UNIVERSITY OF PUNE
SE (COMPUTER ENGINEERING) 2008 COURSE

Term – I

Subject Code No.	Subject	Teaching Scheme Hours / Week		Examination Scheme				Total Marks
		Lecture	Practical	Paper	TW	Practical	Oral	Total
210241	Discrete Structures	04	----	100	---	---	---	100
210242	**Programming & problem solving**	**04**	**----**	**100**	**---**	**---**	**---**	**100**
210243	Digital Electronics and Logic Design	04	----	100	---	---	---	100
210244	Data Structures and Algorithms	04	----	100	---	---	---	100
207005	**Humanities and Social science**	**03**	**----**	**100**	**---**	**----**	**----**	**100**
210246	Programming Laboratory		04	---	25	50	---	075
210247	Digital Electronics Laboratory	---	04	---	25	50	---	075
210248	Soft Skills	---	02	---	50	---	---	050
	Total	**19**	**10**	**500**	**100**	**100**	**---**	**700**
	Total of Part I (A)	**29 Hrs**		**700**				

Term – II

Subject Code No.	Subject	Teaching Scheme Hours / Week		Examination Scheme				Total Marks
		Lecture	Practical	Paper	TW	Practical	Oral	Total
207003	Eng. Maths – III	04	---	100	---	---	---	100
210249	Microprocessors and Interfacing Techniques	04	---	100				100
210250	Data Structures	04	---	100	---	---	---	100
210251	Computer Graphics	03	---	100	---	---	---	100
210252	Computer Organization	03	---	100	---	---	---	100
210253	Object Oriented Programming & Computer Graphics Laboratory	02	02	---	50	50	---	100
210254	Microprocessors and interfacing Laboratory	---	04	---	50	50	---	100
210255	Data Structures Laboratory	---	04	---	50	50	---	100
	Total	**20**	**10**	**500**	**150**	**150**	**---**	**800**
	Total of Part II (B)	**30 Hrs**		**800**				
	Grand Total (A) + (B)			**1500**				

210241: DISCRETE STRUCTURES

Teaching Scheme **Examination Scheme**
Lectures: 4 Hrs/week **Theory: 100 Marks**

Discrete mathematics- the mathematics of integers and of collections of object – underlies the operation of digital computer, and is used widely in all fields of computer science for reasoning about data structures algorithms and complexity. The primary objective of subject is to prepare students mathematically for the study of computer engineering. Topics covered in the course include proof techniques, logic and sets, functions, relations, counting techniques, probability and recurrences. By the end of the course, students should be able to formulate problems precisely, solve the problems, apply formal proof techniques, and explain their reasoning clearly.

Prerequisite: Basic Mathematics

Learning objectives: *… the student will be able to*

- *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.*

Unit I
Sets and Propositions
Sets, Combination of sets, 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 **(8 Hrs)**

Unit II
Groups and Rings
Algebraic Systems, Groups, Semi Groups, Monoid, 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 **(8 Hrs)**

Unit III
Relations and Functions
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 **(10 Hrs)**

Unit IV
Graphs
Basic terminology, multi graphs and weighted graphs, paths and circuits, shortest path in weighted graph, Hamiltonian and Euler paths and circuits, factors of a graph, planer graph and Travelling salesman problem. **(8 Hrs)**

Unit V

Trees

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

Unit VI

Permutations, Combinations and Discrete Probability

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

(8 Hrs)

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. Kenneth H. Rosen, "Discrete Mathematics and its Applications", 6th edition, McGraw-Hill, 2007. ISBN 978-0-07-288008-3
3. E. Goodaire and M. Parmenter, "Discrete Mathematics with Graph Theory", 2nd edition, Pearson Education, 2003 ISBN 81 – 7808 – 827 – 4
4. Semyour Lipschutz & Marc Lipson, " Discrete Mathematics", McGraw-Hill, 3rd Special Indian Edition, ISBN-13 : 978-0-07-060174-1
5. B. Kolman, R. Busby and S. Ross, "Discrete Mathematical Structures", 4th Edition, Pearson Education, 2002, ISBN 81-7808-556-9
6. N. Deo, "Graph Theory with application to Engineering and Computer Science", Prentice Hall of India, 1990, 0 – 87692 – 145 – 4

207005: HUMANITIES AND SOCIAL SCIENCES

Teaching Scheme	Examination Scheme
Lectures: 3 Hrs/week	Theory: 100 Marks

Learning Objectives

This course will lead to the learning of

1. Human and social development;
2. Contemporary national and international affairs;
3. Emergence of Indian society and Economics,
4. Sectoral development and Economic development and related issues (such as international economics, WTO, RBI, etc).

Outcome

Making engineering and technology students aware of the various issues concerning man and society. These issues will help to sensitize students to be broader towards the social, cultural, economic and human issues, involved in social changes.

Methodologies

1. Suitable case studies should be discussed
2. Student group discussion activity.
3. Social Networking activity.

Unit I : Indian Society (6 hrs)

Structure of Indian Society, Indian Social Demography– Social and Cultural, Differentiations: caste, class, gender and tribe; Institutions of marriage, family and kinship- Secularization –Social Movements and Regionalism- Panchayatraj Institutions; Affirmative Action Programme of the Government-various reservations and commissions.

Unit II: Social Development (6 hrs)

Scientific approach to the study of human beings. Evolution of human kind, social change and evolution. Industrial revolution. National policy on education, health and health care and human development.

Unit III : Sectoral Development (6 hrs)

Agriculture : Technology changes , Green revolutions , Employment Rural & Urban , Government Schemes .

Industrial Development : Strategies , Public & Private Sectors, Categories , infrastructure , transport & communication , Consumer Awareness.

Unit IV: Environment & Ecology (6 hrs)

Ecosystems : Structure, Working, components.

Pollution : Water & Air Pollution , Global Warming , Control Strategies , International Treaties.

Energy Sources : Renewable & Non Renewable , Hydro power, Biomass , Ocean , Geothermal & Tidal .

Global Environmental Issues : Population Growth, Soil Degradation, Loss of Biodiversity.

Unit V : Economic Development (6 hrs)

Need for planned economic development – Law of demand and supply. Planning objective, five
year plans, priorities and problems. Population and development.
Indian Economics – basic features, natural recourses population size and composition, national
income concepts, micro economics of India , inflation.

Unit VI : Banking & Trades (6 hrs)

Financial Analysis, Ratios, Cost Analysis,, financial Institutions, Finance Commissions, Budget Analysis.
Indian Banking, Role of Reserve bank of India
International Economy, WTO, International aid for economic growth.

Reference Books:

1. Krugman, International Economics, Pearson Education.
2. Prakash, The Indian Economy, Pearson Education.
3. Thursen Gerald, Engineering Economics, Prentice Hall.
4. C.S. Rao, Environmental Pollution Control Engineering, New Age International Pvt. Ltd.
5. Rangarajan, Environmental Issues in India, Pearson Education.
6. University of Delhi, The Individual & Society, Pearson Education.
7. Wikipedia.org / wiki /social studies.
8. M. N. Srinivas, Social change in modern India, 1991, Orient Longman.
9. David Mandelbaum, Society in India, 1990, Popular.

