Bachelor of Technology (Computer Science & Engineering)										
Scheme of Studies/Examination										
Semester III										
S. No.	**Course No.**	**Subject**	**L:T:P**	**Hours/Week**	**Credits**	**Examination Schedule (Marks)**				**Duration of Exam (Hrs)**
						Major Test	**Minor Test**	**Practical**	**Total**	
1	HS-201	Fundamentals of Management	3:0:0	3	3.0	75	25	0	100	3
2	CSE-201	Discrete Structures	3:1:0	4	3.5	75	25	0	100	3
3	CSE-203	Data Structures	3:1:0	4	3.5	75	25	0	100	3
4	CSE-205	Data Base Management Systems	3:1:0	4	3.5	75	25	0	100	3
5	CSE-207	Digital Electronics	3:1:0	4	3.5	75	25	0	100	3
6	CSE-209	Programming Languages	3:1:0	3	3.5	75	25	0	100	3
7	CSE-211	Data Structures Lab	0:0:3	3	1.5	0	40	60	100	3
8	CSE-213	Digital Electronics Lab	0:0:3	3	1.5	0	40	60	100	3
9	CSE-215	Database Management Systems Lab	0:0:3	3	1.5	0	40	60	100	3
		Total		**31**	**25**	**450**	**270**	**180**	**900**	
10	MPC 202	Energy Studies*	3:0:0	3		75	25	0	100	3

*MPC-202 is a mandatory course which will be a non credit subject and student has to get pass marks in order to qualify for the Degree award

HS-201	FUNDAMENTALS OF MANAGEMENT						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**0**	**-**	**3.0**	**75**	**25**	**100**	**3**
Purpose	**To make the students conversant with the basics concepts in management thereby leading to nurturing their managerial skills**						
COURSE OUTCOMES							
CO1	**An overview about management as a discipline and its evolution**						
CO2	**Understand the concept and importance of planning and organizing in an organization**						
CO3	**Enabling the students to know about the importance of hiring and guiding the workforce by understanding the concept of leadership and communication in detail**						
CO4	**To understand the concept and techniques of controlling and new trends in management**						

UNIT-1

Introduction to Management: Meaning, Definition, nature, importance & Functions, Management as Art, Science & Profession- Management as social System, Concepts of management-Administration

Evolution of Management Thought: Development of Management Thought- Scientific management, Administrative Theory of Management, Bureaucratic Organization, Behavioral approach (Neo Classical Theory): Human Relations Movement; Behavioral Science approach; Modern approach to management – Systems approach and contingency approach.

UNIT-II

Planning: nature, purpose and functions, types of plans, planning process, Strategies and Policies: Concept of Corporate Strategy, formulation of strategy, Types of strategies, Management by objectives (MBO), SWOT analysis, Types of policies, principles of formulation of policies

4. Organizing: nature, importance, process, organization structure: Line and Staff organization, Delegation of Authority and responsibility, Centralization and Decentralization, Decision Making Process , Decision Making Models, Departmentalization: Concept and Types (Project and Matrix), formal & informal organizations

UNIT-III

Staffing: concept, process, features; manpower planning; Job Analysis: concept and process; Recruitment and selection: concept, process, sources of recruitment; performance appraisal, training and development

Directing: Communication- nature, process, formal and informal, barriers to Effective Communication, Theories of motivation-Maslow, Herzberg, Mc Gregor ; Leadership – concept and theories, Managerial Grid, Situational Leadership. Transactional and Transformational Leadership

UNIT-IV

Controlling: concept, process, types, barriers to controlling, controlling Techniques: budgetary control, Return on investment, Management information system-MIS , TQM-Total Quality Management, Network Analysis- PERT and CPM.

Recent Trends in Management: -

Social Responsibility of Management–Management of Crisis, Total Quality Management, Stress Management, ., Concept of Corporate Social Responsibility (CSR) and business ethics.

Functional aspects of business: Conceptual framework of functional areas of management- Finance; Marketing and Human Resources

Text books

1. Management Concepts - Robbins, S.P; Pearson Education India
2. Principles of Management - Koontz & O'Donnel; (McGraw Hill)

Recommended books

1. Business Organization and Management – Basu ; Tata McGraw Hill
2. Management and OB-- Mullins; Pearson Education
3. Essentials of Management – Koontz, Tata McGraw-Hill
4. Management Theory and Practice – Gupta, C.B; Sultan Chand and Sons, new Delhi
5. Prasad, Lallan and S.S. Gulshan. Management Principles and Practices. S. Chand & Co. Ltd., New Delhi.
6. Chhabra, T.N. Principles and Practice of Management. Dhanpat Rai & Co., Delhi.
7. Organizational behavior – Robins Stephen P; PHI.

CSE-201	Discrete Structures						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3**
Purpose	**To provide the conceptual knowledge of Discrete structure.**						
Course Outcomes							
CO 1	**To study various fundamental concepts of Set Theory and Logics.**						
CO 2	**To study and understand the Relations, diagraphs and lattices.**						
CO 3	**To study the Functions and Combinatorics.**						
CO 4	**To study the Algebraic Structures.**						

Unit 1 Set Theory & Logic

Fundamentals - Sets and subsets, Venn Diagrams, Operations on sets, Laws of Set Theory, Power Sets and Products, Partition of sets, The Principle of Inclusion- Exclusion.

Logic : Propositions and Logical operations, Truth tables, Equivalence, Implications, Laws of Logic, Normal forms, Predicates and quantifiers, Mathematical Induction.

Unit 2: Relations, diagraphs and lattices

Product sets and partitions, relations and diagraphs, paths in relations and diagraphs, properties of relations, equivalence and partially ordered relations, computer representation of relations and diagraphs, manipulation of relations, Transitive closure and Warshall's algorithm, Posets and Hasse Diagrams, Lattice.

Unit 3 Functions and Combinatorics

Definitions and types of functions: injective, subjective and bijective, Composition, identity and inverse, Review of Permutation and combination-Mathematical Induction, Pigeon hole principle, Principle of inclusion and exclusion, Generating function-Recurrence relations.

