3,

http://www.diveinto.org/python3/

OTHER REFERENCES

- Project Euler - https://projecteuler.net/

IT221 **COMPUTER NETWORKS** **CREDITS: 3**

Instruction: 3 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

- To provide the students with a sound theoretical and practical knowledge in computer networks.
- To analyze problems associated while connecting components for sharing information.
- To select a protocol stack for specific network.
- To select proper algorithm for the protocols. .
- To identify the parameters for real time applications in networks.
- Prepare students for easy transfer from academia to real world.

COURSE OUTCOMES:

After the completion of the course the student will

1. Be able to analyze different network architecture's and designs
2. Mathematically model various error control and routing schemes.
3. Ability to analyze the working of LAN in an organization.
4. Ability to designs network architecture for an organization.
5. Ability to design and implement a network for scalability and robustness and security.

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	3				3						3	
CO-2	3				3				2			
CO-3	3				3				2			
CO-4	3				3				2		2	
CO-5	2										3	

SYLLABUS

UNIT – 1 **(10 periods)**

Protocol Architecture: The need for a protocol architecture, A simple protocol architecture- *A three-layer model, standardized protocol architectures* , OSI- *The model, standardization within the OSI framework, service primitives and parameters, the OSI layers* , The TCP/IP protocol Architecture- *The TCP/IP layers, TCP and UDP Operation of TCP and IP, TCP/IP applications, protocol Interfaces,* **Local area networks: LAN overview:** Background, LAN protocol Architecture- *LAN standards, IEEE 802, LLC,MAC.* Bridges- *functions, protocol architecture, Fixed routing, spanning tree approach.* Layer 2 and Layer 3 switches- *hubs, layer2 and 3 switches*. **High speed LANs:** The Emergence of High-speed LANs, Ethernet- *MAC, Ethernet, fast Ethernet, gigabit, 10-gbps* TokenRing- *Operation, MAC*. **Wireless LANs:** overview, Wireless LAN Technology, IEEE802.11 Architecture and services, MAC, Physical Layer

UNIT-2 **(10 periods)**

Wide Area Networks: circuit switching and packet switching: switched communication networks, circuit-switching networks and concepts-*Space Division Switching, Time division switching,* packet switching principles-*switching technique, packet size, comparisons*. **Routing in switched networks:** Routing in circuit-switching networks, routing in packet switching networks- *Characteristics, routing strategies, Examples,* Least-cost Algorithms- *Dijikstra's Algorithm, Bellman-Ford algorithm, comparison*. **Congestion Control in Switched Data Networks:** effects of congestion- *ideal performance, practical performance,* congestion control- *Backpressure, choke packet, implicit congestion signaling, explicit congestion signaling,* traffic management- *fairness, QOS, Reservations*, congestion control in packet-switching networks

UNIT-3 **(10 periods)**

Internetworking: Internetwork protocols: Basic protocol Functions, principles of Internetworking- *requirements, Architectural Approaches,* connectionless Internetworking- *operation of connectionless internetworking scheme, Design issues,* Internet protocol- *IP services, Internet protocols, IP Addresses, ICMP,* IPV6- *IP next generation, IPv6 structure, IPv6 header, IPv6 Addresses, Hop-by-Hop options header, routing header, destination options header*. **Internetwork operation:** Multicasting, routing protocols- *Autonomous systems, approaches of routing, BGP, open shortest path first(OSPF) protocol*

UNIT- 4 **(10 periods)**

The Transport Layer: **The Transport Protocols:** connection-Oriented transport protocol mechanisms- *Reliable sequencing network service, unreliable network service,* TCP- *TCP services, TCP header format, TCP Mechanisms, TCP Implementation policy options,* TCP congestion control- *Retransmission timer management, window management,* UDP

Unit -5 **(8 periods)**

Application Layer: **Distributed Applications**: Electronic Mail-SMTP and MIME- *Simple mail transfer protocol (SMTP), multipurpose internet mail extensions(MIME).* Hypertext transfer protocol (HTTP)- *HTTP overview, messages, request message, response messages, Entities* Network management-SNMP- *network management systems,SNMPv1,SNMPv2*

TEXT BOOKS:

1. William Stallings ,"Data & Computer Communication", Pearson Education ,7th edition

REFERENCE BOOKS:

1. Forouzan, "Data communication", TATA McGraw
2. Kurose & Ross, "COMPUTER NETWORKS– A Top-down approach featuring the Internet", Pearson Education, Alberto Leon,Garciak.
3. LeonGartia, IndraWidjaja, "Communication Networks Fundamental Concepts and Key Architectures", TMH.
4. Nader F.Mir, "Computer and Communication Networks", PHI.

