Structure of S.E. (Information Technology) 2012 Course
w.e.f. Academic Year 2013 – 14

Part-I

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks
		Lect	Tuto	Pract	Thoery Paper	Thoery Online	TW	Pract	Oral	
214441	Discrete Structures	4	--	--	50	50	--	--	--	100
214442	Computer Organization	4	--	--	50	50				100
214443	Digital Electronics & Logic Design	3	--	--	50	50	--	--	--	100
214444	Fundamentals of Data Structures	4	--	--	50	50	--	--	--	100
214445	Problem Solving and Object Oriented Programming	4	--	--	50	50	--	--	--	100
214446	Digital Electronics Laboratory	--	--	4	--	--	--	50	50	100
214447	Programming Laboratory	--	--	4	--	--	50	50	--	100
214448	Communication & Language Laboratory	1	--	2	--	--	50	--	--	050
	Total	**20**	**--**	**10**	**250**	**250**	**100**	**100**	**50**	**750**
	Total of Part-I	**30 Hrs**			**750**					

Part-II

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks
		Lect	Tuto	Pract	Thoery Paper	Thoery Online	TW	Pract	Oral	
207003	Engineering Maths-III	4	01	--	50	50	25	--	--	125
214449	Computer Graphics	4	--	--	50	50	--	--	--	100
214450	Processor Architecture & Interfacing	4	--	--	50	50	--	--	--	100
214451	Data Structures & Files	4	--	--	50	50	--	--	--	100
214452	Foundations of Computer Network	3	--	--	50	50	--	--	--	100
214453	Processor Interfacing Laboratory	--	--	4	--	--	25	--	50	075
214454	Data Structure & Files Laboratory	--	--	4	--	--	25	50	--	075
214455	Computer Graphics Laboratory	--	--	2	--	--	25	50	--	075
	Total	**19**	**01**	**10**	**250**	**250**	**100**	**100**	**50**	**750**
	Total of Part-II	**30 Hrs**			**750**					

Semester I

214441: DISCRETE STRUCTURES

Teaching Scheme
Lectures: 4 Hrs/week

Examination Scheme
Theory: *50 Marks*
Online: *50 Marks*

***Prerequisite*:** Basic Mathematics
***Course objectives*:**
- Use appropriate set, function, or relation models to analyze practical examples, interpret the associated operations and terminology in context.
- Determine number of logical possibilities and probability of events
- Learn logic and proof techniques to expand mathematical maturity
- Formulate problems precisely, solve the problems, apply formal proof techniques, and explain their reasoning clearly.
- Illustrate by example basic terminology and model problems in computer engineering using graphs and trees.

Course Outcomes
By the end of the course, students should be able to
1) Formulate problems precisely and solve the problems
2) Apply formal proof techniques, and explain their reasoning clearly.
3) Illustrate by example, basic terminology and model problems in computer engineering using graphs and trees
4) Use graph algorithms for suitable applications

Unit I: Sets and Propositions   (8 Hrs)
Sets, Combination of sets, Venn Diagrams, Finite and Infinite sets, Un-countably infinite sets, Principle of inclusion and exclusion, multisets.
Propositions, Conditional Propositions, Logical Connectivity, Propositional calculus, Universal and Existential Quantifiers, Normal forms, methods of proofs, Mathematical Induction

Unit II: Relations and Functions   (10 Hrs)
Properties of Binary Relations, Closure of relations, Warshall's algorithm, Equivalence Relations and partitions, Partial ordering relations and lattices, Chains and Anti chains.
Functions, Composition of functions, Invertible functions, Pigeonhole Principle, Discrete Numeric functions and Generating functions, Job scheduling Problem.
Recurrence Relations
Recurrence Relation, Linear Recurrence Relations With constant Coefficients, Homogeneous Solutions, Total solutions, solutions by the method of generating functions

Unit III: Groups and Rings   (8 Hrs)
Algebraic Systems, Semi Groups, Groups, Monoid, Abelian Groups, Subgroups, Permutation Groups, Codes and Group codes, Isomorphism and Automorphisms, Homomorphism and Normal Subgroups, Ring, Integral Domain, Field, Ring Homomorphism, Polynomial Rings and Cyclic Codes.
Unit IV: Graph Theory   (8 Hrs)
Basic terminology, representation of graph in computer memory, multi graphs and weighted graphs, Subgraph, Isomorphic graph, Complete ,regular and bipartite graphs, operation on graph, paths and circuits, Hamiltonian and Euler paths and circuits, shortest path in weighted

graph(Dijkstra's algorithm), factors of a graph, planer graph and Travelling salesman problem, Graph coloring.

Unit V: Trees   **(8 Hrs)**
Trees, rooted trees, path length in rooted trees, prefix codes and optimal prefix codes,, binary search trees, tree traversals, spanning trees, Fundamental cicuits and and cut set, minimal spanning trees, Kruskal's and Prim's algorithms for minimal spanning tree, The Max flow –Min cut theorem (transport network).

Unit VI: Permutations, Combinations and Discrete Probability   (8 Hrs)
Permutations and Combinations: rule of sum and product, Permutations, Combinations, Algorithms for generation of Permutations and Combinations. Discrete Probability, Conditional Probability, Bayes' Theorem, Information and Mutual Information

Text Books:
1. C. L. Liu and D. P. Mohapatra, "Elements of Discrete Mathematics", SiE Edition,TataMcGraw-Hill, 2008, ISBN 10:0-07-066913-9
2. R. Johnsonbaugh, "Discrete Mathematics", 5th Edition, Pearson Education, 2001 ISBN 81 – 7808 – 279 - 9 (Recommended for Unit I and Unit II)

Reference Books:
1. N. Biggs, "Discrete Mathematics", 3rd Edition, Oxford University Press, ISBN 0 –19 – 850717 – 8
2. J.K Sharma, "Discrete Mathemetics", MacMillan Publishers, 3rd edition
3. Kenneth H. Rosen, "Discrete Mathematics and its Applications", 6th edition, McGraw-Hill, 2007. ISBN 978-0-07-288008-3
4. E. Goodaire and M. Parmenter, "Discrete Mathematics with Graph Theory", 2nd edition, Pearson Education, 2003 ISBN 81 – 7808 – 827 – 4
5. Semyour Lipschutz & Marc Lipson, " Discrete Mathematics", McGraw-Hill, 3rd Special Indian Edition, ISBN-13 : 978-0-07-060174-1
6. B. Kolman, R. Busby and S. Ross, "Discrete Mathematical Structures", 4th Edition,Pearson Education, 2002, ISBN 81-7808-556-9
7. N. Deo, "Graph Theory with application to Engineering and Computer Science",Prentice Hall of India, 1990, 0 – 87692 – 145 – 4
8. Eric Gossett, "Discrete Mathematics with proof", 2nd edition, Wiley Student Edition, ISBN: 978-81-265-2758-8

214442: COMPUTER ORGANIZATION

Teaching Scheme
Lectures: 4 hrs / week

Examination scheme
Theory: *50 Marks*
Online: *50 Marks*

Learning Objectives
- To understand the structure, function and characteristics of computer systems
- To understand the design of the various functional units of digital computers
- To learn programming construct of 8086 assembly language.

Unit I	**Computer Evolution & Arithmetic**	**(8 Hrs)**

A Brief History of computers, Von Neumann Architecture, Harvard architecture, Bus Interconnection, Scalar Data Types, Fixed and Floating point numbers, Booths algorithm for multiplication and its Hardware Implementation, Division: Restoring and Non Restoring algorithms, IEEE standards of Floating point representations, Floating point arithmetic.

Unit II	**The Central Processing Unit**	**(8 Hrs)**

Machine Instruction characteristics, types of operands, types of operations, Addressing modes, Instruction formats, Instruction types, Processor organization.

8086 microprocessor: Features, Block diagram and pin configuration, Max/min mode, Instruction cycles, Read Write cycles.