Unit 4: Algebraic Structures

Algebraic structures with one binary operation - semi groups, monoids and groups, Product and quotient of algebraic structures, Isomorphism, homomorphism, automorphism, Cyclic groups, Normal sub group, codes and group codes, Ring homomorphism and Isomorphism**.**

Books:

- Elements of Discrete Mathematics C.L Liu, 1985, Reprinted 2000, McGraw Hill
- Discrete mathematical structures by B Kolman RC Busby, S Ross PHI Pvt. Ltd.

- **Engineering Mathematics Through Applications, by Paras Ram CBS Publishers & Distributors**
- Discrete Mathematical Structures with Applications to Computer Science , by Tremblay J.P, and Manohar R., McGraw Hill Book Company, 1975, International Edition, 1987.
- Discrete and Combinatorial mathematics ", Ralph P., Grimaldi, Addison-Wesley Publishing Company, Reprinted in 1985.
- Discrete Mathematics and its Applications ", Kenneth H.Rosen, McGraw Hill Book Company, 1999. Sections: 7.1 to 7.5.
- Discrete Mathematics for computer scientists and Mathematicians, Joe L. Mott, Abraham

CSE-207	Digital Electronics						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3 Hour**
Purpose	**To learn the basic methods for the design of digital circuits and provide the fundamental concepts used in the design of digital systems.**						
Course Outcomes							
CO 1	**To introduce basic postulates of Boolean algebra and shows the correlation between Boolean expressions**						
CO 2	**To introduce the methods for simplifying Boolean expressions**						
CO 3	**To outline the formal procedures for the analysis and design of combinational circuits and sequential circuits**						
CO 4	**To introduce the concept of memories and programmable logic devices.**						

UNIT I MINIMIZATION TECHNIQUES AND LOGIC GATES

Minimization Techniques: Boolean postulates and laws - De-Morgan's Theorem, Principle of Duality, Boolean expression - Minimization of Boolean expressions, Minterm, Maxterm, Sum of Products (SOP), Product of Sums (POS), Karnaugh map Minimization - Don't care conditions, Quine - McCluskey method of minimization.

Logic Gates: AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR- Implementations of Logic Functions using gates, NAND-NOR implementations - Multi level gate implementations- Multi output gate implementations. TTL and CMOS Logic and their characteristics, Tristate gates.

UNIT II COMBINATIONAL CIRCUITS

Design procedure - Half adder, Full Adder, Half subtractor, Full subtractor , Parallel binary adder, parallel binary Subtractor, Fast Adder, Carry Look Ahead adder, Serial Adder/Subtractor, BCD adder, Binary Multiplier, Binary Divider, Multiplexer/ De-multiplexer, decoder, encoder, parity checker, parity generators, code converters, Magnitude Comparator.

UNIT III SEQUENTIAL CIRCUITS

Latches, Flip-flops - SR, JK, D, T, and Master-Slave - Characteristic table and equation, Application table, Edge triggering, Level Triggering, Realization of one flip-flop using other flip-flops, serial adder/subtractor, Asynchronous Ripple or serial counter, Asynchronous Up/Down counter, Synchronous counters, Synchronous Up/Down counters, Programmable counters, Design of Synchronous counters: state diagram, State table, State minimization, State assignment, Excitation table and maps-Circuit implementation, Modulo-n counter, 555 Timer, Registers - shift registers, Universal shift registers, Shift register counters, Ring counter, Shift counters, Sequence generators.

UNIT IV MEMORY DEVICES

Classification of memories - ROM: ROM organization, PROM, EPROM, EEPROM, EAPROM, RAM: - RAM organization - Write operation, Read operation, Memory cycle, Timing wave forms, Memory decoding, memory expansion, Static RAM Cell, Bipolar RAM cell, MOSFET RAM cell structure, Dynamic RAM cell structure, Programmable Logic Devices - Programmable Logic Array (PLA), Programmable Array Logic (PAL), Implementation of PLA, PAL using ROM. Introduction to Field Programmable Gate Arrays (FPGA).

TEXT BOOKS

1. Donald P. Leach and Albert Paul Malvino, Digital Principles and Applications, 8th Edition, TMH, 2003.M.

2. Morris Mano, Digital Design, 3rd Edition, Prentice Hall of India Pvt. Ltd., 2003 / Pearson Education (Singapore) Pvt. Ltd., New Delhi, 2003.

REFERENCES

1. A.K. Maini, Digital Electronics, Wiley India

2. John F. Wakerly, Digital Design, Fourth Edition, Pearson/PHI, 2006

2. John. M Yarbrough, Digital Logic Applications and Design, Thomson Learning, 2002.

3. S. Salivahanan and S. Arivazhagan, Digital Circuits and Design, 3rd Edition., Vikas Publishing House Pvt. Ltd, New Delhi, 2006

4. William H. Gothmann, Digital Electronics, 2nd Edition, PHI, 1982.

5. Thomas L. Floyd, Digital Fundamentals, 8th Edition, Pearson Education Inc, New Delhi, 2003

6. Donald D. Givone, Digital Principles and Design, TMH, 2003.

CSE-205	DATA BASE MANAGEMENT SYSTEMS						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3 Hour**
Purpose	**To familiarize the students with Data Base Management system**						
Course Outcomes							
CO 1	To provide introduction to relational model.						
CO 2	To learn about ER diagrams.						
CO 3	To understand about Query Processing and Transaction Processing.						
CO 4	To understand about the concept of functional dependencies.						
CO 5	To learn the concept of failure recovery.						
CO 6	To understand the concurrency control.						

UNIT I

INTRODUCTION Concept & Overview of DBMS, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS.

Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features.

UNIT II

THE RELATIONAL DATA MODEL & ALGEBRA

Relational Model: Structure of relational Databases, Relational Algebra, Relational Calculus, introduction to Views, updates on views

SQL and Integrity Constraints: Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints, assertions, views, Nested Sub queries, Database security application development using SQL, Stored procedures and triggers.

UNIT III

Relational Database Design:

Functional Dependency, Different anomalies in designing a Database., Normalization using functional dependencies, Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF

Internals of RDBMS:

Physical data structures, Query optimization: join algorithm, statistics and cost base optimization. Transaction processing, Concurrency control and Recovery Management : transaction model properties, state serializability, lock base protocols, two phase locking.