IT222 **INFORMATION SYSTEMS DESIGN** **CREDITS: 3**

Instruction: 3 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

The course will emphasize:

1. On performing a background work prior to begin Project development.
2. To gather information and analyze user requirements in system development
3. To apply the Process models in developing a project.
4. To translate end-user requirements into system and software requirements

COURSE OUTCOMES:

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

1. Understand the Information systems and systems design
2. Apply the knowledge of information gathering , Requirement analysis in Software Engineering
3. Will be able to identify specific components of a software design and use in Interface Designing
4. Use the knowledge of testing and estimate the software development cost

CO PO CO RELATION MATRIX

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	2	3	3	3	1			2	2	3	3	1
CO2	2	3	3	3	1			2	2	3	3	
CO3	2	3	3	3	3			2	2	3	3	
CO4	1	3	3	3	3			2	2	3	3	

SYLLABUS

UNIT I : (Text Book-1) **(10 Periods)**

Information and Management : Types of Information ,Computer based information systems ,Management Structure , Management and Information Requirements, Qualities of information **.** **(Page No 1-12)**

Examples of Information Systems: Various functions in organizations, Information Processing for a store – An overview , varieties of Information Systems**.** **(Page No 31-22)**

Information Gathering : Strategy to Gather information ,Information Sources , Methods of Searching for Information ,Interviewing Techniques, Questionnaires Other methods Case Study – Hostel Information System **(Page No 34-45)**

UNIT II : (Text Book-2) **(15 Periods)**

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths. **(Page No 33-47)**

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models. **(Page No 52-73)**

Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process. **(Page No 77-99)**

UNIT – III: (Text Book-2) **(16 Periods)**

Requirements Engineering : Requirements Engineering Tasks ,Initiating the requirements engineering process , Eliciting requirements ,developing use cases , Building the analysis model ,Negotiating requirements , validating requirements. **(Page No 176-204)**

Building analysis model : Requirement analysis , Analysis modelling approaches **(Page No 208-212)**

UNIT – IV: (Text Book-2) **(16 Periods)**

Design Engineering: Design process and Design quality, Design concepts, the design model. **(Page No 261-280)**

Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation. **(Page No 357-382)**

UNIT – V: (Text Book-2) **(7 Periods)**

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Validation testing, System testing **(Page No 387-404,406-410)**
Black-Box and White-Box testing, Basic-Path Testing, Control-Structure Testing **(Page No 423-434)**

Product metrics: Software Quality, A frame work for Product Metrics **(Page No.462-471)**

TEXT BOOKS :

1. V.Rajaraman ,Analysis and Design of Information System ,Second Edition ,PHI
2. Roger S Pressman ,Software Engineering A practitioner's Approach Sixth edition. McGrawHill International Edition.

REFERENCE BOOKS :

1. Waman S Jawadekar , Software Engineering Principles and Practice ,Tata McGrawHill

Ian Sommerville ,Software Engineering , Ninth Edition, Pearson

IT 223 **OPERATING SYSTEMS** **CREDITS: 4**
(COMMON FOR CSE & IT)

Instruction: 4 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVES:

To make students to

- Understand Functions, Services and structure of Operating Systems.
- Understand processes, threads, schedulers and explanation of CPU scheduling.
- Understand issues related to Process Synchronization and focus on principles of Deadlock and related problems
- Comprehend the mechanisms used in Memory Management and Virtual Memory.
- Understand the concepts of File System, secondary storage management and Disk Scheduling

COURSE OUTCOMES:

After completion of this course, a student will be able to

CO-1	Understand the difference between different types of modern operating systems, virtual machines and their structure of implementation and applications.
CO-2	Understand issues of scheduling, inter process communication in processes & threads and use of locks, semaphores, monitors for synchronization, issues related to threads and issues related to deadlocks
CO-3	Understand the design and management concepts along with issues and challenges of main memory, virtual memory and file system
CO-4	Understand the types of I/O management; disk scheduling, protection and security problems faced by operating systems and how to minimize these problems.