Unit III	**Basics of 8086 programming**	**(8 Hrs)**

Programmers model of 8086, Memory segmentation, Memory organization, 8086 addressing modes, Instruction pipelining, Instruction set, 8086 Interrupt structure.

Unit IV	**The Control Unit**	**(8 Hrs)**

Single Bus Organization, Control Unit Operations: Instruction sequencing, Micro operations and Register Transfer.

Hardwired Control: Design methods – State table and classical method, Design Examples - Multiplier Control unit.

Micro-programmed Control: Basic concepts, Microinstructions and micro-program sequencing.

Unit V	**Memory Organization**	**(8 Hrs)**

Characteristics of memory systems, Internal and External Memory. Types of advanced RAM memories: SDRAM, DDR series.

High-Speed Memories: Cache Memory, Organization and Mapping Techniques, Replacement Algorithms, Cache Coherence, MESI protocol.

Virtual Memory: Main Memory allocation, Segmentation, Paging, Address Translation Virtual to Physical.

Secondary Storage: DAT, RAID, Optical memory**:** CDROM, DVD, Blue Ray Disk.

Unit VI	**I/O Organization**	**(8 Hrs)**

Input/Output Systems, Programmed I/O, Interrupt Driven I/O, and Direct Memory Access (DMA): 8237 features and block diagram. Serial I/O USART 8251 features and block diagram, Synchronous,/Asynchronous buses and standard Interfaces - PCI, SCSI, USB, CAN, and SPI.

IC- 8255 (PPI): Features, block diagram and operating modes.

Text Books:

1. W. Stallings, "Computer Organization and Architecture: Designing for performance", 6th Edition, Prentice Hall of India, 2003, ISBN 81 – 203 – 2962 – 7.
2. C. Hamacher, V. Zvonko, S. Zaky, "Computer Organization", McGraw Hill, 2002, 5th edition, ISBN 007-120411-3.
3. Douglas Hall, "Microprocessor and Interfacing", Revised second edition, TMH, ISBN: 9780070601673.

Reference Books:

1. J. Hays, "Computer Architecture and Organization", 2nd Edition, McGraw-Hill, 1988 ISBN 0 – 07 – 100479 – 3.
2. W. Stallings William, "Computer Organization and Architecture: Principles of Structure and Function", 2nd Ed, Maxwell Macmillan Editions, 1990 ISBN 0 – 02 – 946297 – 5.
3. A. Tanenbaum, "Structured Computer Organization", 4th Ed, Prentice Hall of India, 1991 ISBN 81 – 203 – 1553 – 7.
4. G. George, "Computer Organization: Hardware and Software", 2nd Edition, Prentice Hall of India, 1986.
5. D. Paterson, J. Hennesy, "Computer Organization and Design: The Hardware Software Interface", 2nd Edition, Morgan Kauffman, 2000 ISB.

214443: DIGITAL ELECTRONICS AND LOGIC DESIGN

Teaching Scheme	**Examination scheme**

Teaching Scheme
Lectures: 3 hrs / week

Examination scheme
Theory Paper: *50 Marks*
Online Theory: *50 Marks*

Prerequisites: Basic Electronics Engineering
Course Objectives
    1. To learn and understand basic digital design techniques.
    2. To learn and understand design and construction of combinational and sequential circuits.
    3. To introduce digital logic design software such as VHDL Programming.
Course Outcome
    1) Solve K-MAPs and apply Boolean Algebra
    2) Use necessary A.C, D.C. and Loading characteristics and functioning while designing
        with digital gates
    3) Identify the Digital Circuits, Input/Outputs to replace by FPGA

Unit-I: Number System and Logic families   (08 Hrs)
Introduction to digital electronics & Boolean algebra.
Number Systems - Binary, Octal, Hexadecimal and their conversions.
Binary Arithmetic's - Signed number representations, 2's complement arithmetic.
Codes: BCD, Excess-3, Gray code, Binary Code and their conversion, switching characteristics of BJT & FET, IC Characteristics.
TTL: Standard TTL characteristics, Operation of TTL NAND gate, Subfamilies, Configurations-Active pull-up, Wired AND, totem pole, open collector.
CMOS: Standard TTL characteristics, operation of CMOS NAND, Subfamilies, CMOS configurations Wired Logic, Open drain outputs.
Comparison of TTL & CMOS, Interfacing: TTL to CMOS and CMOS to TTL

Unit-II: Combinational Logic Design   (07 Hrs)
Logic function representation - truth-table, SOP form, POS form, Simplification of logical functions (only up to 4 variables) with K-Maps
CLC design using SSI chips – Code converters, Half- Adder, Full Adder, Half Subtractor, Full Subtractor, n bit Binary adder, look ahead carry generator.
Introduction to MSI functions & chips - Multiplexers (IC 74153), Decoder / Demultiplexer (IC 74138), Encoder (IC 74147), Binary adder (IC 7483).
CLC design using MSI chips – BCD & Excess 3 adder & subtractor using IC 7483, Implementation of logic functions using IC 74153 & 74138.

Unit-III: Sequential Logic   (07 Hrs)
Introduction to sequential circuits. Difference between combinational circuits and sequential circuits, memory element - latch, flip- flops : Design, truth table, excitation table of SR, JK, D, T flip flops. Study of flip flops with asynchronous and synchronous Preset & Clear, Master Slave configuration, conversion from one type to another type of flip flop. Study of flip flop ICs - 7473, 7474, 7476.
Application of flip-flops – Bounce elimination switch, Counters- asynchronous, synchronous and modulo counters study of modulus n counter ICs- 7490, 74191 & their applications to implement mod counters.

Unit-IV: Sequential Logic Design   (07 Hrs)

Registers- Buffer register, shift register types - SISO, SIPO, PISO & PIPO, applications of shift registers - ring counter, twisted ring counter, study of universal shift register IC – 74194, sequence generators using counters & shift register, Pseudo Random Binary Sequence Generator, sequence detector using Moore & Mealy model.

Algorithmic State Machines- ASM notations, charts (eg- counters, washing machine, lift controller, vending machine), design using multiplexer controller method (eg- counters).

Unit-V: Programmable Logic Devices & Introduction to HDL (06 Hrs)

Introduction to PLD's:- ROM, PAL,PLA, Design of 4 variable SOP using PLDs, Basic architecture of SPLD and CPLD, Study of CPLD architecture XC9572, Basic architecture of FPGA, CPLD Design flow (Basic Concept of Simulation and Synthesis)

Introduction to HDL – Necessity, Characteristics & Types.

Unit-V: VHDL Programming (06 Hrs)

Introduction to VHDL - Library, Package, Entity, Architecture, Data Objects (Variable, signal & constant), Data Types (scalar, composite array type & predefined data types, Attributes (necessity and use. 'event attribute)

VHDL Modeling styles – Dataflow, behavioral & structural

VHDL statements - Concurrent Statements (With..Select, When..Else), Sequential Statements (if..else, case)

VHDL design Examples - Multiplexer, binary adder, counter, shift register.

Text Books
1. R. Jain, "Modern Digital Electronics", 3rd Edition, Tata McGraw-Hill, 2003, ISBN 0 – 07 – 049492 – 4
2. Stephen Brown, Zvonko Vranesic " Fundamentals of Digital Logic with VHDL Design" McGraw-Hill

Reference Books
1. John Yarbrough, "Digital Logic applications and Design" Thomson
2. Flyod "Digital Principles", Pearson Education
3. Malvino, D.Leach " Digital Principles and Applications", 5th edition, Tata McGraw Hill
4. J. Bhaskar "VHDL Primer" 3rd Edition, Pearson Education

214444: FUNDAMENTALS OF DATA STRUCTURES

Teaching Scheme
Lectures: *4 hrs / week*

Examination scheme
Theory: *50 Marks*
Online*: 50 Marks*

Prerequisite: Fundamental knowledge of 'C' and basics of algorithms

Course Objectives
1. To learn C language constructs and pointers in depth.
2. To learn algorithm development and analysis of algorithms.
3. To learn linear data structures and their applications
4. To learn different searching and sorting techniques

Course Outcomes
1. Student will be able to apply appropriate constructs of C language, coding standards for application development
2. Students will be able to selection and use appropriate data structures for problem solving and programming
3. Students will be able to use algorithmic foundations for solving problems and programming
4. Students will be able to select appropriate searching and/or sorting techniques for application development

Unit I: Introduction to C   (6 Hrs)
Operators ,control structures, enumeration ,structure , union, macros , arrays ,functions and parameter passing in C ,scope rules , string manipulation , matrix operations.