UNIT IV

FAILURE RECOVERY AND CONCURRENCY CONTROL

Issues and Models for Resilient Operation -Undo/Redo Logging-Protecting against Media Failures.

CONCURRENCY CONTROL: Serial and Serializable Schedules-Conflict Serializability –Enforcing Serializability by Locks-Locking Systems with Several Lock Modes-Concurrency Control by Timestamps, validation.

TRANSACTION MANAGEMENT: Serializability and Recoverability-View, Serializability-Resolving Deadlocks-Distributed Databases: Commit and Lock

Text Books;

1. Ramez Elmasri , Shamkant B. Navathe ,"Fundamentals of Database systems", Pearson
2. Korth, Silberschatz, Sudarshan: database concepts, MGH,

Reference Books:

1. R. Ramakrishnan and J. Gehrks database management system; MGH, International edition,

2 C. J. Date, data base systems: 7th edition, Addison Wesley, Pearson Education,

Chakrabarti, Advance database management systems , Wiley Dreamtech

CSE-209	Programming Languages						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3 Hour**
Purpose	To introduce the principles and paradigms of programming languages for design and implement the software intensive systems.						
Course Outcomes (CO)							
CO 1	To introduce the basic concepts of programming language, the general problems and methods related to syntax & semantics.						
CO 2	To introduce the structured data objects, subprograms and programmer defined data types.						
CO 3	To outline the sequence control and data control.						
CO 4	To introduce the concepts of storage management using programming languages.						

Unit-I: Introduction, Syntax and Semantics

Introduction: A brief history, Characteristics of a good programming language, Programming language translators compiler & interpreters , Elementary data types – data objects, variable & constants, data types, Specification & implementation of elementary data types, Declarations, type checking & type conversions, Assignment & initialization, Numeric data types, enumerations, Booleans & characters.

Syntax & Semantics: Introduction, general problem of describing syntax, formal method of describing syntax, attribute grammar dynamic semantic.

Unit-II: Structured data objects, Subprograms and Programmer Defined Data Types

Structured data objects: Structured data objects & data types, specification & implementation of structured data types, Declaration & type checking of data structure, vector & arrays, records Character strings, variable size data structures, Union, pointer & programmer defined data objects, sets, files.

Subprograms and Programmer Defined Data Types: Evolution of data type concept abstraction, encapsulation & information hiding, Subprograms, type definitions, abstract data types, over loaded subprograms, generic subprograms.

Unit–III: Sequence Control and Data Control

Sequence Control: Implicit & explicit sequence control, sequence control within expressions, sequence control within statement, Subprogram sequence control: simple call return, recursive subprograms, Exception & exception handlers, co routines, sequence control. Concurrency – subprogram level concurrency, synchronization through semaphores, monitors & message passing

Data Control: Names & referencing environment, static & dynamic scope, block structure, Local data & local referencing environment, Shared data: dynamic & static scope, Parameter & parameter transmission schemes.

Unit-IV: Storage Management and Programming Languages

Storage Management: Major run time elements requiring storage, programmer and system controlled storage management & phases, Static storage management, Stack based storage management, Heap storage management, variable & fixed size elements.

Programming Languages: Introduction to procedural, non-procedural, structured, logical, functional and object oriented programming language, Comparison of C & C++ programming languages.

Text Books:

1. Terrence W. Pratt, Marvin V. Zelkowitz, Programming Languages Design & Implementation, Pearson.
2. Allen Tucker & Robert Noonan, Programming Languages–Principles and Paradigms, Tata McGraw-Hill, 2009.

Reference Books:

1. Ellis Horowitz, Fundamentals of Programming Languages, Galgotia Publications, 2010.
2. C. Ghezzi, Programming Languages Concepts, Wiley Publications, 2010.

CSE-203	Data Structures						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3 Hour**
Purpose	**To introduce the principles and paradigms of Data Structures for design and implement the software systems logically and physically**						
Course Outcomes (CO)							
CO 1	**To introduce the basic concepts of Data structure , basic data types ,searching and sorting based on array data types.**						
CO 2	**To introduce the structured data types like Stacks and Queue and its basic operations's implementation.**						
CO 3	**To introduces dynamic implementation of linked list.**						
CO 4	**To introduce the concepts of Tree and graph and implementation of traversal algorithms.**						

Unit-1

Introduction to Data Structures, Data Types, Built in and User Defined Data Structures, Applications of Data Structure, Algorithm Analysis, Worst, Best and Average Case Analysis, Notations of Space and Time Complexity, **Arrays**, One Dimensional Arrays, Two Dimensional Arrays and Multi-Dimensional Arrays, Sparse Matrices, Storage Class, Basics of Recursion.

Searching from array using Linear and Binary Searching Algorithm, Sorting of array using Selection, Insertion, Bubble, Radix Algorithm

Unit-2

Stacks: Definition, Implementation of Stacks and Its Operations, Evaluation of Infix, prefix and Postfix Expression, Inter-conversion of Infix Expression, Prefix and Post-Fix Expression, Implementation of Merge Sort and Quick Sort Algorithm.

Queues: Definition, Sequential Implementation of Linear Queues and Its Operations, Circular Queue and Its Implementation, Priority Queues and Its Implementation, Applications of queues.

Unit-3

Linked Lists: Dynamic Implementations, Need of Dynamic Data Structures, Single Link List and Its Dynamic Implementation, Traversing, Insertion, Deletion Operations on Single Link Lists. Comparison between Static and Dynamic, Implementation of Linked List. Dynamic Implementation of Stacks and Queues.

Circular Link Lists and Doubly Link List, Dynamic Implementation of Primitive Operations on Doubly Linked Lists and Circular Link List.

Unit-4

Trees: Definition, Basic Terminology, Binary Tree, External and Internal Nodes, Static and Dynamic Implementation of a Binary Tree, Primitive Operations on Binary Trees, Binary Tree Traversals: Per-Order, In-Order And Post-Order Traversals. Representation of Infix, Post-Fix and Prefix Expressions using Trees.