MAPPING OF COS AND POS:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3	3	3		1	2	1			2		3
CO-2	3	2	2					2				
CO-3	3	1	2					3				
CO-4	3	3	1		2		2	3		3		2

SYLLABUS

UNIT I **(14 Periods)**

INTRODUCTION TO OS

Introduction to operating systems – operating system structures – system calls – system structure – virtual machines.

PROCESS MANAGEMENT

Processes: Process concept – Process scheduling – Operations on processes –Cooperating processes – Interprocess communication. Multi threaded programming. Communication in client-server systems. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.

UNIT II **(14 Periods)**

PROCESS SCHEDULING AND SYNCHRONIZATION

CPU Scheduling: Scheduling criteria – Scheduling algorithms – Multiple-processor scheduling – Real time scheduling – Algorithm Evaluation. Process Synchronization: The critical-section problem –Synchronization hardware – Semaphores – Classic problems of synchronization – critical regions – Monitors. Deadlock: System model – Deadlock characterization –Methods for handling deadlocks – Deadlock prevention – Deadlock avoidance, Deadlock detection – Recovery from deadlock.

UNIT III **(12 Periods)**

MEMORY MANAGEMENT

Memory Management: Background – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging. Virtual Memory: Background – Demand paging – Process creation – Page replacement – Allocation of frames – Thrashing.

UNIT IV **(10 Periods)**

FILE SYSTEMS AND ITS IMPLEMENTATION

File-System Interface: File concept – Access methods – Directory structure – File system mounting – Protection. File-System Implementation : Directory implementation – Allocation methods – Free-space management – efficiency and performance – recovery – log-structured file systems.

UNIT V **(14 Periods)**

SECONDARY STORAGE STRUCTURES AND 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 access matrix, Access control, Revocation of access rights, Capability-Based systems.

CASE STUDY: THE LINUX OPERATING SYSTEM: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication

TEXT BOOK:

1. Silberschatz, Galvin, and Gagne, "Operating System Concepts", Sixth Edition, Wiley India Pvt Ltd, 2003.

REFERENCE BOOKS:

1. Andrew S. Tanenbaum, "Modern Operating Systems", Second Edition, Pearson Education, 2004.
2. Gary Nutt, "Operating Systems", Third Edition, Pearson Education, 2004.
3. Harvey M. Deitel, "Operating Systems", Third Edition, Pearson Education, 2004.

IT224 PROBABILITY STATISTICS & QUEUING THEORY CREDITS: 4

Instruction: 4 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

- The knowledge of Mathematics is necessary for a better understanding of almost all the Engineering and Science subjects. Here our intention is to make the students acquainted with the concept of basic topics from Mathematics, which they need to pursue their Engineering degree in different disciplines.

COURSE OUTCOMES:

At the end of the course student should be able to:

1. Understand the concepts of various statistical measures like mean , variance and standard deviation of a random variable.
2. Familiarize the different types probability distributions and their properties.
3. Compute simple correlation between the variables and fit straight line , parabola by the principle of least squares.
4. Analyze the statistical data and apply various small or large sample test for testing the hypothesis.
5. Learn about different Queuing models and its applications.

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	3	3									3	
CO-2	2	3									3	
CO-3	3	3									3	
CO-4	3	3									3	
CO-5	3	3									3	

SYLLABUS

UNIT I : PROBABILITY & MATHEMATICAL EXPECTATIONS (12 Periods)

Introduction to probability: Definition of Random Experiment, Events and Sample space, Definition of probability, Addition and Multiplication theorems, Conditional probability, Baye's Theorem, Simple Problems on Baye's theorem. Random Variables: Discrete and Continuous random variables, Distribution function of random variable, Properties, Probability mass function, Probability density function, Mathematical expectation, Properties of Mathematical expectations, Mean and Variance.