Unit II: Pointers in C and file handling   (8 Hrs)
Pointer, pointer to pointer ,pointer to single and multidimensional arrays, array of pointers , string and structure manipulation using pointers , pointer to functions. Pointer to file structure and basic operations on file, functions used for text and binary file handling in C.

Unit III: Introduction to Data structures & Analysis of Algorithms   (7 Hrs)
Introduction to Data Structures: Concept of data, Data object, Data structure, Abstract Data Types, realization of ADT in 'C'. Concept of Primitive and non-primitive, linear and Non-linear, static and dynamic, persistent and ephemeral data structures. Analysis of algorithm: frequency count and its importance in analysis of an algorithm, Time complexity & Space complexity of an algorithm, Big 'O', 'Ω' and 'θ' notations, Best, Worst and Average case analysis of an algorithm.

Unit IV: Searching and sorting techniques   (7 Hrs)
Need of searching and sorting, Concept of internal and external sorting, sort stability.
Searching methods: Linear and binary search algorithms their comparison and complexity analysis
Sorting methods: Bubble, selection, insertion, merge, quick, bucket sort and their time and space complexity analysis

Unit V: Linear data structures using sequential organization   (8 Hrs)
Concept of sequential organization, Concept of Linear data structures, Concept of ordered list, Multidimensional arrays and their storage representation: row major and column major form and address calculation. Representation of sparse matrix using arrays, algorithms for sparse matrix addition, simple and fast transpose, polynomial representation using arrays. Analysis of these algorithms. Concept of stack and use of it in recursion, Recursive algorithms e.g. Factorial, Fibonacci series, etc.

Unit VI: Linear data structures using linked organization   **(8 Hrs)**

Concept of linked organization, singly linked list, doubly linked list, circular linked list. Linked list as an ADT. Representation and manipulations of polynomials using linked lists, comparison of a sequential and linked memory organization, concept of Generalized Linked List, representation polynomial using GLL.

Text Books

1. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C", Cenage Learning, SBN 9788131503140.
2. G. A.V, PAI , "Data structures and Algorithms ", Mc Graw Hill, ISBN -13: 978-0-07-066726-6
3. Yashwant Kanetkar, "Let us C" and "Pointers in C" , BPB Publication

References Books

 1. R S Bichkar, "Programming with C", University Press, ISBN 978-81-7371-771-0
 2. Dennis Ritchie, Kernighan, "The C Programming Language", Prentice Hall
 3. Treamblay, Sorenson,"An introduction to data structures with applications",Tata McGraw Hill, Second Edition
 4. Seymour Lipschutz, "Data structures with C", Schaum's Publication
5. E. Horowitz, S. Sahani, S. Anderson-Freed "Fundamentals of Data Structures in C", Universities Press, 2008 ,
 6. Aaron Tanenbaum, "Data Structures using C", Pearson Education

214445: Problem Solving and Object Oriented Programming Concepts

Teaching scheme: **Examination Scheme:**

Lectures: 4 Hrs/Week   Theory: 50 Marks

                                                         Online: 50 marks

Learning Objectives:-

      1) Breaking down large problem in small problems.

      2) Using different approaches to divide this problem in small problems.

      3) Solving these small problems systematically by using different problem solving techniques.

      4) Reassembling solutions of all these problems to create solution of main problem.

      5) Understanding basics of Object Oriented Programming.

      6) Designing a real life problem using OO concepts and techniques

Course Outcomes:-

   After studying this subject students should be able to

    1) Model a real life problem into computer programming.

    2) Break the problem, solve it and reassemble it to get solution of original problem.

    3) Apply the all fundamentals of Object Oriented approach to solve a given problem.

    4) Should be able to solve a real life problem by using OO approach.

UNIT I: Problem Solving Concepts   (7 Hrs)

General Problem Solving Concepts-Types of problems, problems solving with computers, difficulties with problem solving, Problem Solving Aspects, Problem Solving Concepts for computer- constants and variables, data types, functions, operators, expressions and equations, Programming Concepts – communicating with computers, organizing the problem, using the tools, testing the solution, coding the program, Top down design

UNIT II: Problem solving with logic Structures   (6 Hrs)

Programming Structure - modules and their functions, cohesion & Coupling, Local and global variable, parameters, return values, variable names and data dictionaries, four logic structures. Problem solving with sequential logic structure - The sequential logic structure, solution development. Problem Solving with Decisions – decision logic structure, multiple if/then/else instructions, straight-through logic, positive logic, negative logic, logic conversion, decision tables. Problem solving with loops and case logic structures

UNIT III: Array and Text processing   (8 Hrs)

Processing Array - One dimensional, multidimensional arrays, table lookup technique, the pointer technique, Array Techniques - Array order reversals, array counting, and finding maximum number in a set, Partitioning of array, finding smallest element, searching an array for a range. Text processing Technique -Text Line Length Adjustment, Left and right justification of text, keyword searching in text, text line editing, Pattern searching -linear pattern search, sub linear pattern search

UNIT IV: Foundations of Object Oriented Programming   (8 Hrs)

Introduction to procedural, modular, object-oriented and generic programming techniques, Limitations of procedural programming, Need of object-oriented programming, fundamentals of object-oriented programming: objects, classes, data members, methods, messages, data encapsulation, data abstraction and information hiding, inheritance, polymorphism,

++: Extensions to C - Variable declarations, global scope, 'const', reference variables, comments, default parameters, function prototypes, function overloading, inline functions, default and constant arguments, 'cin', 'cout', formatting and I/O manipulators, new and delete operators Defining a class, data members and methods, public, private and protected members, inline member functions, static data members, static member functions, 'this' pointer, constructors, destructors, friend function, dynamic memory allocation, array of objects, pointers and classes, class as ADTs and code reuse

UNIT V: Overloading and Inheritance   (9 Hrs)
Introduction, Need of operator overloading, overloading the assignment, binary and unary operators, overloading using friends, rules for operator overloading, type conversions Concept and need, single inheritance, base and derived classes, friend classes, types of inheritance, hybrid inheritance, member access control, static class, multiple inheritance, ambiguity, virtual base class, polymorphism, virtual functions, pure virtual functions, abstract base class, virtual destructors, early and late binding, container classes

UNIT VI: Templates and Exception Handling   (10 Hrs)
Introduction, Templates: Function template and class template, function overloading vs. function templates, member function templates and template arguments, Introduction to Generic Programming: Introduction to Standard Template Library (STL), containers, iterators and algorithms, study of container template classes for vectors and stacks and related algorithms, **Namespaces:** Introduction, Rules of namespaces **Exception Handling:** Introduction, syntax for exception handling code: try-catchthrow, Multiple Exceptions, Exceptions with arguments, Introduction to RTTI **Managing Console I/O Operations:** Introduction, C++ streams, stream classes, unformatted I/O, formatted I/O and I/O manipulators

Text book
1. "How to Solve it by Computer", R G Dromey ISBN 978-81-317-0562-9 2. "Problem Solving and Programming Concepts", Maureen Spankle, ISBN81-317-0711-3
2. Robert Lafore, "Object-Oriented Programming in C++", SAMS Techmedia