Introduction to Binary Search Trees: B trees, B+ trees, AVL Trees, Threaded Binary trees, Balanced Multi-way search trees, Implementation of Heap Sort Algorithm.

Graphs: Basic Terminology, Definition of Undirected & Directed Graphs, Memory Representation of Graphs, Minimum-Spanning Trees, Warshal Algorithm, Graph Traversals Algorithms: Breadth First and Depth First,.

Text Book:

- Theory & Problems of Data Structures by Jr. Symour Lipschetz, Schaum's outline by TMH

•Fundamentals of Data structures by Ellis Horowitz & Sartaj Sahni, Pub, 1983,AW

- Shukla, Data Structures using C++, Wiley India
- Data Structures and Algorithms by A.V. Aho, J.E. Hopcroft and T.D. Ullman, Original edition, Addison-Wesley, 1999, Low Priced Edition.
- Fundamentals of computer algorithms by Horowitz Sahni and Rajasekaran.
- Data Structures and Program Design in C by Robert Kruse, PHI,
- Shukla, Data Structures using C++, Wiley India
- Introduction to Computers Science -An algorithms approach , Jean Paul Tremblay, Richard B. Bunt, 2002, T.M.H.
- Data Structure and the Standard Template library – Willam J. Collins, 2003, T.M.H

CSE-211	Data Structures Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Practical**	**Total**	**Time**
0	**0**	**3**	**1.5**	**40**	**60**	**100**	**3**
Purpose	**To introduce the principles and paradigms of Data Structures for design and implement the software systems logically and physically**						
Course Outcomes (CO)							
CO 1	**To introduce the basic concepts of Data structure, basic data types, searching and sorting based on array data types.**						
CO 2	**To introduce the structured data types like Stacks and Queue and its basic operation's implementation.**						
CO 3	**To introduces dynamic implementation of linked list.**						
CO 4	**To introduce the concepts of Tree and graph and implementation of traversal algorithms.**						

1. Write a program for Binary serach methods.
2. Write a program for insertion sort, selection sort and bubble sort.
3. Write a program to implement Stack and its operation.
4. Write a program for quick sort.
5. Write a program for merge sort.
6. Write a program to implement Queue and its operation.
7. Write a program to implement Circular Queue and its operation.
8. Write a program to implement singly linked list for the following operations: Create, Display, searching, traversing and deletion.
9. Write a program to implement doubly linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.
10. Write a program to implement circular linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.
11. Write a program to implement insertion, deletion and traversing in B tree

NOTE:
At least seven experiments are to be performed from above list and the concerned institution as per the scope of the syllabus may set remaining eight.

CSE-213	Digital Electronics Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Practical**	**Total**	**Time**
0	**0**	**3**	**1.5**	**40**	**60**	**100**	**3**
Purpose	**To learn the basic methods for the design of digital circuits and systems.**						
Course Outcomes							
CO 1	**To Familiarization with Digital Trainer Kit and associated equipment.**						
CO 2	**To Study and design of TTL gates**						
CO 3	**To learn the formal procedures for the analysis and design of combinational circuits.**						
CO 4	**To learn the formal procedures for the analysis and design of sequential circuits**						

LIST OF EXPERIMENTS:

1. Familiarization with Digital Trainer Kit and associated equipment.
2. Study of TTL gates AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
3. Design and realize a given function using K-Maps and verify its performance.
4. To verify the operation of Multiplexer and De-multiplexer.
5. To verify the operation of Comparator.
6. To verify the truth table of S-R, J-K, T, D Flip-flops.
7. To verify the operation of Bi-directional shift register.
8. To design and verify the operation of 3-bit asynchronous counter.
9. To design and verify the operation of asynchronous Up/down counter using J-K FFs.
10. To design and verify the operation of asynchronous Decade counter.
11. Study of TTL logic family characteristics.
12. Study of Encoder and Decoder.
13. Study of BCD to 7 segment Decoder.

NOTE:

At least ten experiments are to be performed from above list and the concerned institution as per the scope of the syllabus may set remaining five.

CSE-215	DATABASE MANAGEMENT SYSTEMS LAB						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Practical**	**Total**	**Time**
-	**-**	**3**	**1.5**	**40**	**60**	**100**	**3**
Purpose	**To familiarize the students with the basics of Operating Systems**						
Course Outcomes							
CO1	To understand basic DDL commands						
CO 2	To learn about DML and DCL commands						
CO 3	To understand the sql queries using SQL operators						
CO 4	To understand the concept of relational algebra						
CO5	To learn various queries using date and group functions						
CO6	To understand the nested queries						
CO7	To learn view, cursors and triggers.						

1. Write the queries for Data Definition Language (DDL) in RDBMS.
2. Write the queries for Data Manipulation Language (DML) in RDBMS.
3. Write the queries for Data Control Language (DCL) in RDBMS.
4. Write SQL queries using logical operations (=,,etc)
5. Write SQL queries using SQL operators
6. Write SQL query using character, number, date and group functions
7. Write SQL queries for relational algebra
8. Write SQL queries for extracting data from more than one table
9. Write SQL queries for sub queries, nested queries
10. Concepts for ROLL BACK, COMMIT & CHECK POINTS
11. Create VIEWS, CURSORS and TR
12. High level language extension with Cursors.
13. High level language extension with Triggers.
14. To study the concept of Procedures and Functions..