210242: PROGRAMMING & PROBLEM SOLVING

Teaching scheme: **Examination Scheme:**
Lectures: 4 Hrs/Week **Theory: 100 Marks**

UNIT I **(8 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 **(8 Hrs)**

Introduction to programming structure, pointer for structuring the program, modules and their function cohesion & Coupling, Local and global variable, parameters, return values, variable names and data dictionaries, Problem solving with sequential logic structure, Solution development, Problem solving with decision- Logic structure, multiple if-then-else using straight through logic using positive & negative logic, logic conversion, decision tables

UNIT III **(8 Hrs)**

Fundamental algorithm - Exchanging Values of two variables, Counting, summation of set of numbers , factorial computation, sine function computation, Fibonacci series, reverse of digit, BCD conversion , Char to number conversion), Factoring methods - Square root of number, smallest divisor, GCD of two number, prime number, prime factors of integer, pseudo random number generation, raising the number to a large power

UNIT IV **(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

UNIT V **(6 Hrs)**

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 VI **(10 Hrs)**

Concept of object oriented programming – objects, classes, Methods, Abstraction, Inheritance, Encapsulation, Understanding public, private, protected access, Constructor, Destructor, Implementation of above concepts using of C++

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

Reference book

1. “Programming Logic and Design” by Joyce Farrell ISBN 978-81-315-0263-1
2. Balaguruswamy, “Object-oriented Programming with C++”

210243: Digital Electronics & Logic Design

Teaching Scheme:	**Examination Scheme:**
4 Hrs/ week	**Theory: 100 Marks**

Prerequisites: Basic Electronics Engineering

Learning 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 basic components of microprocessors

UNIT I **8 Hrs.(TB-1: Ch- 2, RB-1)**

Number System & Logic Design Minimization Techniques:

Introduction: Binary, Hexadecimal numbers, octal numbers and number conversion.
Signed Binary number representation: Signed Magnitude, 1's complement and 2's complement representation.
Binary, Octal, Hexadecimal Arithmetic: 2's complement arithmetic.
Algebra for logic circuits: Logic variables, Logic functions -NOT, AND, NOR, XOR, OR, XNOR, NAND
Boolean algebra: Truth tables and Boolean algebra. Idealized logic gates and symbols. DeMorgan's rules Axiomatic definition of Boolean algebra, Basic theorems and properties of Boolean algebra
Logic minimization: Representation of truth-table, SOP form, POS form, Simplification of logical functions, Minimization of SOP and POS forms, Don't care conditions
Reduction techniques: K-Maps up to 4 variables and Quine-McClusky technique

UNIT II **6 Hrs. (TB-1:** Ch-3, 4, **RB- 2)**

Logic Families:

TTL: Standard TTL characteristics- Speed, power dissipation, fan-in, fan-out, current and voltage parameters, noise margin, operating temperature etc. Operation of TTL NAND gate. TTL Configurations- Active pull-up, Wired AND, totem pole, open collector.
CMOS: CMOS Inverter, CMOS characteristics, CMOS configurations- Wired Logic, Open drain outputs
Interfacing: TTL to CMOS and CMOS to TTL

UNIT III **9 Hrs.(TB-1: Ch-5,6, RB- 1, 3)**

Combinational Logic:

Codes:- BCD, Excess-3, Gray code , Binary Code and their conversion
Arithmetic Operations: - Binary Addition, Subtraction, Multiplication, Division, BCD Addition
Circuits: - Half- Adder, Full Adder, Half Subtract or, Full Sub tractor, BCD adder using and subtract using 7483, look ahead and carry, parity generator and checker using 74180, magnitude comparator using 7485.
Multiplexers (MUX):- Working of MUX, Implementation of expression using MUX (IC 74153, 74151).
Demultiplexers (DEMUX):- Implementation of expression using DEMUX, Decoder. (IC 74138).

UNIT IV **8 Hrs.(TB-1:** Ch-12, **TB2:** Ch-13, **RB-1)**

Sequential Logic:

Introduction: Sequential Circuits. Difference between combinational circuits and sequential circuits

Flip- flop: SR, JK, D, T; Preset & Clear, Master and Slave Flip Flops their truth tables and excitation tables, Conversion from one type to another type of Flip Flop.

Application of Flip-flops: Bounce Elimination Switch, registers, counters.

Registers: Buffer register; shift register;

Counters: Asynchronous counter. Synchronous counter, ring counters, BCD Counter, Johnson Counter, Modulus of the counter (IC 7490), Pseudo Random Binary Sequence Generator, Sequence generator and detector

UNIT V **9 Hrs.(TB-2:** Ch-6,7, Appendix-A, **RB-4)**

ASM & VHDL :

Algorithmic State Machines: ASM charts, notations, design of simple controller, multiplexer controller method

Examples: Sequence Generator, Types of Counter

VHDL: Introduction to HDL, VHDL- Library, Entity, Architecture, Modeling Styles, Concurrent and Sequential Statements, Data Objects & Data Types, Attributes.

Design Examples: VHDL for Combinational Circuits-Adder, MUX. VHDL for Sequential Circuits-Synchronous and Asynchronous Counter

UNIT VI **8 Hrs.(TB-1:** Ch-13 (13.1 to 13.9))

PLDs & Introduction to Microprocessor :

PLD: PLA- Input, Output Buffers, AND, OR, Invert/ Non-Invert Matrix.

Design Example: Any 4 Variables SOP function using PLDs, Study of basic architecture of FPGA.

Introduction to Microprocessor: Introduction of Ideal Microprocessor, Data Bus, Address Bus, Control Bus, Microprocessor Based System- Basic Operation, Microprocessor Operation, Microprocessor Architecture, Instruction Set

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 Mc-Graw Hill

4. J.Bhaskar "VHDL Primer" 3rd Edition, Pearson Edition

210244: DATA STRUCTURES AND ALGORITHMS

Teaching Scheme	Examination Scheme
Lectures: 4 Hrs/week	Theory: 100 Marks

Prerequisite: Fundamentals of Programming Languages (Subject Code: 110013).

- To understand the different ways of data representation.
- To define high level of abstraction of the needed linear data structure and algorithm.
- To develop the ability to synthesize and analyze algorithms
- To study the representation, implementation and applications of linear data structures

UNIT I: Review of 'C' [7 Hrs]

Arrays, Pointers: arrays & pointers.

Functions: Parameter passing call by value and call by reference, scope rules, functions and pointers, function returning pointer and pointer to function, String manipulations using arrays, pointer to pointer.

Structure and Union: Passing and returning structure as parameter for function ,structure and pointer.

Recursion: Definition, writing recursive functions & how recursion works.

File handling using C.