Reference book
1. "Programming Logic and Design" by Joyce Farrell ISBN 978-81-315-0263-1 2.
2. E. Balaguruswamy, "Object-oriented Programming with C++" Tata McGraw Hill, 5th edition

214446: DIGITAL ELECTRONICS LABORATORY

Teaching Scheme   **Examination scheme**
Practical: 4 hrs / week   Practical: 50 Marks
                                                      Oral: 50 Marks

A.   Combinational logic design:
1.   TTL & CMOS Characteristics.
2.   Design (truth table, K-map) and implementation of 4 bit Code converters.
     i.   Binary to gray and vice versa
     ii.   BCD to Excess-3 and vice versa
3.   Design (truth table, K-map) and implementation of 4 bit BCD & Excess 3 Adder using IC7483.
4.   Implementation of logic functions using multiplexer IC 74153 & decoder IC 74138.
     (Verification, cascading & logic function implementation)

B. Sequential logic design

1. Design (State diagram, state table & K map) and implementation of 4 bit Up, Down, Controlled Up/Down asynchronous counter using master slave JK flip-flop IC 7476.
2. Design (State diagram, state table & K map) and implementation of 3 bit Up, Down, Controlled Up/Down Synchronous counter using master slave JK flip-flop IC 7476.
3. Design and implementation of Modulo 'n' counter with IC 7490 and IC 74191.
4. Design (State diagram, state table, K map, Bush table & Bush diagram) and implementation of Sequence Generator (with & without bushing) using master slave JK flip-flop IC 7476 & shift register IC 74194.

C.   VHDL Programming:

   Simulation & Implementation of

1.   4:1 multiplexer using data flow & structural modeling.
2.   Full adder using behavioral & structural modeling.
3.   3 bit controlled up / down synchronous counter with preset & clear.
4.   3 bit bidirectional shift register.

Instructor will frame assignments based on the suggested assignments as given above. Students will submit the term work in the form of journal consisting of 12 assignments listed above.

Practical examination will be based on practical assignments and questions will be asked to judge the understanding of assignments performed at the time of examination.

Note   - Instructor should take care that datasheets of all the required ICs are available in the laboratory & students are verifying the functionality of ICs being used.

214447: PROGRAMMING LABORATORY

Teaching Scheme
Practical: *4 hrs / week / batch*

Examination scheme
Practical: *50 Marks*
Term work: *50 Marks*

- These laboratory assignments are based on Fundamentals of Data Structures
- These assignments are to be implemented in GCC or tools like eclipse under Unix/Linux/any open source operating system platform

List of experiments:
1. Represent sets using one dimensional arrays and implement functions to perform
   i) union ii) intersection, iii) difference, iv) symmetric difference of two sets.
2. Represent matrix using two dimensional arrays and perform following operations:
   i) Addition using pointers ii) multiplication without pointers iii) transpose using pointers
3. Implement following operations on string with and without   using pointers
   i) Length ii) Palindrome iii) String comparison iv) Copy v) Reverse vi) Substring
4. Create a Database using array of structures and perform following operations on it:
   i) Create Database ii) Display Database iii) Add record iv) Search a record   v) Modify a records.   vi) Delete a record.
5.   a) Sort the set of strings in ascending order using Bubble sort and descending order by using Selection sort.
   b) Search for particular string using binary search.
6. a) Write C program to implement TYPE and COPY commands of DOS using command line arguments.
   b) Find out number of characters, words, spaces and sentences form a file and write result in another file.
7. Implement Quick Sort recursively to sort the given list of numbers/records. Display pivot position and its corresponding list in each pass.
8. Represent sparse matrix using two dimensional array or structure and perform simple and fast transpose
9. Implement a singly linked list with following options
   i) Insertion of a node at any location ii) Deletion of a node from any location iii) display a list
   iv) Display in reverse   v) Reverse the list without using additional data structure.
10. Implement polynomial using CLL and perform i) Addition of Polynomials ii) Multiplication of polynomials and iii) Evaluation of polynomial.
11. Implement any database using doubly linked list with following options
   i) Insert a record ii) delete a record iii) modify a record iii) Display list forward, d) Display list backward

Note:
1. For all programs implementations students are expected to use meaningful identifiers, proper indentation, use of functions, minimal use of global variables and writing time complexity using any one notation is mandatory.
2. Student should submit term work in the form of a journal based on the above assignments. Practical examination will be based on the term work. Questions will be asked during the examination to judge the understanding of the practical performed in the examination. Candidate is expected to know the theory involved in the experiment.

Teaching scheme	Examination scheme
Theory Lectures: *1 hr/week*	Term Work: *50 Marks*
Practical: *2 hrs/week*	

Objectives
- Improve students' overall linguistic & communicative competence in English
- Enhance their pronunciation, vocabulary and LSRW skills
- Foster their confidence in public speaking and group communication skills

Course Outcomes
- Provides an ability to understand, analyze and interpret the essentiality of grammar and its proper usage.
- Build the students' vocabulary by means of communication via web, direct communication and indirect communication.
- Improves Students' Pronunciation skills and understanding between various phonetic sounds during communication.
- Understanding the various rules and means of written communication.
- Effective communication with active listening, facing problems while communication and how to overcome it.

Overview
The course has been designed for the students of second year Information Technology students for enhancing their linguistic and communicative competence. It attempts to give them exposure to the essential linguistic and communicate skills by focusing upon the key areas of immediate significance. Students will be given a sound theoretical knowledge through lectures about the fundamental concepts in the English language & communication such as grammar, vocabulary, pronunciation and LSRW skills. At the same time adequate practical exposure to these skills will be provided through laboratory sessions. The course aims at striking a fine balance between theory and practice to ensure the all round improvement of students in these skills. Students will be able to improve their command over communicative English which will enable them not only to enhance their academic performance but also it will contribute to their growth as engineering professionals.

Teaching Methodology in the Language Laboratory
1. Direct Method – Use of English for communication between the teacher and students. Teachers must emphasize on the use of English in the lab. All the instructions and interactions must take place in English.
2. Theory lectures should also be interactive and the teacher should encourage students' participation in the classroom sessions.
3. Laboratory sessions should be activity based and should be conducted in groups and pairs. Guidelines for conducting laboratory sessions have been given below each activity.

Unit I: ESSENTIAL GRAMMAR   (5 hrs)
Tenses: Basic forms and use, sentence formation (general & Technical), Common errors, Parts

of speech through context, Direct and reported speech structures and voices.

Unit II: VOCABULARY ENRICHMENT (5 hrs)

Exposure to words from General Service List (GSL) by West, Academic word list (AWL) by Averil Coxhead (2000) and specific technical terms related to the field of Information technology. Phrases, idioms, significant abbreviations, formal (business) vocabulary. Use of web-based applications for academic vocabulary building.

Unit III: PHONETICS (5 hrs)

Introduction to the 44 phonemes of English i.e. 20 vowels & 24 consonants, phonemic transcriptions of words and sentences, stress & intonation, voice modulation, exercises on pronunciation, use of software and web-based applications for exercises on pronunciation.

Unit IV: WRITTEN COMMUNICATION (5 hrs)

Letter Writing - Business letters, Application letters, Covering letters, Report Writing - Academic and Business reports, Technical Project writing, Job application letter and Resume writing

Unit V: FUNDAMENTALS OF EFFECTIVE COMMUNICATION (4 hrs)

Concept of communication, types of communication, Barriers to communication, communication process, essentials of effective communication, Active Listening.