Bachelor of Technology (Computer Science & Engineering)

Scheme of Studies/Examination

Semester IV

S. No.	Course No.	Subject	L:T:P	Hours/ Week	Credits	Examination Schedule (Marks)				Duration of Exam (Hrs)
						Major Test	Minor Test	Practical	Total	
1	AS-201	Mathematics-III	3:1:0	4	3.5	75	25	0	100	3
2	CSE-202	Object Oriented Programming	3:1:0	4	3.5	75	25	0	100	3
3	CSE-204	Internet Fundamental	3:0:0	3	3.0	75	25	0	100	3
4	CSE-206	Digital Data Communication	3:1:0	4	3.5	75	25	0	100	3
5	CSE-208	Microprocessor & Interfacing	3:1:0	4	3.5	75	25	0	100	3
6	CSE-210	Operating System	3:1:0	4	3.5	75	25	0	100	3
7	CSE-212	Object Oriented Programming Lab	0:0:3	3	1.5	0	40	60	100	3
8	CSE-214	Microprocessor Lab	0:0:3	3	1.5	0	40	60	100	3
9	CSE-216	Internet Lab	0:0:3	3	1.5	0	40	60	100	3
		Total		**32**	**25.0**	**450**	**270**	**180**	**900**	

10	MPC 201	Enviornment Studies*	3:0:0	3		75	25		100	3

*MPC-201 is a mandatory course which will be a non credit subject and student has to get pass marks in order to qualify for the Degree award

AS-201	Mathematics-III						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3**
Purpose	**To provide the conceptual knowledge of Engineering mathematics**						
Course Outcomes							
CO 1	**To study various fundamental concepts of Fourier series and Fourier Transformation.**						
CO 2	**To study and understand the functions of a complex variables.**						
CO 3	**To study the Probability Distributions.**						
CO 4	**To study the linear programming problem formulation.**						

UNIT – I

Fourier series: Euler's Formulae, Conditions for Fourier expansions, Fourier expansion of functions having points of discontinuity, change of interval, Odd & even functions, Half-range series.

Fourier Transforms: Fourier integrals, Fourier transforms, Fourier cosine and sine transforms. Properties of Fourier transforms, Convolution theorem, Perseval's identity, Relation between Fourier and Laplace transforms, Fourier transforms of the derivatives of a function, Application to boundary value problems.

UNIT-II

Functions of Complex Variables: Functions of a complex variable, Exponential function, Trigonometric, Hyperbolic and Logarithmic functions, limit and continuity of a function, Differentiability and analyticity.

Cauchy-Riemann equations, Necessary and sufficient conditions for a function to be analytic, Polar form of the Cauchy-Riemann equations, Harmonic functions, Application to flow problems, Conformal transformation, Standard transformations (Translation, Magnification & rotation, inversion & reflection, Bilinear).

UNIT-III

Probability Distributions : Probability, Baye's theorem, Discrete & Continuous probability distributions, Moment generating function, Probability generating function, Properties and applications of Binomial, Poisson and normal distributions.

UNIT-IV

Linear Programming : Linear programming problems formulation, Solution of Linear Programming Problem using Graphical method, Simplex Method, Dual-Simplex Method.

Text Book

1. Higher Engg. Mathematics: B.S. Grewal
2. Advanced Engg. Mathematics: E. Kreyzig

Reference Book

1. Engineering Mathematics Through Applications, by Paras Ram CBS Publishers & Distributors
2. Complex variables and Applications: R.V. Churchil; Mc. Graw Hill
3. Engg. Mathematics Vol. II: S.S. Sastry; Prentice Hall of India.
4. Operation Research: H.A. Taha
5. Probability and statistics for Engineer: Johnson. PHI.

CSE-202	Object Oriented Programming						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3 Hour**
Purpose	**To introduce the principles and paradigms of Object Oriented Programming Language for design and implement the Object Oriented System**						
Course Outcomes (CO)							
CO 1	**To introduce the basic concepts of object oriented programming language and the its representation**						
CO 2	**To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.**						
CO 3	**To introduce polymorphism, interface design and overloading of operator.**						
CO 4	**To handle backup system using file, general purpose template and handling of raised exception during programming**						

Unit–1

Introduction to C++, C++ Standard Library, Illustrative Simple C++ Programs. Header Files, Namespaces, Application of object oriented programming.

Object Oriented Concepts, Introduction to Objects and Object Oriented Programming, Encapsulation, Polymorphism, Overloading, Inheritance, Abstract Classes, Accessifier (public/ protected/ private), Class Scope and Accessing Class Members, Controlling Access Function, Constant, Class Member, Structure and Class

Unit-2

Friend Function and Friend Classes, This Pointer, Dynamic Memory Allocation and Deallocation (New and Delete), Static Class Members, Constructors, parameter Constructors and Copy Constructors, Deconstructors,

Introduction of inheritance, Types of Inheritance, Overriding Base Class Members in a Derived Class, Public, Protected and Private Inheritance, Effect of Constructors and Deconstructors of Base Class in Derived Classes.

Unit-3

Polymorphism, Pointer to Derived class, Virtual Functions, Pure Virtual Function, Abstract Base Classes, Static and Dynamic Binding, Virtual Deconstructors.

Fundamentals of Operator Overloading, Rules for Operators Overloading, Implementation of Operator Overloading Like <<,>> Unary Operators, Binary Operators.

Unit-4

Text Streams and binary stream, Sequential and Random Access File, Stream Input/ Output Classes, Stream Manipulators.

Basics of C++ Exception Handling, Try, Throw, Catch, multiple catch, Re-throwing an Exception, Exception specifications.

Templates: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non- Type Template arguments.

Text Books:

- Object Oriented Programming in Turbo C++ by Robert Lafore, 1994, The WAITE Group Press.
- The complete reference C ++ by Herbert shieldt Tata McGraw Hill

- Shukla, Object Oriented Programming in c++, wiley india
- C++ How to Program by H M Deitel and P J Deitel, 1998, Prentice Hall
- Programming with C++ By D Ravichandran, 2003, T.M.H

CSE-204	Internet Fundamentals						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**0**	**-**	**3**	**75**	**25**	**100**	**3**
Purpose	**To provide the conceptual knowledge of Internet and methodologies used in web and secure internet communication and networking.**						
Course Outcomes							
CO 1	**To study various fundamental concepts of Internetworking techniques with their characteristics.**						
CO 2	**To study and understand the requirements for world-wide-web formats and techniques.**						
CO 3	**To study the E-mail functioning and basics of HTML, XML and DHTML languages.**						
CO 4	**To study the functioning of Servers and Privacy and Security related mechanisms.**						

UNIT-1 : THE INTERNET

Introduction to networks and internet, history, Internet, Intranet & Extranet, Working of Internet, Internet Congestion, internet culture, business culture on internet. Collaborative computing & the internet. Modes of Connecting to Internet, Internet Service Providers(ISPs), Internet address, standard address, domain name, DNS, IP.v6.Modems, Speed and time continuum, communications software; internet tools.