UNIT II: [TB 2] [7 Hrs]

Introduction to Algorithm, Data structures & Analysis of algorithms :

Introduction to Data Structures: Concept of data, Data object, Data structure, Abstract Data Types (ADT), 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 III:[TB 1] [8 Hrs]

Linear Data Structures using Sequential Organization:

Concept of sequential organization, Concept of Linear data structures, arrays as ADT, Storage representation of array – Row major and Column major & their address calculation, Multidimensional arrays, Concept of ordered list.

Applications : Polynomial representation using array, Concept of Sparse Matrix, it's usage & representation using arrays, Algorithms for sparse matrix operations like addition, simple transpose, fast transpose & multiplication.

Analysis of the algorithms used.

Unit IV:[TB 1 & TB 2] [8 Hrs]

Sorting and searching techniques:

Need of sorting and searching, sorting order & stability in sorting.

Sorting Techniques: Algorithms for Bubble sort, Selection sort, Insertion sort, Shell sort, Radix sort, Quick sort and Merge sort.

Analysis of each sorting technique for best, worst and average case, Concept of Internal & External sorting.

Searching Techniques: Algorithms for Sequential search, Binary search, Fibonacci search & concept of Index Sequential search, analysis of each searching technique for best, worst and average case.

UNIT V:[TB 1] **[9 Hrs]**

Linear Data Structures using Linked Organization:

Limitations of static memory allocation. Dynamic memory allocation in C.

Concept of linked organization, Singly linked list, Doubly linked list, Circular linked list. Operations like insertion, deletion, traversal & other operations on these data structures.

Applications: Representation & manipulation of polynomials using circular linked lists, Application of doubly linked list in dynamic storage management, garbage collection and compaction. Representation of polynomial using generalized linked list (implementation not expected), Concept of skip list.

Analysis of the algorithms used.

UNIT VI:[TB 1 & TB 2] **[9 Hrs]**

Stacks and Queues:

Stacks: Concept of stack as ADT, Representation and implementation of stack using sequential & linked organization.

Applications: Examples using implicit stack, Simulating recursion using explicit stack, Arithmetic expression conversion & evaluation, reversing a string, parsing : well- formed parenthesis checking, concept of multi-stack & it's representation. **Analysis** of the algorithms used.

Queues: Concept of queue as ADT, Representation and implementation of linear queue & circular queue using sequential & linked organization.

Applications: Josephus problem, Job scheduling, Queue simulation, Categorizing data, Double ended queue, Multi-queue and Priority queue. Analysis of the algorithms used.(Implementation not expected)

Text Books (TB):

1. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C", Cenage Learning, ISBN 9788131503140.
2. E. Horowitz , S.Sahani, S.Anderson-Freed ""Fundamentals of Data Structures in C", Universities Press ,2008 ,ISBN 10:8173716056

Reference Books(RB):

1. A. Aho, J. Hopcroft, J. Ulman, "Data Structures and Algorithms", Pearson Education, 1998, ISBN-0-201-43578-0

2. Y. Langsam, M. Augenstin and A. Tannenbaum, "Data Structures using C and C++", 2nd Edition, Prentice Hall of India, 2002, ISBN-81-203-1177-9

3. J. Tremblay, P. Soresan, "An introduction to data structures with Applications", 2nd edition, Tata McGraw-Hill International Editions, 1984, ISBN-0-07-462471-7.

210246: PROGRAMMING LABORATORY

Teaching Scheme	Examination Scheme
Practical : 4 Hours / Week	**Term Work** : 25 Marks
	Practical : 50 Marks

Suggested List of Assignments Based on Data Structures and Algorithm:

1. Write a program to perform Set operations - Union, Intersection, Difference, Symmetric Difference etc.
2. Write a program to perform various string operations such as Copy, Length, Reversing, Palindrome, Concatenation and to find occurrence substring etc with and without using library functions.
3. Write a program to perform operations on matrices like addition, multiplication, saddle point, magic square ,inverse & transpose etc using functions & pointers.[Minimum 4 operations]
4. Write a program to perform following operations on any database: Add, Delete, Modify, Display, Search & Sort etc.
5. Implement Sorting Methods using functions- Bubble Sort, Selection Sort, Insertion Sort, and Shell Sort.
6. Implement Sorting Methods using recursion- Quick Sort and Merge Sort.
7. Implement Searching Methods-Sequential Search, Binary Search, Fibonacci Search and Index Sequential Search.[Minimum 3 searching methods]
8. Represent polynomial using structures and write a menu driven program to perform Addition, Multiplication and Evaluation.
9. Represent Sparse Matrix using array and perform Matrix Addition, Simple and Fast Transpose.
10. Write a menu driven program to perform following operations on SLL/CDLL : Create, Insert – Start, end, between, Search & delete, Reverse ,Display etc.
11. Create two Singly Linked lists, sort one after creation and one while creation using Pointer manipulation. Merge these two lists into one list without creating a new node or swapping of the data.
12. Represent a polynomial using Circular Linked List and write a menu driven program to perform Addition, Multiplication and Evaluation.
13. Implement Stack as an ADT using Array. Use this ADT to perform expression conversion and evaluation (infix to postfix, infix to prefix, prefix to infix, prefix to postfix, postfix to infix and postfix to prefix).
14. Represent Circular Queue using Linked List and write a program to perform operations like Insert, Delete, Finding front and rear element.
15. Implement the Mini Project of Student Database using Linked list for following requirements:

a. Creation of Student Database in memory containing student ID, Name, Name Initials, Address, Contact No and Date of Birth .
b. Insertion, Deletion, Modification of student record for a given student ID.
c. Sorting on name initials and searching a particular student record on name initials

Note: All Assignments to be implemented using C and Time and Space Complexity is to be verified with theoretical findings.

- Students will submit Term Work in the form Journal that will include minimum above 15 assignments. Each assignment will consist of Pseudo algorithm, program listing with proper documentation and printout of the output.
- Practical examination will based on the Term Work and questions will be asked to judge the understanding of the assignments performed at the time of the examination.

210247: Digital Electronics Lab

Teaching Scheme
Practical: 4 Hrs/week

Examination Scheme
Practical: 50 Marks
Term Work: 25 Marks

Suggested List of Assignments:

A. Combinational Logic Design

1. T.T.L Characteristics (Study and Write up only).
2. Code converters e.g. Excess-3 to BCD and vice versa
3. Multiplexers: Application like Realization of Boolean expression using Multiplexer.
4. Demultiplexers: Applications like Realization of ROM using Demultiplexer.
5. BCD adder/Subtractor using 4 bit binary adder 7483.
6. Parity generator / detector.

B. Sequential Circuit Design

1. Flip flops, Registers and Counters (Study and Write up only).
2. 4-bit Multiplier / Divider (Study and Write up only).
3. Ripple counter using flip-flops.
4. Sequence generator using JK flip-flop.
5. Sequence detector using JK flip-flop.
6. Up-down counter using JK flip-flop.
7. Modulo N counter using 7490 & 74190 (N>10).
8. Pseudo random number generator.
9. Design of a barrel shifter

C. Study /Implement of VHDL and examples of Combinational and sequential circuits

Combinational Circuits: Adder, MUX
Sequential Circuits: Asynchronous or Synchronous Counter

D. ASM, PALS and FPGA

1. Simple ASM using multiplexer controller method.
2. Implementation of combinational logic using PLAs
3. Study of FPGA devices (Study and Write up only).