1. **Public Speaking**
   Any one of the following activities may be conducted :
   a. **Prepared speech** (topics are given in advance, students get 10 minutes to prepare the speech and 5 minutes to deliver.
   b. **Extempore speech (**students deliver speeches sontaneously for 5 minutes each on a given topic)
   c. **Story telling (Each student narrates a fictional or real life story for 5 minutes each)**
   d. **Oral review** (Each student orally presents a review on a story or a book read by them)

2. **Power-point Presentations**
   Students should make a presentation on any informative topic of their choice. The topic may be technical or non-technical
3. **Formal Group Discussion**
   Each batch is divided into two groups of 12 to 14 students each. Two rounds of a GD for each group should be conducted and teacher should give them feedback.
4. **English Language Proficiency Test**
   The teacher should conduct a 50 mark English proficiency test in the lab and discuss the answers with explanation and more illustrations.
5. **Mock Meetings**
   In order to enhance students' formal oral communication, mock meetings can be conducted. Teacher should give a topic for the meeting and teach students how a notice and agenda for a meeting is prepared. Students will participate in the meeting assuming the roles assigned by the teacher. After the meeting, teacher should guide students on how minutes of meeting are recorded.
6. **Reading and Listening skills**
   The batch can be divided into pairs. Each pair will be given an article (any topic) by the teacher. Each pair would come on the stage and read aloud the article one by one. After reading by each pair, the other students will be asked questions on the

article by the readers. Students will get marks for correct answers and also for their reading skills. This will evaluate their reading and listening skills. The teacher should give them guidelines on improving their reading and listening skills.

7. **Pronunciation through software or web-based applications**
   Teachers should make use of software and web-based applications for giving exercises on pronunciation to students.

8. **Vocabulary exercises through web-based applications**
   Teachers should make use of software and web-based applications for giving exercises on vocabulary to students.

9. **Letter, Report & review writing**
   Each student will write one formal letter, one report and a review on the topics given by the teacher.

10. **Grammar exercises through web-based applications**
    Teachers should make use of software and web-based applications for giving exercises on grammar to students.

The term work shall consist of 10 activities carrying 10 marks each. The total marks earned by the students out of 100 will be scaled down to 50. The online exam and term work marks will be further scaled down to 50. Students will have to submit journals or files containing record of each activity performed in laboratory, at the term end.

References
1. Rutherford A. J. : Communication skills for Technical Communication, Pearson Education
2. Meenakshi Raman, Sangeeta Sharma : Technical Communication – Principles and practice, Oxford
3. Kishna Mohan, "Developing Communications Skills", MacMillan Publishers, 2nd Edition
4. M.S. Rao : Strategies for improving your business communication, SPD
5. Preeti Shirodkar : Knowing your words' worth, SPD
6. Murphy : Essential English Grammar, Cambridge
7. Dutt et.al. : A course in Communication Skills, Foundation
8. Ibbotson: Cambridge English for Engineering, Cambridge
9. Turk: Effective Speaking, Taylor & Francis
10. Patnaik: Group Discussion and Interview Skills, Foundation
11. Peter Roach: English Phonetics and Phonology, 4th Edition, Cambridge
12. Mishra: A companion to communication skills in English, PHI
13. Sweeney: English for business communication, 2nd edition, Cambridge
14. Lynch: listening, Cambridge
15. Malcom Goodale: Professional Presentations, Cambridge
16. Ham-Lyons & Heasley: Writing, 2nd Edition, Cambridge
17. Jennings: Communication basics, Cengage Learning
18. ASTD: 10 steps to successful meetings, Cengage Learning

ESL Sites (Web-based applications) for vocabulary learning
1. http://www.nottingham.ac.uk/%7Ealzsh3/acvocab/awlhighlighter.htm
2. http://www.visuwords.com/
3. http://www.vocabulary.com/
4. http://www.uefap.com/vocab/exercise/exercise.htm
5. www.englishvocabularyexercises.com
6. www.eslgold.com
7. http://www.webcorp.org.uk/
8. http://www.soundkeepers.com/GMAT/

Semester II

UNIVERSITY OF PUNE
For Computer Engineering & Information Technology (Sem II)
207003 ENGINEERING MATHEMATICS – III (2012 Course)

Teaching Scheme:
Lectures – 4 Hrs./Week
Tutorials – 1 Hr./Week

Examination Scheme:
Paper – 50 Marks (2 Hrs.)
Online – 50 Marks
Term work: 25 Marks

Section I

Unit I: Linear Differential Equations (LDE) and Applications   (09 Hours)
LDE of nth order with constant coefficients, Method of variation of parameters, Cauchy's & Legendre's DE, Simultaneous & Symmetric simultaneous DE. Modeling of Electrical circuits.

Unit II: Transforms   (09 Hours)
Fourier Transform (**FT**): Complex Exponential form of Fourier series, Fourier integral theorem, Sine & Cosine integrals, Fourier Transform, Fourier Sine and Cosine Transform and their inverses.
Z - Transform (**ZT**): Introduction, Definition, Standard properties, ZT of standard sequences and their inverses. Solution of difference equations.

Unit III: Statistics and Probability   (09 Hours)
Measures of central tendency, Standard deviation, Coefficient of variation, Moments, Skewness and Kurtosis, Correlation and Regression, Reliability of Regression Estimates.
Probability, Probability density function, Probability distributions: Binomial, Poisson, Normal and Hypergometric. Test of Hypothesis: Chi-Square test.

Section II

Unit IV: Vector Differential Calculus   (09 Hours)
Physical interpretation of Vector differentiation, Vector differential operator, Gradient, Divergence and Curl, Directional derivative, Solenoidal, Irrotational and Conservative fields, Scalar potential, Vector identities.

Unit V: Vector Integral Calculus and Applications   (09 Hours)
Line, Surface and Volume integrals, Work-done, Green's Lemma, Gauss's Divergence theorem, Stoke's theorem. Applications to problems in Electro-magnetic fields.

Unit VI: Complex Variables   (09 Hours)
Functions of Complex variables, Analytic functions, Cauchy-Riemann equations, Conformal mapping, Bilinear transformation, Cauchy's integral theorem, Cauchy's integral formula, Laurent's series, Residue theorem.

Text Books:
1. Advanced Engineering Mathematics, 9e, by Erwin Kreyszig (Wiley India).
2. Advanced Engineering Mathematics, 7e, by Peter V. O'Neil (Cengage Learning).

Reference Books:
1. Advanced Engineering Mathematics, 2e, by M. D. Greenberg (Pearson Education).
2. Advanced Engineering Mathematics, Wylie C.R. & Barrett L.C. (McGraw-Hill, Inc.)
3. Higher Engineering Mathematics by B. S. Grewal (Khanna Publication, Delhi).
4. Applied Mathematics (Volumes I and II) by P. N. Wartikar & J. N. Wartikar (Pune Vidyarthi Griha Prakashan, Pune).
5. Higher Engineering Mathematics by B.V. Ramana (Tata McGraw-Hill).
6. Advanced Engineering Mathematics with MATLAB, 2e, by Thomas L. Harman, James Dabney and Norman Richert (Brooks/Cole, Thomson Learning).

Tutorial and Term Work:
i)   Tutorial for the subject shall be engaged in minimum of four batches (batch size of 20 students maximum) per division.
ii) Term work shall consist of six assignments (one per each unit) based on performance and continuous internal assessment.

214449: COMPUTER GRAPHICS

Teaching Scheme   **Examination scheme**
Lectures: 4 hrs / week   Theory: *50 Marks*
                                                                        Online: *50 Marks*

Pre-requisites
1. Computer Programming and basic data structures.
2. Mathematics topics such as analytical geometry, trigonometry, linear algebra and matrices.
3. Knowledge of vector space, Matrices, Dot products and distances.

Course Learning Objectives
1. Understand the foundations of computer graphics: hardware systems, math basis, light and color.
2. Understand the complexities of modeling realistic objects through modeling complex scenes using a high-level scene description language.
3. Become acquainted with some advanced topics in computer graphics.
4. The student should gain an expanded vocabulary for discussing issues relevant to computer graphics (including both the underlying mathematics and the actual programming).
5. The student should gain an appreciation and understanding of the hardware and software utilized in constructing computer graphics applications.
6. The student should gain a comprehension of windows, clipping and view-ports in relation to images displayed on screen.
7. The student should gain an understanding of geometric, mathematical and algorithmic concepts necessary for programming computer graphics.