UNIT-II : WORLD WIDW WEB

Introduction, Miscellaneous Web Browser details, searching the www: Directories search engines and meta search engines, search fundamentals, search strategies, working of the search engines, Telnet and FTP, HTTP, Gophar Commands, TCP/IP. Introduction to Browser, Coast-to-coast surfing, hypertext markup language, Web page installation, Web page setup, Basics of HTML & formatting and hyperlink creation.Using FrontPage Express, Plug-ins.

UNIT-III : INTERNET PLATEFORM AND MAILING SYSTEMS

Introduction, advantages and disadvantages, User Ids, Pass words, e-mail addresses, message components, message composition, mailer features, E-mail inner workings, E-mail management, MIME types, Newsgroups, mailing lists, chat rooms, secure-mails, SMTP, PICO, Pine, Library cards catalog, online ref. works.
Languages: Basic and advanced HTML, Basics of scripting languages – XML, DHTML, Java Script.

UNIT-IV : SERVERS

Introduction to Web Servers: PWS, IIS, Apache; Microsoft Personal Web Server. Accessing & using these servers.
Privacy and security topics: Introduction, Software Complexity, Attacks, security and privacy levels, security policy, accessibility and risk analysis, Encryption schemes, Secure Web document, Digital Signatures, Firewalls, Intrusion detection systems

Text Book:

- Fundamentals of the Internet and the World Wide Web, Raymond Greenlaw and Ellen Hepp, TMH- 2012
- Internet & World Wide Programming, Deitel,Deitel & Nieto, 2012, Pearson Education

Reference Books:

- Complete idiots guide to java script,. Aron Weiss, QUE, 2013
- Network firewalls, Kironjeet syan -New Rider Pub.2014
- Networking Essentials – Firewall Media.Latest-2015
- www.secinf.com
- www.hackers.com
- Alfred Glkossbrenner-Internet 101 Computing MGH, 2013

CSE-206	Digital Data Communication						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3**
Purpose	**To provide the conceptual knowledge of data preparation and signal transmission methodologies used in data communication and networking.**						
Course Outcomes							
CO 1	**To study various analog communication techniques and with their characteristics.**						
CO 2	**To study and understand the requirements for analog/digital data to analog/digital signal conversion techniques.**						
CO 3	**To study the error and flow control techniques in communication and networking.**						
CO 4	**To study the concept of multiplexing and applied multiple access techniques specially in satellite communication.**						

UNIT-1

MODULATION TECHNIQUES

Basic constituents of Communication Systems need of modulation, Amplitude modulation, spectrum of AM wave, modulation index, DSBSC modulation, SSB Modulation, vestigial side band modulation.

ANGLE MODULATION: Frequency and Phase Modulation, spectrum of FM Wave, modulation index and Bandwidth of FM Signal, NBFM and WBFM.

UNIT-II

DATA ENCODING

Digital data, Digital signals: Encoding schemes: NRZ-L, NRZ-I, Manchester-Diff-Manchester-encoding,Pseudoternary-Bipolar-AMI,B8ZS- HDB3 – Evaluation factors-Digital data, analog signals: Encoding Techniques –ASK-FSK-PSK-QPSK-Performance comparison-Analog data, digital signals: Quantization- Sampling theorem-PCM-Delta modulation-Errors- comparison- Analog Data, analog signals: Need for modulation -0 Modulation methods – Amplitude modulation- Angle modulation- Comparison.

UNIT-III

DIGITAL DATA COMMUNICATION TECHNIQUES

Asynchronous and synchronous transmission –Error Detection techniques: Parity checks – Cycle redundancy checks-Checksum-Error Correcting codes: Forwards and backward error corrections, Transmission media. Communication Topologies.

DTE & DCE interface: Characteristics of DTE-DCE interface. Interfaces: Rs-232-C, Rs-449/422, A/423-A.

UNIT-IV

SATELITE COMMUNICATION

Multiplexing: Advantages – Types of Multiplexing – FDM – Synchronous TDM – Statistical TDM or Asynchronous TDM, Study of their characteristics.

Satellite Communication Systems: Satellite parameters and configurations – Capacity allocation, Frequency Division FDMA; Time Division TDMA- Fixed assigned multiple access (FAMA), Demand assign multiple access (DAMA) – The concept of spread spectrum: FHSS, DSSS – CDMA – Transmission and reception.

TEXT BOOKS

1. Forouzen, “Data Communication & Networking”,Tata Mcgraw Hill
2. Proakin, “Digital Communications”,Mc Graw Hill.
3. W. Stalling, “Wireless Communication and Networks” Pearson.

REFERENCES

1. Stallings, “Data & computer Communications”,PHI.
2. Roden, “Digital & Data Communication Systems”,PHI.
3. Irvine, Data communications & Networks An engineering approach, wiley india

CSE-208	Microprocessor & Interfacing						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3**
Purpose	**To learn the architecture and programming of Intel family microprocessors and its interfacing.**						
Course Outcomes							
CO 1	**To study the Architecture of 8085 microprocessors**						
CO 2	**To learn the architecture 8086 Microprocessor and its interfacing to memories**						
CO 3	**To learn the instruction set of 8086 Microprocessor and assembly language programming of 8086 Microprocessor.**						
CO 4	**To learn interfacing of interrupts, basic I/O and DMA with 8086 Microprocessor**						

Unit I

Evolution of Microprocessor, Introduction to 8085 - 8085 architecture - Pin Details - Addressing Modes -Instruction Set and Assembler Directives, Instruction Timing Diagram.

UNIT-II

8086 CPU ARCHITECTURE: 8086 Block diagram; description of data registers, address registers; pointer and index registers, PSW, Queue, BIU and EU. 8086 Pin diagram descriptions. Generating 8086 CLK and reset signals using 8284. WAIT state generation. Microprocessor BUS types and buffering techniques, 8086 minimum mode and maximum mode CPU module. MAIN MEMORY SYSTEM DESIGN: Memory devices, 8086 CPU Read/Write timing diagrams in minimum mode and maximum mode. Address decoding techniques. Interfacing SRAMS; ROMS/PROMS. Interfacing and refreshing DRAMS.