- 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 minimum of 16 assignments of which assignment of Group C and 2 assignments from Group D are compulsory.

- Practical examination will be based on the term work and questions will be asked to judge the understanding of assignments performed at the time of examination

210248: SOFT SKILLS

Teaching Scheme **Examination Scheme**
Practical: 2 Hrs/week **Term Work: 50 marks**

Learning Objectives

1. To encourage the all round development of students by focusing on soft skills.
2. To make the engineering students aware of the importance, the role and the content of soft skills through instruction, knowledge acquisition, demonstration and practice.
3. To develop and nurture the soft skills of the students through individual and group activities.
4. To expose students to right attitudinal and behavioral aspects, and to build the same through activities.

The coverage of soft skills that help develop a student as a team member, leader, all round professional in the long run have been identified and listed here for reference. As the time allotment for the soft kills laboratory is small and the fact that these skills are nurtured over years, students are encouraged to follow up on these skills as self-study and self driven process.

UNIT I: Self-Development and Assessment
Self-Assessment, Self-Awareness, Perceptions and Attitudes, Positive Attitude, Values and Belief Systems, Personal Goal setting, Career Planning, Self-Esteem, Building of Self Confidence, Personal success factors, Handling failure, Depression and Habit, SWOT analysis, prioritization, Emotional Intelligence (EI) and Emotional Quotients (EQ), Self appraisal

UNIT II: Interpersonal Relations
Nature of groups and teams, Team effectiveness, Group discussions and decision making, Emotional Intelligence (EI) and Emotional Quotients (EQ), and its effect on team, Cross Cultural Aspects, Inter dependence, Peer Reviews

UNIT III:. Ethics and Social Responsibilities
Personal professional and corporate ethics, Ethical dilemma, Corporate social responsibilities: Green computing, Social accounting, Auditing, Civic sense.

UNIT IV: Corporate
Corporate grooming and dressing, Etiquettes in social as well as office settings, Email Etiquettes, Telephone Etiquettes, Contemporary issues in corporate life: diversity, Attrition, Work life balance, Hygiene and health.

UNIT V: Leadership Skills
Leaders: their skills, roles, and responsibilities. Vision, Empowering and delegation, motivating others, organizational skills, team building, Organizing and conducting meetings, decision making, giving support, Vision, Mission, Coaching, Mentoring and counseling, Appraisals and feedback, conflict, Power and Politics, Public Speaking.

UNIT VI: Other Skills
Managing Time, Managing Stress, Meditation. Improving personal memory, Study skills that include Rapid Reading, Notes Taking, Self learning, Complex problem solving and creativity.

References (Note : No textbooks have been assigned for the subject as this is a lab based course)

Topic 1: Any good books like

1. Stephen Covey, "7 Habits of highly effective people"
2. Daniel Goleman " Working with Emotional Intelligence"

Topic 2 -6

(Note: Organizational behavior books give a formal theoretical, in depth approach to topics here. We don't expect that rigor here as this course is expected to just give a general idea to students about the concepts involved (through some trainer sessions, group discussions..) And focus more on usable skills practice that they can use in their personal and professional life honed through lab assignments)

1. Organizational Behavior (Special Indian Edition), 4/e
   Steven McShane, Mary Ann Von Glinow, Radha R Sharma,
2. Organizational Behaviour**:** Dipak Kumar Bhattacharyya oxford press
3. Essentials of Organizational Behavior Global Edition
   10th Edition Stephen Robbins, Timothy Judge
4. Tim Hindle, "Reducing Stress", Essential Manager Series DK publishing.
5. Dr R L Bhatia, "Managing Time for a competitive edge".
6. Lorayne, Lucas "Memory Book".
7. Robert Heller, "Effective leadership", Essential Managers DK publishers.

LIST OF SUGGESTED ASSIGNMENTS

1. Write a Personal essays and resume or statement of purpose which MAY include some of the following:
   a. Who am I (family background, past achievements, past activities of significance).
   b. Strengths and weaknesses (how to tackle them) (SWOT analysis).
   c. Personal short-term goals, long-term goals and schedule and prioritization to achieve them.
   d. Self-assessment on your soft skills capabilities.
   e. Self appraisal on your last year at college
   f.. Self assessment on your EQ
2. Students could review and present to a group from following ideas:
   a. Book review.
   b. Biographical sketch.
   c. Any topic such as an inspirational story/personal values/beliefs/current topic, speeches.
   d. Ethics and etiquettes and social responsibilities as a professional.
   e. Success Factors.
3. Students will present to a group from following ideas:
   a. Public speaking exercise in form of debate or elocution on current affairs/socials issues/ethics and etiquettes
   b. Public speaking exercise in form of debate or elocution on Contemporary issues in corporate life: diversity, Attrition, Work life-Balance, Hygiene and health.
   c. Preparing Vision/Mission/Goals statements for
      - College
      - Hypothetical Organization
4. Students will participate in FEW activities from following:
   a. Team games for team building.
   b. Situational games for role playing as leaders, members.

c. Organizing mock events.
d. Conducting meetings and documenting.
e. Group discussion current affairs/socials issues/ethics and etiquettes
f. Group discussions on Contemporary issues in corporate life: diversity, Attrition, Work life balance, Hygiene and health.

5. Faculty may arrange one or more sessions from following:
Yoga and meditation.
Stress management, relaxation exercises, and fitness exercises.
Time management and personal planning sessions.
Improving memory skills.
Improving leadership skills.
Improving English conversation skills.
Reading comprehension skills and Notes taking skills.

Students are expected to keep a personal record of ANY SIX activities that they undertook in the Soft skills Laboratory in the form of a journal. All students need not do the same assignments. Colleges have a freedom within the framework to customize set of activities to be followed, sessions to be conducted and references to follow.

Guidelines for Conduction and Assessment of Laboratory Work

1. This laboratory can be seen as a departmental activity with one of the faculty as coordinator.
2. Professionals from HRD departments of companies could assist in training sessions based on individual college contacts.
3. Certain activities can even be team activities such as Group Discussion..
4. Popular science, INTERNET, Magazines, Newspapers, and Training MEDIA from BCL, BBC, Management Institutes, and Management Gurus can also be used as resources.
5. Generally an exercise can be designed to allow multiple skills exposure for example a group task encouraging discussions, team building, value sharing, leadership and role play all at the same time.

ASSESMENT Guidelines Evaluation can be based on

1. Overall participation in soft skills based lab activities
Attendance and enthusiasm, Participation and contribution in event management, organizing, Group games, group exercises, and interpersonal skills observed.
2. Quality of journal for soft skills laboratory indicating personal progress, participation.