UNIT II : PROBABILITY DISTRIBUTION (14 Periods)

Discrete Distributions: Binomial Distribution, Mean and Standard Deviations of Binomial Distribution, Poisson distribution, Mean and Standard Deviations of Poisson Distribution, Applications. Continuous Probability Distributions: Uniform Distribution, Exponential Distribution, Normal Distribution, Properties of Normal Distribution, Importance of Normal Distribution, Area properties of Normal curve.

UNIT III : CURVE FITTING , CORRELATION AND REGRESSION (10 Periods)

Curve Fitting : Principle of Least Squares , Method of Least Squares
(Straight Line and Parabola) .
Correlation : Definition, Measures of correlation, Correlation for Bivariate Distribution, Rank correlation coefficients.
Regression : Simple linear regression, regression lines and properties.

UNIT IV : TESTING OF HYPOTHESIS (14 Periods)
Formulation of Null Hypothesis, Critical Region, Level of Significance.
Small Samples : Students t - distribution (Significance test of a sample mean, Significance test of difference between sample means),F- distribution, x^2 - test, Goodness of fit.
Large samples : Test of Significance of Large Samples – Single Proportion, Difference between two Proportions , Single mean and Difference of means.

UNIT V : QUEUEING THEORY (10 Periods)
Queue description, characteristics of a queuing model, study state solutions of M/M/1: α Model, M/M/1 ; N Model.

TEXT BOOK :

1. T.Veerarajan, Probability, Statistics and Random Processes , Tata McGraw Hill Publications.

REFERENCE BOOKS:

1. Kishore S. Trivedi ,Probability & Statistics with Reliability, Queuing and Computer Applications , Prentice Hall of India ,1999 .

IT225 COMPUTER GRAPHICS & MULTIMEDIA CREDITS: 3

Instruction: 3 Periods & 1 Tut /week Sessional Marks: 40
End- Exam : 3 Hours End-Exam-Marks:60

COURSE OBJECTIVE:

This course provides an introduction to the principles of computer graphics. In particular, the course will consider methods for modeling 3-dimensional objects and efficiently generating photorealistic renderings on color raster graphics devices. The emphasis of the course will be placed on understanding how the various elements that underlie computer graphics (algebra, geometry, algorithms and data structures, optics, and photometry) interact in the design of graphics software systems.

COURSE OUTCOMES:

1. Understand Computer graphics applications and apply algorithms to obtain output primitives..
2. Apply Geometric Transformations on multimedia.
3. Students will understand the concepts and techniques used in 3D computer graphics.
4. Apply 2D and 3D multimedia building blocks to Develop multimedia applications.

MAPPING OF COS AND POS

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3		2						3		3	
CO-2	3		3						3		3	
CO-3	3		3						2		2	
CO-4	3		2						3		3	

SYLLABUS

UNIT I : OUTPUT PRIMITIVES (10 hours)

Introduction - Line - Circle and Ellipse Drawing Algorithms – Attributes – Two-Dimensional Geometric Transformations – Two-Dimensional Clipping and Viewing.

UNIT II : THREE-DIMENSIONAL CONCEPTS (8 hours)

Three-Dimensional Object Representations – Three-Dimensional Geometric and Modeling Transformations – Three-Dimensional Viewing – Color models – Animation.

UNIT III : MULTIMEDIA SYSTEMS DESIGN (10 hours)

An Introduction – Multimedia applications – Multimedia System Architecture – Evolving technologies for Multimedia – Defining objects for Multimedia systems – Multimedia Data interface standards – Multimedia Databases.

UNIT IV : MULTIMEDIA FILE HANDLING (10 hours)

Compression & Decompression – Data & File Format standards – Multimedia I/O technologies - Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval Technologies.