UNIT-III

8086 INSTRUCTION SET: Instruction formats, addressing modes, Data transfer instructions, string instructions, logical instructions, arithmetic instructions, transfer of control instructions; process control instructions; Assembler directives.

8086 PROGRAMMING TECHNIQUES: Writing assembly Language programs for logical processing, arithmetic processing, timing delays; loops, data conversions.

UNIT-IV

BASIC I/O INTERFACE: Parallel and Serial I/O Port design and address decoding. Memory mapped I/O Vs Isolated I/O Intel's 8255 and 8251- description and interfacing with 8086. ADCs and DACs, - types, operation and interfacing with 8086. Interfacing Keyboards, alphanumeric displays, multiplexed displays, and stepper motor, optical encoder with 8086.

INTERRRUPTS AND DMA: 8086 Interrupt mechanism; interrupt types and interrupt vector table. Applications of interrupts, Intel's 8259. DMA operation. Intel's 8237.

Text Books:

- Barry B. Brey, "The Intel Microprocessor8086/8088, 80186", Pearson Education, Eighth Edition, 2009
- D.V. Hall, Microprocessors and Interfacing, McGraw Hill 2nd ed.

- Liu, Gibson, "Microcomputer Systems: The 8086/88 Family", 2nd Edition, PHI,2005
- Kenneth Ayala, "The 8086 Microprocessor: Programming & Interfacing the PC", Cengage Learning, Indian Edition, 2008
- Kip Irvine, "Assembly language for IBM PC", PHI, 2nd Edition, 1993
- Peter Abel, "Assembly language programming", Pearson Edu,5th Edition,2002
- Uffenback, "The 8086 Family Design" PHI, 2nd Edition.
- Walter A Triebel and Avtar Singh; The 8088 and 8086 Microprocessors, Programming, Interfacing, Software, Hardware and Applications, Fourth Edition, Pearson Education.

CSE-210	OPERATING SYSTEMS						
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Total**	**Time**
3	**1**	**-**	**3.5**	**75**	**25**	**100**	**3**
Purpose	**To familiarize the students with the basics of Operating Systems**						
Course Outcomes							
CO1	**To understand the structure and functions of Operating system.**						
CO 2	**To learn about processes, threads and scheduling algorithms.**						
CO 3	**To understand the principle of concurrency.**						
CO 4	**To understand the concept of deadlocks.**						
CO5	**To learn various memory management schemes.**						
CO6	**To study I/O management and file systems.**						
CO7	**To study the concept of protection and security.**						

UNIT 1

Introduction: Introduction to OS. Operating system functions, Different types of O.S.: batch process, multi-programmed, time-sharing, real-time, distributed, parallel.

System Structure: Computer system operation, I/O structure, storage structure, storage hierarchy, different types of protections, operating system structure (simple, layered, virtual machine), O/S services, system calls.

UNIT II

CPU scheduling: scheduling criteria, preemptive & non-preemptive scheduling, scheduling algorithms, algorithm evaluation, multi-processor scheduling.

Threads: overview, benefits of threads, user and kernel threads.

Process Management: Concept of processes, process states, process control, co-operating processes, inter-process communication.

Process Synchronization: background, critical section problem, critical region, synchronization hardware, Classical problems of synchronization, semaphores.

UNIT III

Deadlocks: Concept of deadlock, deadlock characterization, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock.

Memory Management: background, logical vs. physical address space, contiguous memory allocation, paging, segmentation, segmentation with paging. Concept of fragmentation.

Virtual Memory: background, demand paging, concept of page replacement, page replacement algorithms , allocation of frames, thrashing.

UNIT IV

File Systems: file concept, file organization and access methods, allocation methods, directory structure, free-space management

I/O Management: I/O hardware, polling, interrupts, DMA, kernel I/O subsystem (scheduling, buffering, caching, spooling and device reservation)

Disk Management: disk structure, disk scheduling (FCFS, SSTF, SCAN,C-SCAN) , disk reliability, disk Performance parameters

Protection & Security:

Goals of protection and security, security attacks, authentication, program threats, system threats, threat monitoring. **Case studies:** UNIX file system, Windows file system

Text Books:

1. Operating System Concepts", Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, Wiley
2. Operating Systems : Internals and Design Principles, William Stallings, Pearson

Reference books:

1. Operating systems: a concept based approach", Dhananjay M. Dhamdhere, McGraw Hill .
2. Operating Systems Design and Implementation" ,(Prentice Hall Software Series) Andrew S Tanenbaum and Albert S Woodhull
3. Taub & Schilling, Principles of Communication Systems, TMH.
4. Mithal G K, Radio Engineering, Khanna Pub.
5. Sirnon Haykin, Communication Systems, John Wiley

CSE-212	Object Oriented Programming Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Lab**	**Total**	**Time**
0	**0**	**3**	**1.5**	**40**	**60**	**100**	**3 Hour**
Purpose	**To introduce the principles and paradigms of Object Oriented Programming Language for design and implement the Object Oriented System**						
Course Outcomes (CO)							
CO 1	**To introduce the basic concepts of object oriented programming language and the its representation**						
CO 2	**To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.**						
CO 3	**To introduce polymorphism, interface design and overloading of operator.**						
CO 4	**To handle backup system using file, general purpose template and handling of raised exception during programming**						

Q1. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.

Q2. A point on the two dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:

Enter coordinates for P1: 3 4
Enter coordinates for P2: 5 7
Coordinates of P1 + P2 are : 8, 11

Q3. Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this.

Enter first number, operator, and second number: 10/ 3
Answer = 3.333333
Do another (Y/ N)? Y
Enter first number, operator, second number 12 + 100
Answer = 112
Do another (Y/ N) ? N

Q4. A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this:

- Enter your area code, exchange, and number: 415 555 1212
- My number is (212) 767-8900
- Your number is (415) 555-1212

Q5. Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results maybe a DM object or DB objects, depending on the units in which the results are required. The display should be in the format of feet and inches or metres and cenitmetres depending on the object on display.