Guidelines for batch wise Time management for laboratory sessions (Two hour session at a time)

A Semester allows for 12-14 sessions. Students can do Lot of preparation at leisure time.

1. Batches could be of size 25 to 30 students.
2. Group discussions could be done for groups of 5-8 students at a time (2 sessions)
3. Sessions could be organized for trainers to give directions, knowledge, experience sharing. AND/OR Sessions of common viewing of training material on Video etc. **(4 sessions)**
4. Group exercises for team building, role-playing and interaction with professional. **(3 sessions)**
5. Some individual presentations / write-ups **(3 sessions)**

210249: Microprocessor and Interfacing Techniques

Teaching Scheme **Examination Scheme**
Lectures: 4 Hrs/week **Theory: 100 Marks**

Prerequisites: Digital Electronics and Logic Design (Subject Code: 210243)

Learning Objectives:

1. To learn the architecture and assembly language programming 0f 8086 Microprocessor.
2. To learn peripherals and their interfacing with 8086 Microprocessor.
3. To study the DOS Internals.
4. To Study NDP and Design of Microprocessor based System.

Unit I

Introduction to 16 bit microprocessor, Architecture and Pin diagram of 8086, Programmers model of **8086** (Registers), Segmentation, logical to physical address translation, even and odd memory banks, Read write cycle timing diagrams, Address mapping and decoding, I/O: memory mapped I/O & I/O Mapped I/O. **(8 Hrs)**

Unit II

Addressing modes, Instruction set of 8086 in detail, Instruction Formats, Stacks, Assembly Language Programming, Assembler, Linker, Debugger (Turbo debugger), Directives, Procedures (Near & Far), Macros, Loop constructs, 8086 Programming examples. **(8 Hrs)**

Unit III

8086 Interrupt Structure, Interrupt Vector Table (IVT), ISR, Hardware and software Interrupts Internals of DOS, DOS loading, DOS memory map, Internal and external commands of DOS, BIOS & DOS Interrupts. Concepts of PSP, .EXE & .COM files, Concepts of TSR, **8259** (Programmable Interrupt Controller): Features, Block Diagram, Control & status registers, Interfacing & Programming **(8 Hrs)**

Unit IV

Study of Peripheral chips:

8255 (Programmable Peripheral Interface), Serial Communication- Synchronous & Asynchronous, **8251**(USART): Features, Block Diagram, Control & status registers, Operating modes, Interfacing & Programming (8255 and 8251)

Concept of ADC -Successive Approximation & Interfacing, Concept of DAC R-2R (ladder) & Interfacing, Introduction to Sensors & Transducers, Keyboard Display & Centronics Printer Parallel Interfacing using 8255. **(9 Hrs)**

Unit V

8279 Keyboard and Display Controller, **8253** (Programmable Interval Timer): Features, Block Diagram, Control & status registers, Operating modes, Interfacing & Programming

Concept of DMA, **8237** DMA Controller: Features, Block Diagram **(7 Hrs)**

Unit VI

Minimum & Maximum mode of 8086, Support chips 8282,8284,8286,8288

8087(NDP) - Features, Block Diagram, Control & status registers, typical Instruction Set & Programming

Detail Design of 8086 based minimum system with EPROM, SRAM & Peripherals such as 8255,8253,8251,8279 with keyboard & seven segments Display. **(8 Hrs)**

Text Books:

1. Douglas Hall, "Microprocessors & Interfacing", McGraw Hill, Revised 2nd Edition, 2006 ISBN 0-07-100462-9
2. John Uffenbeck," The 8086/88 Family: Design, Programming & Interfacing", PHI,
3. A.Ray, K.Bhurchandi, "Advanced Microprocessors and peripherals: Arch, Programming & Interfacing",Tata McGraw Hill,2004 ISBN 0-07-463841-6

References Books:

1. Liu, Gibson, "Microcomputer Systems: The 8086/88 Family", 2nd Edition, PHI,2005
2. Kenneth Ayala, "The 8086 Microprocessor: Programming & Interfacing the PC", Cengage Learning, Indian Edition, 2008
   Ray Dunkon, "Advanced MSDOS Programming", 2nd Edition, BPB Publication.
3. Kip Irvine, "Assembly language for IBM PC", PHI, 2nd Edition, 1993
4. Peter Abel, "Assembly language programming", Pearson Edu,5th Edition,2002
5. Intel Microprocessor and peripheral Handbook: Volume 1
6. Yashwant Kanitkar, "TSR through C", BPB Publication, 1995, ISBN 81-7029-520-3.

210250: DATA STRUCTURES

Teaching Scheme
Lectures: 4 Hrs/week

Examination Scheme
Theory: 100 Marks

Prerequisite: Data Structures and Algorithms (Subject Code: 210244)
Learning Objectives

1. To study the representation, implementation and applications of data structures
2. To study implementation of data structures using OOP concepts
3. To compare the benefits of static and dynamic data structures
4. To choose the appropriate data structure for modelling a given problem

UNIT – I [TB 1]
Trees
Basic tree concepts, binary trees and their properties, representation using sequential and linked organization, full and complete binary trees, converting tree to a binary tree, binary tree traversals, BFS ,DFS (recursive and non recursive), infix, postfix, prefix, Huffman's codes. Binary search trees & operations.BST as an ADT, Threaded binary trees, Insertion and deletion of nodes in in-order threaded binary tree, pre-order, in-order and post order traversals of in-order threaded binary tree, applications of binary trees: Gaming, Expression and decision trees **(8 Hrs)**

UNIT – II [TB 1]
Graphs
Basic concepts, operations, graphs storage structures, Traversals: Depth First and Breadth First. Graph algorithm, Graph as an ADT, Minimum spanning trees: Kruskal's and Prim's. Algorithm for shortest path and topological sorting **(8 Hrs)**

UNIT – III [TB 1]
Symbol Tables: Static & dynamic tree table ,AVL tree ,AVL tree implementation, AVL tree algorithms.
Hash Tables: Basic concepts, hash function, hashing methods, collision resolution, bucket hashing. **(8 Hrs)**

UNIT IV [TB 1]
Heaps and multi way trees
Heap: Basic concepts, heap implementation algorithm & heap sort, heap as an ADT, heap applications.
Multi way trees: B tree implementation, B-tree variations **(8 Hrs)**

UNIT V [RB4]:

Files

External storage devices, Files: Definition and concepts, File organization: Sequential files, random, linked, inverted and cellular partitions. Processing of sequential, Index-sequential and direct files.
Sequential file organisation, direct file organisation, index sequential file organisation and their implementation. **(8 Hrs)**

UNIT VI [TB 2]
Abstract data types: ADT, classes and objects, generic programming: introduction to STL (Standard Template Library), containers, iterators and algorithms, study of container template classes for vectors and stacks and related algorithms. **(8 Hrs)**

Text Books(TB):

1. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C", Cengage Learning, ISBN 9788131503140.
2. A. Michael Berman, "Data structures via C++", Oxford University Press, 2002, ISBN-0-19-510843-4.