Course Outcomes
1) Apply mathematics and computer programming to computer graphics applications and problem solutions
2) Systematically identify, evaluate and solve complex technical and aesthetic problems
3) Be ready to contribute in a significant way to the computer graphics industry
4) Demonstrate the knowledge and technical skills to be successful in a specialized, computer-based, graphics field

Unit I   Introduction   (8 Hrs)
Basic Concepts:-
Graphics Primitives: Introduction to computer graphics, Basics of graphics systems, Raster scan & random scan displays, display processor, display file structure, algorithms and display file   interpreter. primitive operations on display file, Display devices, Interactive devices: Tablets, touch panels, mouse, joysticks, track balls, light pen etc., Data generating devices: Scanners and digitizers,
Scan conversions, lines, line segments, vectors, pixels and frame buffers, vector generation, Line drawing Algorithms : DDA , Bresenham's, Thick Line generations
Circle drawing Algorithms :- DDA, Bresenham's and Mid-Point
Character Generation: Stroke Principle, Starburst Principle, Bit map method Aliasing, and anti-aliasing techniques, (Line & Circle algorithms should be given mathematical treatment).
[Reference book : Harington, Rogers, Bakers]

Unit II   Polygons and 2D Transformations   (8 Hrs)
Polygons, types, inside test, polygon filling methods: seed fill, scan line.
2D Geometric Transformations, Basic transformations- translation, scaling, rotation, other transformations such as reflection, shearing, matrix representation and homogeneous coordinate system, Composite transformation [Reference book : Harington]

Unit III   Segments ,Windowing and Clipping   (8 hrs)
Segment: Introduction, Segment table ,segment creation ,closing, delete and renaming visibility
Windowing: Concept of window and viewport, viewing transformations,
Clipping: Line Clipping – Cohen Sutherland Out code Method, Midpoint subdivision method, Polygon Clipping : Sutherland Hodgman method, generalized clipping. [Reference book : Harington]

Unit IV   3D Transformations and Projections   (8 Hrs)
Translation, scaling, rotation, rotation about X, Y, Z and arbitrary axis reflection about XY,YZ, XZ, and arbitrary plane. Projections: Types Parallel (Oblique: Cavalier , Cabinet and orthographic :isometric, dimetric, trimetric) and Perspective (Vanishing Points – 1 point, 2point and 3 point) Mathematical treatment to be given [Reference book : Harington, Schaum's Series outlines]

Unit V   Shading, Color models & Animation   (8 Hrs)
Colors spaces: RGB, HSV, CMY, CMYK, YIQ, Color Mixing.
Shading : Halftoning, Gaurand and Phong Shadding,
Computer Animation: Animation sequences, functions & Languages, Key-frame systems, Motion Specifications. [Reference book : Harington, Baker,]

Unit VI   Curves and Fractals   (8 Hrs)
Introduction, Curve generation, Interpolation, interpolating algorithms, interpolating polygons, B-Splines and corners, Bezier curves, Fractals, fractal lines and surfaces (With complete mathematical treatment of this unit) Interactive Graphics & usage of at least two tools of computer graphics Maya, Similar tools) (Usage of tools in Lab).[Reference book : Harington,]

Text Books :
1. Steven Harrington ,"Computer Graphics A Programming approach ",Tata McGraw Hill.
2. M Paulin Baker ,"Computer Graphics", Pearson Education.
3. Zhigang Xiang, Roy Plastock " Computer Graphics " .Schaum's Series outlines
4. Er. Rajiv Chopra " Computer Graphics". S Chand & Company Ltd.

Reference Books :
1. Davis Rogers ,"Procedural Elements for Computer Graphics", Tata McGraw Hill
2. Shirley, Marschner, "Fundamentals of Computer Graphics", Third Ed, A K Peters SPD.
3. Foley and Van Dam ,"Computer Graphics: Principles and Practice", Pearson Education
4. F.S. Hill JR, "Computer Graphics Using Open GL". Pearson Education
5. Amarendra N Sinha, Arun D Udai,"Computer Graphics", Tata McGraw Hill
6. Shirley, "Computer Graphics", First Edition,Cengage lerning publication
7. D.P.Mukharjee,Debasish Jana,"Computer Graphics Algorithms and implementation", PHI Learning
8. Singh P K, Rajendra Kumar," Computer Graphics", Vikas publication.
9. C.S.Verma,"Computer Graphics", Anne Books Pvt Ltd.
10 Samuel R. Buss," 3D Computer Graphics", Cambridge University Press.

214450: PROCESSOR ARCHITECTURE AND INTERFACING

Teaching Scheme   **Examination scheme**
Lectures: 4 *hrs / week*   Theory: *50 Marks*
                                                              Online: *50 Marks*

Prerequisites: *Computer Organization*
Learning Objectives
1. To learn the architecture and assembly language programming of 80386 Microprocessor.
2. To study architecture and programming of 8051 micro-controllers

Course Outcomes

Unit I: Introduction to Assembly Language Programming and 80386 Processors   (8 Hrs)
Introduction to assembly language programming, ALP tools- Assembler, linker, loader, debugger, emulator concepts, Assembler directives, far and near procedures, macros.
80386 - Features and Architecture, Register Set, 80386 Real mode Segmentation, Instruction format & addressing modes, Instruction set

Unit II: 80386 Memory Segmentation   (8 Hrs)
Pin Description, Bus Cycles, and Bus operations – Non-pipelined & Pipelined, Initialization and configuration, 16/32 bit data transfer mechanism.
Segmentation- support registers, related instructions descriptors, memory management through segmentation, logical to linear/physical address translation, protection in segmentation, Privileged instructions.

Unit III: Protected Mode   (8 Hrs)
Paging - support registers, descriptors, linear to physical address translation, page level protection, virtual memory, entering into PM mode and returning back to RM mode, Inter-privilege level transfer using Call gates and confirming code segment.
Multitasking - Support registers, related descriptors, Task switching, I/O permission bit map.

Unit IV: Interrupts   (8 Hrs)
Virtual Mode - features, address generation, registers available, entering and leaving V86 mode.
Real and Protected mode Interrupt structure - IVT, types of exception and processing, through IDT, Halt and Shutdown , Comparison of all three modes
Introduction to duo-core processor- features and block diagram.

Unit V: Microcontroller-I   (8 Hrs)
Microcontroller 8051 Architecture, On-Chip data memory and program memory organization - Register set, Register bank, SFRs, Instruction format & addressing modes. Instruction set. External data memory and program memory & its interfacing, I/O ports programming.

Unit VI: Microcontroller-II   (8 Hrs)
Interrupts structure and Response. Timers/counters and their programming, Serial port and programming, Interrupt programming, Design of minimum system using 8051 micro-controller for various applications.

Text Books
1. James Turley, "Advanced 80386 Programming Techniques ", McGraw Hill Education, ISBN: 9780070598416
2. Kenneth Ayala ,"The 8051 Micro Controller", Delmar Cengage Learning; 3rd edition (June 1, 2004) 3rd Edition

Reference Books

1. I. Scott MacKenzie , Raphael Chung-Wei Phan "8051 Microcontroller", 4th edition, Prentice Hall
2. Tribel Singh "8088 /8086 Processor", PHI
3. Mazidi M.Gillipse J. "The 8051 Microcontroller and Embedded Systems", Pearson education,2002,ISBN-81-7808-574-7

214451: DATA STRUCTURES AND FILES

Teaching Scheme   **Examination scheme**
Lectures: *4 hrs / week*   Theory: *50 Marks*
                                                            Online: *50 Marks*

Pre-requisite
  Fundamentals of data structure, discrete structures
Course learning objectives
  1. To study data structures their implementations using OOP (C++) and applications.
  3. To learn different file organizations
  4. To study some advanced data structures such as trees, graphs, tables
Course Outcomes
  1. Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
  2. Understand different advanced abstract data type (ADT) and data structures and their implementations.
  3. Understand different algorithm design techniques (brute -force, divide and conquer, greedy, etc.) and their implementation
  4. Ability to apply and implement learned algorithm design techniques and data structures to solve problems.