Q6. Create a class rational which represents a numerical value by two double values- NUMERATOR & DENOMINATOR. Include the following public member Functions:

- constructor with no arguments (default).
- constructor with two arguments.
- void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator.

• Overload + operator to add two rational number.
• Overload >> operator to enable input through cin.
• Overload << operator to enable output through
cout. Write a main ( ) to test all the functions in the class.

Q7. Consider the following class

```
definition class father {
protected : int age;
public;
father (int x) {age = x;}
virtual void iam ( )
{ cout < < "I AM THE FATHER, my age is : "<< age<< end1:}
};
```

Derive the two classes son and daughter from the above class and for each, define iam ( ) to write our similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

Q8. Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.

Q9. A hospital wants to create a database regarding its indoor patients. The information to store
include a) Name of the patient
b) Date of admission
c) Disease
d) Date of discharge

Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

Q10. Make a class **Employee** with a name and salary. Make a class **Manager** inherit from **Employee**. Add an instance variable, named department, of type string. Supply a method to **to String** that prints the manager's name, department and salary. Make a class **Executive** inherits from **Manager**. Supply a method **to String** that prints the string **"Executive"** followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

Q11. Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 1. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC kay should cause the program to print out the total cars and total cash and then exit.

Q12. Write a function called reversit ( ) that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument. Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase, "Able was I ere I saw Elba)".

Q13. Create some objects of the string class, and put them in a Deque-some at the head of the Deque and some at the tail. Display the contents of the Deque using the forEach ( ) function and a user written display function. Then search the Deque for a particular string, using the first That ( ) function and display any strings that match. Finally remove all the items from the Deque using the getLeft ( ) function and display each item. Notice the order in which the items are displayed: Using getLeft ( ), those inserted on the left (head) of the Deque are removed in "last in first out" order while those put on the right side are removed in "first in first out" order. The opposite would be true if getRight ( ) were used.

Q14. Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class account that stores customer name, account number and type of account. From this derive the classes cur_acct and sav_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks:

a) Accept deposit from a customer and update the balance.
b) Display the balance.
c) Compute and deposit interest.
d) Permit withdrawal and update the balance.
e) Check for the minimum balance, impose penalty, necessary and update the balance.
f) Do not use any constructors. Use member functions to initialize the class members.

Q15. Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get_data( ) to initialize baseclass data members and another member function display_area( ) to compute and display the area of figures. Make display_area ( ) as a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively and display the area.

Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles and used as follows:

Area of rectangle = x * y

Area of triangle = ½ * x * y

CSE-214	Microprocessor Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Lab**	**Total**	**Time**
0	**0**	**3**	**1.5**	**40**	**60**	**100**	**3 Hour**
Purpose	**Write the efficient Assembly Language Program for different problem statements and implement different system interfacing.**						
Course Outcomes							
CO 1	**Understanding different steps to develop program such as Problem definition, Analysis, Design of logic, Coding, Testing, Maintenance (Modifications, error corrections, making changes etc.)**						
CO 2	**To be able to apply different logics to solve given problem.**						
CO 3	**To be able to write program using different implementations for the same problem**						
CO 4	**Use of programming language constructs in program implementation**						

Write an Assembly Language Program to

1. Add / Sub two 16 bit numbers.
2. Find sum of series of numbers.
3. Multiply two 16 bit unsigned/ signed numbers.
4. Divide two unsigned/ signed numbers (32/16 , 16/8, 16/16, 8/8)
5. Add / Sub / multiply / Divide two BCD numbers.
6. Find smallest/ largest number from array of n numbers.
7. Arrange numbers in array in ascending/ descending order.
8. Perform block transfer data using string instructions / without using string instructions.
9. Compare two strings using string instructions / without using string instructions.
10. Display string in reverse order, string length, Concatenation of two strings.
11. Convert Hex to Decimal, Decimal to Hex.
12. To find 1's and 2's complement of a number.

CSE-216	Internet Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Sessional**	**Lab**	**Total**	**Time**
0	**0**	**3**	**1.5**	**40**	**60**	**100**	**3 Hour**
Purpose	**Learn the internet and design different web pages using HTML and installation of different MODEMS.**						
Course Outcomes							
CO 1	**Understanding different PC software and their applications**						
CO 2	**To be able to learn HTML.**						
CO 3	**To be able to write Web pages using HTML.**						
CO 4	**To be able to install modems and understand the e-mail systems.**						

PC Software: Application of basics of MS Word 2000, MS Excel 2000, MS Power Point 2000, MS Access 2000, HTML

1. To prepare the Your Bio Data using MS Word
2. To prepare the list of marks obtained by students in different subjects and show with the help of chart/graph the average, min and max marks in each subject.
3. Prepare a presentation explaining the facilities/infrastructure available in your college/institute.
4. Design Web pages containing information of the Deptt.

HTML Lists:

1. Create a new document that takes the format of a business letter. Combine <P> and
 tags to properly separate the different parts of the documents. Such as the address, greeting, content and signature. What works best for each?
2. Create a document that uses multiple
 and <P> tags, and put returns between <PRE> tags to add blank lines to your document see if your browser senders them differently.
3. Create a document using the <PRE>tags to work as an invoice or bill of sale, complete with aligned dollar values and a total. Remember not to use the Tab key, and avoid using emphasis tags like <B> or <EM> within your list.
4. Create a seven-item ordered list using Roman numerals. After the fifth item, increase the next list value by 5.
5. Beginning with an ordered list, create a list that nests both an unordered list and a definition list.
6. Use the ALIGN attribute of an <IMG> tags to align another image to the top of the first image.. play with this feature, aligning images to TOP, MIDDLE and BOTTOM.
7. Create a 'table of contents' style page (using regular and section links) that loads a different document for each chapter or section of the document.

Internet:

1. Instilling internet & external modems, NIC and assign IP address.
2. Study of E-mail system.
3. Create your own mail-id in yahoo and indiatimes.com.
4. Add names (mail-id's) in your address book, compose and search an element.