Reference Books(RB):

1. E. Horowitz, S. Sahni, D. Mehta "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, 1995, ISBN 16782928.
2. Y. Langsam, M. Augenstin and A. Tannenbaum, "Data Structures using C and C++", 2nd Edition, Prentice Hall of India, 2002, ISBN-81-203-1177-9.
3. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C++", Cengage Learning, ISBN 9788131504925.
4. A. Tharp ,"File organisation and processing",2008 ,Willey India edition ,9788126518685
5. A. Drozdek, "Data Structures in C++", 2nd Edition, Thomson Brookes /COLE Books, 2002, ISBN 981 – 240 – 079 – 6.
6. J. Tremblay, P. Soresan, "An introduction to data structures with Applications", 2nd edition, Tata McGraw-Hill International Editions, 1984, ISBN-0-07-462471-7.
7. M. Folk, B. Zoellick, G. Riccardi, "File Structure An Object oriented approach with C++", Pearson Education, 2002, ISBN 81 – 7808 – 131 – 8.
8. M. Weiss, "Data Structures and Algorithm Analysis in C++", 2nd edition, Pearson Education, 2002, ISBN-81-7808-670-0

210251: COMPUTER GRAPHICS

Teaching Scheme **Examination Scheme**
Lectures: 3 Hrs/week **Theory: 100 Marks**

Prerequisite: Knowledge of C Programming and Basic Data Structures & Mathematics

Learning Objectives

1. To understand basics of computer graphics
2. To give more emphasis on implementation aspect of Computer Graphics Algorithm.
3. To prepare the student for advance courses like multimedia / Computer Vision.

Teaching aid
Faculties should use LCD to demonstrate the concept of Graphics.

Unit I - 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.
Display devices, Interactive devices: Tablets, touch panels, mouse, joysticks, track balls, light pen etc., Data generating devices: Scanners and digitizers, primitive operations, display file structure, algorithms and display file interpreter, Text and line styles.
Scan conversions, lines, line segments, vectors, pixels and frame buffers, vector generation, DDA and Bresenham's line and circle drawing algorithms*, initialising, thick lines, character generation: Stroke Principle, Starburst Principle, Bit map method, display of frame buffer.
(* Scan conversion algorithms should be given mathematical treatment) **(6 Hrs)**

Unit II - Polygons
Introduction, representation, entering Polygons, Polygon filling: Seed fill, Edge fill, scan conversion algorithm, filling with patterns.
Windowing and Clipping
Introduction, viewing transforms, 2D clipping, Cohen-Sutherland outcode algorithm, Polygon Clipping, Sutherland-Hodgman algorithm, Generalized clipping. **(6 Hrs)**

Unit III- Geometric Transformations
2D Transformations
Introduction, matrices, Scaling, Rotation, homogeneous coordinates & matrix representation of 2D transformation, Translation, Co-ordinate transformation, rotation about an arbitrary point, inverse transforms and shear transforms.
3-D Transformations
Introduction, 3-D geometry, primitives, 3D transformations & matrix representation of 2D transformation, Rotation about an arbitrary axis, Concept of parallel and perspective projections, Viewing parameters, 3D clipping, 3D viewing transformations. **(6 Hrs)**

Unit – IV

Segments

Introduction, segment table, segment creation, deletion, renaming. Image transformations, raster techniques.

Animation

Conventional and computer based animation, Methods of Controlling Animations, Basic guidelines of animation, Animation languages. **(6 Hrs)**

Unit V - Hidden Surfaces and Lines

Introduction, Back-face removal algorithm, Z buffers, scan-line, Painters algorithm, Warnock algorithm, hidden line methods, binary space partition.

Light, Color and Shading

Introduction, Diffused illumination, point source illumination, shading algorithm, reflections, shadows, ray tracing, Colour models and tables, shading algorithm, transparency **(6 Hrs)**

Unit VI- Curves and Fractals

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 (3D studio, Maya, Similar tools) (6 Hrs)

Text Books:

1. S. Harrington, "Computer Graphics", 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.
2. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.

Reference Books:

1. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice", 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
2. D. Hearn, M. Baker, "Computer Graphics – C Version", 2nd Edition, Pearson Education, 2002, ISBN 81 – 7808 – 794 – 4.
3. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.

* - To be removed fro theory only to include during Practical.

210252: COMPUTER ORGANIZATION

Teaching Scheme	**Examination Scheme**
Lectures: 3 Hrs/Week	Theory: 100 Marks

Pre requisites: Digital Electronics and Logic Design (Subject Code: 210243)

Learning Objectives

1. **To understand the structure, function and characteristics of computer systems**
2. **To understand the design of the various functional units of digital computers**
3. To learn basics of Parallel Computer Architecture.

UNIT I

Computer Evolution & Arithmetic

A Brief History of computers, Designing for Performance, Von Neumann Architecture, Computer Components, Interconnection Structures, Bus Interconnection, Scalar Data Types, Fixed and Floating point numbers, Signed numbers, Integer Arithmetic, 2's Complement method for multiplication, Booths Algorithm, Hardware Implementation, Division, Restoring and Non Restoring algorithms, Floating point representations, IEEE standards, Floating point arithmetic. **(8 Hrs)**

UNIT II

The Central Processing Unit

Machine Instruction characteristics, types of operands, types of operations, Addressing, Instruction formats, Processor organization, Register Organization, Instruction cycles, Instruction pipelining, ALU – Combinational ALUs and Sequential ALUs. **(6 Hrs)**

UNIT III

The Control Unit

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

. **(6 Hrs)**

UNIT IV

Memory Organization

Characteristics of memory systems, Internal and External Memory, Types of memories: ROM: PROM, EPROM, EEPROM, RAM: SRAM, DRAM, SDRAM, RDRAM High-Speed Memories: Cache Memory, Organization and Mapping Techniques, Replacement Algorithms, Cache Coherence, MESI protocol. Virtual Memory: Main Memory allocation, Segmentation, Paging, Address Translation.

Secondary Storage: Magnetic Disk, Tape, DAT, RAID, Optical memory, CDROM, DVD **(6 Hrs)**

UNIT V

I/O Organization: Input/Output Systems, Programmed I/O, Interrupt Driven I/O, I/O channels, Direct Memory Access (DMA), Buses and standard Interfaces:

Synchronous, Asynchronous, Parallel, Serial, PCI, SCSI, USB Ports Working mechanisms of Peripherals: Keyboard, Mouse, Scanners, Video Displays, Touch Screen panel, Dot Matrix, Desk-jet and Laser Printers. OS Support: OS Overview, Scheduling and Memory Management **(6 Hrs)**

UNIT VI

Parallel Organizations

Superscalar Processors, Multiple Processor Organizations, Symmetric Multiprocessors, Clusters, Non-uniform Memory Access, Vector Computations, Bus allocation Schemes.