UNIT V : HYPERMEDIA (10 hours)

Multimedia Authoring & User Interface – Hypermedia messaging - Mobile Messaging – Hypermedia message component – Creating Hypermedia message – Integrated multimedia message standards – Integrated Document management – Distributed Multimedia Systems.

TEXT BOOKS :

1. Donald Hearn and M.Pauline Baker, “Computer Graphics C Version”, Pearson Education, 2003. (UNIT I : Chapters 1 to 6; UNIT 2: Chapter 9 – 12, 15, 16)
2. Prabat K Andleigh and Kiran Thakrar, “Multimedia Systems and Design”, PHI, 2003. (UNIT 3 to 5)

REFERENCES :

1. Judith Jeffcoate, “Multimedia in practice technology and Applications”, PHI, 1998.
2. Foley, Vandam, Feiner, Huges, “Computer Graphics: Principles & Practice”, Pearson Education, second edition 2003.

IT226 **NETWORKING LAB** **Credits: 2**

Instruction: 3 Periods/Week
End. Exam: 3 Hours

Sessional Marks: 50
End-Exam-Marks: 50

Course Objectives:

- The objective of this lab is to introduce students to the design issues that arise in building and using networks and to give students hands on experience with building and using network services.
- The practical issues to be stressed include design and installation of LAN, network operating system, setting up a network system such as users and their permissions and rights, groups and domains, adding workstations and sharing of resources across the network

Course outcomes:

1. Understand and identify the various network infrastructure and command needed for network design and troubleshooting.
2. Understand the basic concepts and functions of Layer 1 (Hubs), Layer 2(Switches and bridges) and Layer 3 (Router).
3. Understand the building components of network design
4. Understand the basic format of known protocols such as TCP, UDP, ICMP..Etc.
5. Allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks..

MAPPING OF COS AND POS

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	3				3							
CO-2	3				3							
CO-3	3				3							
CO-4	3				3							
CO-5	3				3							

LIST OF EXPERIMENTS

I. STUDY EXPERIMENTS: (2 weeks duration)

This study experiments helps the learners to understand certain network components like Hubs, switches, routers, wireless access modems, transmission medium (coaxial cables, twisted pair cables, optical fiber) and several networking components

1. Study of specifications of latest desktops and laptops
2. Familiarization with Networking Components and devices: LAN Adapters, Hubs, Switches, routers etc.
3. Familiarization with Transmission media and Tools: Co-axial cable, UTP Cable, Crimping tool, Connectors etc.
4. Study of various LAN topologies and their creation using network devices, cables and computers
5. Study of Client Server Architecture
6. To study LAN using bus, tree, star topology
7. To study pc to pc communication using parallel port
8. To study fiber optics communication
9. To study wireless communication

II. HANDS ON EXPERIMENTS (8 weeks duration)

This set of experiments helps the learners in gaining expertise in developing and maintaining a certain network which includes setting up a LAN network and maintaining it, configuring routers, switches and firewalls using a certain Hardware components.

1. preparing straight and cross cables.
2. Study of network commands and network configuration commands
3. Implementation of file and printer sharing
4. Designing and implementing Class A, B, and C Networks
5. Subnet planning and its implementation.
6. To configure the IP address for a computer connected to LAN and to configure network parameters of a web browser for the same computer.
7. To install any one open source packet capture software like wire shark etc.
8. To configure WLAN
9. To install and configure wireless access points
10. To configure modem of a computer
11. To configure hub/switch and router
12. Configuring Network Neighborhood.
13. Configuring a router based firewall

III. PROGRAMMING EXPERIMENTS (5weeks duration)

This set of programming experiments helps the learners in simulating different routing protocols, network topologies and several layered protocols using simulators like NS2 and packet tracing software's

1. Configure a network topology using packet tracing software
2. Configure a network using Distance vector routing protocol using packet tracer software
3. Static routing using packet tracer software
4. DHCP, DNS, HTTP configuration using packet tracer software

EXPERIMENTS BEYOND THE SYLLABUS:

1. Developing a VPN network for number of 50 users
2. TCP, UDP protocol simulation using NS2

IT227 COMPUTER GRAPHICS & MULTIMEDIA LAB Credits: 2

Instruction: 3 Periods/Week Sessional Marks: 50
End. Exam: 3 Hours End-Exam-Marks: 50