Unit I: Stacks   **(8 Hrs)**
Trees Concept of stack, stack as ADT, Implementation of stack using sequential and linked organization. Concept of implicit and explicit stack, Concept of multistacks and their implementation, Applications of stack.

Unit II: Queues   **(6 Hrs)**
Concept of queues as ADT, Implementation of linear and circular queue using linked and sequential organization. Concept of multiqueues, double ended queue and priority queue. Applications of queues.

Unit III: Trees   **(10 Hrs)**
Difference in linear and non-linear data structure, Trees and binary trees-concept and terminology. Expression tree. Conversion of general tree to binary tree.Binary tree as an ADT. Recursive and non recursive algorithms for binary tree traversals, Binary search trees, Binary search tree as ADT, Concept of threaded binary tree. Preorder and Inorder traversals of inorder threaded binary tree . Applications of trees

Unit IV: Graphs   **(8 Hrs)**
Graph as an ADT, Representation of graphs using adjacency matrix and adjacency list, Depth First Search and Breadth First Search traversal. Prim's and Kruskal's algorithms for minimum spanning tree, shortest path using Warshall's and Dijkstra's algorithm, topological sorting.

Unit V: Tables   **(8 Hrs)**
Symbol Table : Notion of Symbol Table, Concept of red and black trees, AVL Trees, OBST, Huffman's algorithm, Heap data structure, Min and Max Heap , Heap sort implementation, applications of heap

Hash tables and scattered tables: Basic concepts, hash function, characteristics of good hash function, different key-to-address transformations techniques, synonyms or collisions, collision resolution techniques- linear probing, quadratic probing, rehashing, chaining without replacement and chaining with replacement.

Unit VI: File Organization (8 Hrs)

External storage devices ,File ,File types and file organization (sequential , index sequential and Direct access), Primitive operations and implementations for each type and comparison.

Text Books
1. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C++", Cengage Learning, ISBN 9788131503140.
2. E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, 1995, ISBN 16782928

Reference Books
1. Bruno R Preiss, "Data Structures and Algorithms with object-oriented design patterns in C++", Wiley India Edition
2. G. A.V, PAI , "Data structures and Algorithms ", Mc Graw Hill, ISBN -13: 978-0-07-066726-6
3. Y. Langsam, M. Augenstin, A. Tannenbaum, "Data Structures using C and C++", 2nd Edition, Prentice Hall of India, 2002, ISBN-81-203-1177-9.
4. A. Tharp ,"File organisation and processinng", 2008 ,Willey India edition ,9788126518685
5. J. Tremblay, P. Soresan, "An introduction to data structures with Applications", 2nd edition, Tata McGraw Hill International Editions, 1984, ISBN-0-07-462471-7.
6. M. Folk, B. Zoellick, G. Riccardi, "File Structure An Object oriented approach with C++", Pearson Education, 2002, ISBN 81 - 7808 - 131 - 8.
7. M. Weiss, "Data Structures and Algorithm Analysis in C++", 2nd edition, Pearson Education, 2002, ISBN-81-7808-670-0

214452: FOUNDATION OF COMPUTER NETWORKS

Teaching Scheme   **Examination scheme**
Lectures: *3 hrs / week*   Theory: 50 *Marks*
                                                                   Online: *50 Marks*

Course Learning Objectives
   1. To learn fundamentals of data communications
   2. To learn basic Network configurations
   3. To understand the differences between data communications and telecommunications
   4. To understand basics of computer network

Course Outcomes
 At the end of this subject, students should be
   1. Explain the basic concepts of data communication
   2. Describe emerging network technologies
   3. Understand the terminology and concepts of the OSI reference model and the TCP-IP reference model.
   4. Explain the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks

Unit I: Fundamentals of Signals   **(6 Hrs)**
Analog and Digital: Analog and Digital Data, Analog and Digital Signals, Periodic and Non-periodic Signal
Periodic Analog Signals: Sine Wave, Phase, Wavelength, Time and Frequency Domains, Composite Signals Bandwidth
Digital Signals: Bit Rate, bit Length, Digital Signal as a Composite Analog Signal, Transmission of Digital Signals
Transmission Impairment: Attenuation, Distortion, Noise
Data Rate Limits: Noiseless Channel: Nyquist Bit Rate, Noisy Channel: Shannon Capacity, Using Both Limits
Performance: Bandwidth, Throughput, Latency (delay), Bandwidth-delay Product, Jitter

Unit II: Modulation and Multiplexing Techniques   **(6 Hrs)**
*Digital-to-digital Conversion***:** Line Coding, Line Coding Schemes, Block Coding, Scrambling
*Analog to digital Conversion***:** Pulse Code Modulation (PCM), Delta Modulation (DM), ADM
*Transmission modes***:** parallel transmission, serial transmission
*Digital-to-analog Conversion***:** Aspects of Digital-to-Analog Conversion, Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK), Quadrature Amplitude Modulation (QAM)
*Analog-to-analog Conversion***:** Amplitude Modulation, Frequency Modulation, Phase Modulation
Multiplexing: Frequency-Division Multiplexing (FDM), Wavelength-Division Multiplexing Synchronous Time-Division Multiplexing, Statistical Time-Division Multiplexing
*Spread Spectrum***:** Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum.

Unit III: Transmission Media and Switching   **(6 Hrs)**
Guided Media: Twisted-Pair, Coaxial and Fiber-Optic Cable,
Unguided Media: Wireless, Radio Waves, Microwaves, Infrared
IEEE standard and connectors for media.
Circuit-switched Networks: Three Phases, Efficiency, Delay, Circuit-Switched Technology in Telephone Networks

Datagram networks: Routing Table, Efficiency, Delay, Datagram Networks in the Internet
Virtual-circuit networks: Addressing, Three Phases, Efficiency, Delay in Virtual-Circuit Networks. Circuit-Switched Technology in WAN
Structure of a switch: Structure of Circuit Switches, Structure of Packet Switches
Brief introduction of Digital Subscriber Line: ADSL, HDSL, SDSL, VDSL (DMT).

Unit IV: Layer Models and Topologies   (6 Hrs)
Layered Tasks: Sender, Receiver, and Carrier, Hierarchy
The OSI Model: Layered Architecture, peer-to-peer Processes, Encapsulation of Layers in the OSI Model, TCP/IP Protocol Suite, LAN, MAN, WAN, IEEE Standards for different topologies and applications.
Addressing: Physical &logical Addresses, Port Addresses, Specific Addresses
Connecting devices: hubs, repeaters, active hubs, bridges, layer two switches, Routers, layer three switches, gateway
Backbone networks: bus backbone, star backbone.

Unit V: Error Control and Data Link Control   (6 Hrs)
Types of errors: Redundancy, detection versus correction, forward error correction versus retransmission, coding, modular arithmetic
Block coding: error detection, error correction, hamming distance, minimum hamming distance
Linear block codes: minimum distance for linear block codes, some linear block codes
Cyclic codes: cyclic redundancy check, polynomials, cyclic code analysis, advantages of cyclic codes. *Checksum*: one's complement, Internet checksum. *Framing*: fixed-size framing, variable size framing. *Flow and error control*: flow control, error control protocols. *Noiseless channels:* simplest protocol, stop-and-wait protocol. *Noisy channels*: stop-and-wait automatic repeat request, go-back-n automatic repeat request, Selective repeat automatic repeat request, piggybacking

Unit VI: Multiple Access and Ethernet   (6 Hrs)
Random access: Aloha, Carrier Sense Multiple Access (CSMA), Carrier Sense Multiple Access with Collision Detection (CSMA/CD), Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). Controlled access: reservation, polling, token passing. Channelization: Frequency Division Multiple Access (FDMA), Time-Division Multiple Access (TDMA), Code Division Multiple Access (CDMA). Ethernet: IEEE standards, data link layer, physical layer.
Standard Ethernet: MAC Sub-layer, Physical Layer, bridged Ethernet, switched Ethernet, full-duplex Ethernet. Fast Ethernet: MAC Sub-layer, Physical Layer. Gigabit Ethernet: MAC sub-layer, Physical Layer, 10 gigabit Ethernet. HDLC: configurations and transfer modes, frames, control field. Point-to-point Protocol: Framing, Transition Phases, Multiplexing, Multilink PPP.