RISC: Instruction execution characteristics, use of large register file, compiler based register optimization, RISC architecture and pipelining. RISC Vs CISC **(8 Hrs)**

Text Books:

1. W. Stallings, "Computer Organization and Architecture: Designing for performance", 6th Edition, Prentice Hall of India, 2003, ISBN 81 – 203 – 2103 – 0
2. C. Hamacher, V. Zvonko, S. Zaky, "Computer Organization", McGraw Hill, 2002

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 (Chapter: 2,3,4,5,7,8,9,10,11,12,13,14).
3. A. Tanenbaum, "Structured Computer Organization", 4th Ed, Prentice Hall of India, 1991 ISBN 81 – 203 – 1553 – 7 (Chapter: 1,4,5,6,8).
4. G. George, "Computer Organization: Hardware and Software", 2nd Edition, Prentice Hall of India, 1986 (Chapter: 3,4,5).
5. D. Paterson, J. Hennesy, "Computer Organization and Design: The Hardware Software Interface", 2nd Edition, Morgan Kauffman, 2000 ISBN 981 – 4033 – 588.

210253: OBJECT ORIENTED PROGRAMMING AND GRAPHICS LABORATORY

Teaching Scheme	**Examination Scheme**
Theory: 2 Hrs/Week	Practical: 50 marks
Practical: 2 Hrs/week	Term Work: 50 marks

1: Introduction to Object Oriented Programming
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. **(3 Hrs)**

2: Programming with C++
++: 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 **(2 Hrs)**

3: Classes and Objects: 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 **(4 Hrs)**

4: Operator Overloading
Introduction, Need of operator overloading, overloading the assignment, binary and unary operators, overloading using friends, rules for operator overloading, type conversions **(3 Hrs)**

5: Inheritance and Polymorphism
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 **(4 Hrs)**

6: Templates: 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 **(5 Hrs)**

7: Exception Handling: Introduction, syntax for exception handling code: try-catch-throw, 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 **(2 Hrs)**

8: Files and Streams
Concept of a file, file operations, streams, opening and closing a file, detecting end-of-file, file modes, file pointer, structures and files, classes and files, sequential file processing, Error handling **(2 Hrs)**

Text Books:

1. E. Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill PublishingCompany Ltd, New Delhi ISBN 0 – 07 – 462038 – X.
2. Saurav Sahay, "Object Oriented Programming with C++", Oxford.

Reference Books:

1. R. Lafore, "The Waite Group's Object oriented Programming in C++", 3rd Edition, Galgotia Publications, 2001, ISBN 81-7515-269-9.
2. B. Stroustrup, "C++ Programming Language", 3rd Edition, Pearson Education, 1997, ISBN 0 – 201 – 32755 – 4.
3. Joyce Farrell, "Object Oriented Programming with C++", Cengage learning, India Edition.
4. Rajesh Shukla "Object Oriented Programming in C++", Wiley India Edu.

Suggested list of Assignments

GROUP A:

- **Constructor, Destructor:**

1. Create a class named weather report that holds a daily weather report with data members day_of_month, hightemp, lowtemp, amount_rain and amount_snow. The constructor initializes the fields with default values: 99 for day_of_month, 999 for hightemp,-999 for low emp and 0 for amount_rain and amount_snow. Include a function that prompts the user and sets values for each field so that you can override the default values. Write a program that creates a monthly report.

Static member functions, friend class, this pointer, inline code and dynamic memory allocation :

2. Develop an object oriented program in C++ to create a database of the personnel information system containing the following information: Name, Date of Birth, Blood group, Height, Weight, Insurance Policy number, Contact address, telephone number, driving licence no. etc Construct the database with suitable member functions for initializing and destroying the data viz constructor, default constructor, copy constructor, destructor, static member functions, friend class, this pointer, inline code and dynamic memory allocation operators-new and delete.

- **Operator overloading :**

3. Design a Class 'Complex ' with data members for real and imaginary part. Provide default and parameterized constructors. Write a program to perform arithmetic operations of two complex numbers using operator overloading (using either member functions or friend functions).

4. Write a C++ program to perform String operations
   i. = Equality
   ii. == String Copy
   iii. + Concatenation
   iv. << To display a string
   v. >> To reverse a string
   vi. Function to determine whether a string is a palindrome

   To find occurrence of a sub-string. Use Operator Overloading

- **Inheritance :**

5. Design a base class with name, date of birth, blood group and another base class consisting of the data members such as height and weight. Design one more base class consisting of the insurance policy number and contact

address. The derived class contains the data member's telephone numbers and driving licence number.

Write a menu driven program to carry out the following things:

i) Build a master table ii) Display iii) Insert a new entry

iv) Delete entry v) Edit vi) Search for a record

- **Templates :**
  6. Write a program in C++ using function template to read two matrices of different data types such as integers and floating point values and perform simple arithmetic operations on these matrices separately and display it.
- **Virtual functions & files :**
  7. Design a base class consisting of the data members such as name of the student, roll number and subject. The derived class consists of the data members subject code ,internal assessment and university examination marks. Construct a virtual base class for the item name of the student and roll number. The program should have the facilities.

  i) Build a master table ii) List a table iii) Insert a new entry

  iv) Delete old entry v) Edit an entry vi) Search for a record
- **Exception Handling :**
  8. Create a class named Television that has data members to hold the model number and the screen size in inches, and the price. Member functions include overloaded insertion and extraction operators. If more than four digits are entered for the model, if the screen size is smaller than 12 or greater than 70 inches, or if the price is negative or over $5000 then throw an integer. Write a main() function that instantiates a television object, allows user to enter data and displays the data members .If an exception is caught ,replace all the data member values with zero values.

GROUP B:

1. Assignments to understand functions available in graphics library such as,
   (a) Text and Graphics mode, initialization of graphics mode, graphics drivers, switching between text and graphics mode, error handling.
   (b) Colour, Colour Palette, Aspect ratio, Text: fonts, alignment, size, orientation and justification.
   (c) Graphics Primitives: Pixel, Line, Circle, Ellipse, Polygons, Line styles, bar graphs, Pie Charts, Histograms, filling a polygon, windowing.
   (d) Writing a Graphics Editor
2. Write a program to implement algorithm for line and circle drawing.
3. Write a program to implement algorithm for filling a polygon using scan-fill method.
4. Write a program to implement 2-D transformations.
5. Case study of any graphics tool.

- Instructor will frame assignments based on the suggested assignments as given above. Instructors are expected to incorporate variations in above list.
- Students will submit Term Work in the form of a journal that will include at least 13 assignments. Each programming assignment will consists of pseudo-algorithm, program listing with proper documentation and printout of the output.
- Practical Examination will be based on the term work and questions will be asked to judge understanding of the assignments at the time of the examination.