COURSE OBJECTIVE:

The computer graphics and multimedia laboratory is established for the purpose of providing working area for development of computer graphics and multimedia

COURSE OUTCOME

1. Draw various types of lines and curves.
2. Create animations using various editing tools
3. Use audio, video , internet editing tools to develop multimedia applications

MAPPING OF COS AND POS:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3	3	3	3	3	-	-	-	-	3	-	3
CO-2	3	3	3	3	3	-	-	-	-	3	-	3
CO-3	3	3	3	3	3	-	-	-	-	3	-	3

LIST OF EXPERIMENTS

1. To implement Bresenham's algorithms for line, circle and ellipse drawing
2. To perform 2D Transformations such as translation, rotation, scaling, Reflection and sharing.
3. To implement Cohen–Sutherland 2D clipping and window–viewport mapping
4. To perform 3D Transformations such as translation, rotation and scaling.
5. User Interface Design & Graphics II: Create a user interface for your final project. Include 2 backgrounds and 1 button set. Aim for a cohesive look.
6. Multimedia Sound: Create 2 soundtracks and 2 EFX sounds for a previous project.
7. Procedure to create an animation to indicate a ball bouncing on steps
8. Procedure to simulate movement of a cloud.
9. Procedure to create an animation with the following features. WELCOME
   - Letters should appear one by one
   - The fill color of the text should change to a different color after the display of the full word.
10. Procedure to create an animation to represent the growing moon
11. Procedure to extract the flower only from given photographic image and organize it on a background. Selecting your own background for organization.
12. Procedure to use appropriate tool(s) from the toolbox cut the objects from 3 files (f1.jpg, f2.jpg & f3.jpg); organize them in a single file and apply feather effects.

REFERENCE BOOKS :

1. Vaughan, T. "Multimedia – Making it work (5th edition) ", McGraw-Hill.
2. Boyle, T. "Design for Multimedia Learning", Prentice-Hall, 1997.

IT228 OPERATING SYSTEMS (LINUX) LAB Credits: 2

Instruction: 3 Periods/Week Sessional Marks: 50
End. Exam: 3 Hours End-Exam-Marks: 50

COURSE OUTCOMES:

- Analyze the working of an operating system, its programming interface and file system.
- Develop algorithms for process scheduling, memory management, page replacement algorithms and disk scheduling.

MAPPING OF COS AND POS:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO-1	3	3	3		3	2	3			3		
CO-2	3	3	3		3		2	2		3		3

LIST OF EXPERIMENTS:

1. Shell Programming & AWK scripts
2. Write programs using the following system calls of LINUX operating system: Fork, exec, getpid, exit, wait, close, stat, opendir, readdir
3. Write programs using the I/O system calls of LINUX operating system (open, read, write, etc) and error reporting using errno
4. Write C programs to simulate UNIX commands like ls, grep, etc.
5. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for scheduling algorithms FCFS,SJF, PRIORITY & RR. For each of the scheduling policies, compute and print the average waiting time, average turnaround time and Gantt chart
6. Implement the Producer – Consumer problem using semaphores (using LINUX system calls).
7. Programs using pipes
8. Implement Banker's algorithm for handling deadlock
9. Implement free space management strategies such as First fit, Best fit and Worst fit
10. Implement page replacement algorithms such as FIFO, LRU
11. Implement file allocation techniques (Linked, Indexed and Contiguous)
12. Implement disk arm scheduling algorithms such as FCFS, SSTF

REFERENCE BOOKS:

1. Sumitabha Das, UNIX AND SHELL PROGRAMMING, Tata Mcgraw Hill Publishing Co Ltd
2. Yashwanth Kanetkar , UNIX shell programming, BPB publications
3. W. Richard Stevens, Stephen A.Rago , Advanced programming in the UNIX environment", 3rd Edition Pearson education.
4. Silberschatz, Galvin, and Gagne, “Operating System Concepts”, Sixth Edition, Wiley India Pvt Ltd, 2003.