Text Books
1. Behrouz a Forouzan, "Data Communications and Networking", McGraw Hill.
2. A. Tanenbaum, "Computer Networks", Pearson Education.

Reference Books
1. Douglas E. Comer, "Computer Networks and Internets", Pearson Education.
2. Achyut Godbole, "Data Communication Networks", Tata McGraw Hill.
3. Bruce Hartpence, "Core Network Protocols", O'REILLY.
4. P.C.Gupta, Data Communications and Computer Networks, PHI.
5. James Irvin, Data Communications and Networks an Engineering Approach, Wiley.

Teaching Scheme	**Examination scheme**

Practical: *4 hrs / week*   Term Work: *25 Marks*
    Oral: *50 Marks*

A. Microprocessor Programming

1. Write ALP to convert 4-digit Hex number into its equivalent BCD number and 5-digit BCD number into its equivalent HEX number. Make your program user friendly to accept the choice from user for
    i) HEX to BCD   ii) BCD to HEX   iii) EXIT.
   Display proper strings to prompt the user while accepting the input and displaying the result.

2. Write ALP to perform string manipulation to calculate string length and reverse a string. The strings to be accepted from the user is to be stored in code segment Module_1 and write FAR PROCEDURES in code segment Module_2 for following operations on the string:
    i)   Concatenation of two strings
    ii) Compare two strings
    iii) Number of occurrences of a sub-string in the given string
    iv) Find number of words, characters, number of lines and number of capital letters from the given text in the data segment

   Note: Use PUBLIC and EXTERN directive. Create .OBJ files of both the modules and link them to create an EXE file.

3. Write following programs in C using int86, int86x, intdos, intdosx functions
    i.   To delete a file
    ii. To create a directory
    iii. Read and display disk information such as Drive, tracks, sectors etc

4. Study of 80386 architecture (functional diagram, register set and addressing modes)
5. Write ALP to switch from real mode to protected mode and back to real mode.

6. Classify the protected mode exceptions as per the vector.

B. Microcontroller Programming

Assignments based on programming 8051 microcontroller using 8051 development board.

1. Write a program to add n, 8 bits numbers found in internal ram location 40H onwards and store results in R6 and R7.

2. Write a program to multiply 16 bit number by 8 bit number and store the result in internal memory location
                    OR
2. Write a program for the block transfer (external to internal memory)

3. Timer programming: ISR based
   Write ALP to generate 2 KHz square wave using Timer interrupt on any port pin.

4. Write ALP to interface 8051 with:
   Select any one of the given assignment.
       i.   Interfacing DAC and writing programs to generate triangular, trapezoidal and sine waveforms.
       ii.   Interfacing 8/12 bit ADC to 8051 or equivalent and to write a program to find out the average value for 10 readings.
       iii. Interface stepper motor to 8051 and write a program to rotate motor with different step angles and with different speeds.

- Student should submit term work in the form of a journal based on the above assignments.
- Oral examination will be based on the term work
- Candidate is expected to know the theory involved in the experiment.

Text Books
3. James Turley, "Advanced 80386 Programming Techniques ", McGraw Hill Education, ISBN: 9780070598416
4. Douglas V Hall.," Microprocessors and Interfacing", Tata McGraw Hill
5. Mazidi M. Gillipse J. "The 8051 Microcontroller and Embedded Systems", Pearson education,2002,ISBN-81-7808-574-7

Reference Books
1. Tribel Singh,"8088 /8086 Processor", PHI
2. Kenneth Ayala ,"The 8051 Micro Controller", Delmar Cengage Learning; 3rd edition (June 1, 2004) 3rd Edition
3. Intel 8 bit Microcontroller manual.

214454: DATA STRUCTURES AND FILES LIBORATORY

Teaching Scheme   **Examination scheme**
Practical: *4 hrs / week*   Term Work: *50 Marks*
                                                           Practical: *50 Marks*

List of assignments

1. Implement stack as an abstract data type using linked list and use this ADT for conversion of infix expression to postfix , prefix and evaluation of postfix/prefix expression.
2. Implement circular queue using array and perform following operations on it.
   i) Add a record ii) Delete a record iii) Checking Empty iv) Checking Underflow v) Checking overflow
3. Implement priority queue as ADT using multiple linked lists ,one list for each priority for servicing patients in an hospital with priorities as i) Serious (top priority) ii) medium illness (medium priority) iii) General (Least priority).
4. Construct and expression tree from postfix/prefix expression and perform recursive and non-recursive Inorder , preorder and postorder traversals.
5. Implement binary search tree as an ADT
6. Construct an inorder threaded binary tree from inorder/postorder expression and traverse it in inorder and preorder
7. Represent any real world graph using adjacency list/adjacency matrix find minimum spanning tree using Prim's or Kruskal's algorithm.
8. Represent a given graph using adjacency matrix/adjacency list and find the shortest path using Dijkstra's algorithm.
9. Implementation of Hash table using array and handle collisions using Linear probing, chaining without replacement and Chaining with replacement
10. Implement Heap sort by constructing max or min heap .
11. Implement an index sequential file for any Database and perform following operations on it
i) Create Database ii) Display Database iii) Add a record iv) Delete a record v)Modify a record

Note:
1. For all programs implementations students are expected to use meaningful identifiers, proper indentation, use of functions, minimal use of global variables and writing time complexity using any one notation is mandatory.
2. Student should submit term work in the form of a journal based on the above assignments. Practical examination will be based on the term work. Questions will be asked during the examination to judge the understanding of the practical performed in the examination. Candidate is expected to know the theory involved in the experiment.
3. All the assignments are to be implemented in C++ using Unix/Linux operating system .
4. Teaching object oriented technology using C++ is expected in 2 hour theory / week.

214455: COMPUTER GRAPHICS LABROTARY

Teaching Scheme   **Examination scheme**

Practical Hours: 2 *hrs / week*   Term Work: *25 Marks*
                                                        Practical: *50 Marks*

List of Assignments

1) Line Drawing using DDA and Bresenham with patterns such as simple, dash, dash dot, thick

   Students are expected to divide the screen in four quadrants and should draw line in all quadrants of all slopes with key board/Mouse interface.

2) Circle Drawing using DDA, Midpoint & Bresenham Algorihtms with key board/Mouse interface.

3) **Polygon filling** :

   i) Non Recursive Seed Fill / Flood Fill with implementation of inside test

   ii) Scan Line algorithm with Pattern Filling

4) **2 D Transformations:**

   i) Simple Translation   ii) Scaling, rotation about origin   iii) Scaling, Rotation about arbitrary point

5) **2D Transformations** : Reflection about X and Y axis and about arbitrary axis

6) **2 D Clipping** : i) Line Clipping – Cohen Sutherland Outcode Method   ii) Sutherland Hodgman Polygon Clipping

7) **3 D Projections**: Display of 3D object such as Cube using general parallel Projection

8) **Curve Generation**: Design any object such as flower, waves etc using following curve generation techniques i) Curve generation using Interpolation ii) Bezier Curve

9) Design an animation Sequence using Segmentation

10) Design an animation Sequence using any animation tool.