210254 : MICROPROCESSOR INTERFACING LABORATORY

Teaching scheme
Practical: 4 Mrs/week

Examination scheme
Practical: 50 Marks
Term work: 50 Marks

Suggested List of Assignments

Group A

1. Write 8086 Assembly language program (ALP) to add array of N hexadecimal numbers stored in the memory. Accept input from the user.
2. Write 8086 ALP to perform non-overlapped and overlapped block transfer(with and without string specific instructions). Block containing data can be defined in the data segment.
3. Write 8086 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 :
(a) HEX to BCD b) BCD to HEX (c) EXIT.
Display proper strings to prompt the user while accepting the input and displaying the result.
4. Write 8086 ALP for the following operations on the string entered by the user.
a) Calculate Length of the string b) Reverse the string
c) Check whether the string is palindrome
OR
Make your program user friendly by providing MENU like:
(a) Enter the string b) Calculate length of string c) Reverse string d) Check palindrome e) Exit
Display appropriate messages to prompt the user while accepting the input and displaying the result.

5. Write 8086 ALP to perform string manipulation. The strings to be accepted from the user is to be stored in data segment of program_l and write FAR PROCEDURES in code segment program_2 for following operations on the string:
(a) Concatenation of two strings (b) Number of occurrences of a sub-string in the given string
Use PUBLIC and EXTERN directive. Create .OBJ files of both the modules and link them to create an EXE file.

6. Write 8086 ALP to perform multiplication of two 8-bit hexadecimal numbers. Use successive addition and add and shift method. Accept input from the user.
7. Write 8087ALP to obtain:
i) Mean ii) Variance iii) Standard Deviation
For a given set of data elements defined in data segment. Also display result.

Group B

1. 8255

(a) Write 8086 ALP to convert an analog signal in the range of 0V to 5V to its corresponding digital signal using successive approximation ADC and dual slope ADC. Find resolution used in both the ADC's and compare the results.
(b) Write 8086 ALP to interface DAC and generate following waveforms on oscilloscope, (i) Square wave - Variable Duty Cycle and frequency.
(ii) Ramp wave - Variable direction, (iii) Trapezoidal wave (iv) Stair case wave

(c) Write 8086 ALP to rotate a stepper motor for given number of steps at a given angle and in the given direction of rotation based on the user choice such as
(i) If 'C' key is pressed - clockwise rotation, (ii) If 'A' key is pressed - anticlockwise rotation. (iii) If 'B' is pressed - 1/2 clockwise and Viz Anti-clock wise rotation, (iv) If 'S' key is pressed - stop rotation.
Also write routines to accelerate and de-accelerate the motor.
(d)Write 8086 ALP to print a text message on printer using Centronix parallel printer interface.

NOTE : Select any two from 8255 assignments

2. 8253

Write 8086 ALP to program 8253 in Mode 0, modify the program for hardware re-triggerable Mono shot mode. Generate a square wave with a pulse of 1 ms. Comment on the difference between Hardware Triggered and software triggered strobe mode. Observe the waveform at GATE & out pin of 1C 8254 on CRO

3. 8279

Write 8086 ALP to initialize 8279 and to display characters in right entry mode. Provide also the facility to display

- Character in left entry mode.
- Rolling display.
- Flashing display

4. 8251

Perform an experiment to establish communication between two 8251 systems A and B. Program 8251 system A in asynchronous transmitter mode and 8251 system B in asynchronous receiver mode. Write an ALP to transmit the data from system A and receive the data at system B. The requirements are as follows:

Transmission:

- message is stored as ASCII characters in the memory.
- message specifies the number of characters to be transmitted as the first byte.

Reception:

- Message is retrieved and stored in the memory.
- Successful reception should be indicated.

5. 8259

Write 8086 APL to interface 8259 in cascade mode (M/S) and demonstrate execution of ISR in following manner:
Main program will display two digits up counter. When slave IRQ interrupt occurs, it clears the counter and starts up counting again. When Master IR1 interrupt occurs, it resets the counter to FFH and starts down counting.

6. TSR Program

Write a TSR program in 8086 ALP to implement Real Time Clock (RTC). Read the Real Time from CMOS chip by suitable INT and FUNCTION and display the RTC at the bottom right corner on the screen. Access the video RAM directly in your routine.

7. TSR Program

Write a TSR program in 8086 ALP to implement Screen Saver. Screen Saver should get activated if the keyboard is idle for 7 seconds. Access the video RAM directly in your routine.

Student will submit the term work in the form of Journal consisting of minimum of 13 experiments with all seven experiments from group A and any 6 assignments from group B. Students should be exposed to theoretical aspects of Computer organization, 8086 programming, peripheral interfacing, DOS interrupts and function calls.
Practical examination will be based on the term work and questions will be asked to judge the understanding of assignments performed at the time of examination.

210255: DATA STRUCTURES LABORATORY

Teaching Scheme
Practical: 4 Hrs/week

Examination Scheme
Practical: 50 marks
Term Work: 50 marks

Suggested List of Assignments

Group A (to be implemented in C++ programming)

1. Create binary tree and perform recursive and non-recursive traversals
2. Create binary tree. Find height of the tree and print leaf nodes. Find mirror image, print original and mirror image using level-wise printing
3. Create in-order threaded binary tree and perform traversals
4. Represent graph using adjacency list and perform DFS and BFS
5. Represent graph using adjacency list or matrix and generate minimum spanning tree using Prim's algorithm
6. Implementation of B tree
7. Implementation of AVL tree
8. Implementation of direct access file using different collision resolution techniques

Group B (to be implemented in C++ and compare with STL implementation)

1. Implementation of SLL , DLL and CLL
2. Implementation of stack & queue using SLL
3. Write a program to add binary numbers (assume one bit as one number) Use STL stack.
4. Implement Dqueue (Double ended queue) using STL.
5. Use STL for Sorting and searching with user-defined records such as Person Record (Name, birth date, telephone no), item record (item code, item name, quantity and cost)

Group C

1. Write a C++ program to implement a small database mini project to understand persistent objects and operations on sequential files (eg library information, inventory systems, automated banking system, reservation systems etc.) For example, write a program to create a database for reservation system using information such as Name, sex, age, starting place of journey and destination. Program should have following facilities
   a) To display entire passenger list b) To display particular record
   c) To update record d) To delete and sort record
   Use Exception Handling for data verification

Instructions:

- Instructor will frame assignments based on the assignments as given above. Students will submit Term Work in the form of a journal that will include at least 16 assignments with minimum of 2 assignments based on STL. Assignments to be implemented in C++ using object oriented features. Each assignment will consists of pseudo-algorithm, analysis, program listing with proper documentation and printout of the output.
- Practical Examination will be based on the term work and questions will be asked to judge understanding of assignments performed at the time of examination.

Reference Books:

1. S. McConnell, "Code Complete", WP Publishers and Distributors.
2. J. Roberge, "Data Structures in C++, A Laboratory Course", D. C. Heath and Company, Lexington MASSACHUSETTS, TORONTO, 1995 ISBN 0-669-34719-1.
3. G. Heileman, "Data Structures, Algorithms and Object oriented Programming", Tata-McGraw-Hill, 2002, ISBN 0 – 07 –048641 – 7.
4. A. Drozdek, "Data Structures in C++", 2nd Edition, Thomson Brookes/COLE Books (Vikas Publishing House, New Delhi), 2002, ISBN 981 – 240 – 079 – 6.