

## SCHEME FOR B.TECH INFORMATION TECHNOLOGY

### THIRD SEMESTER

S.No.	Sub. Code	Subjects	L	T	P	Credits
1	ITX 201	Object Oriented Programming Concepts	3	0	0	3
2	ITX 203	Data Structures	3	0	0	3
3	ITX 205	Data Communication and Networking	3	0	0	3
4	ITX 207	Introduction to Digital Logic Design	3	0	0	3
5	MAX XXX	Mathematics-II	3	1	0	4
6	HM 201	Engineering Economics & Industrial Management	3	0	0	3
<b>Practical /Training /Projects</b>						
7	ITX 221	Object Oriented Programming Concepts Lab	0	0	2	1
8	ITX 223	Data Structures Lab	0	0	4	2
9	ITX 225	Data Communication and Networking Lab	0	0	2	1
10	ITX 227	Introduction to Digital Logic Design Lab	0	0	2	1
<b>TOTAL</b>			<b>18</b>	<b>1</b>	<b>10</b>	<b>24</b>

### FOURTH SEMESTER

S.No.	Sub. Code	Subjects	L	T	P	Credits
1	ITX 202	Computer System Architecture	3	0	0	3
2	ITX 204	Introduction to Design and Analysis of Algorithms	3	0	0	3
3	ITX 206	Fundamentals of Database Management Systems	3	0	0	3
4	ITX 208	Programming in Java	3	0	0	3
5	ITX 210	Computer Graphics	3	0	0	3
6	HM 311	Financial and Cost Accounting	3	1	0	4
<b>Practical /Training /Projects</b>						
7	ITX 222	Introduction to Design and Analysis of Algorithms Lab	0	0	2	1
8	ITX 224	Fundamentals of Database Management Systems Lab	0	0	2	1
9	ITX 226	Programming in Java Lab	0	0	4	2
10	ITX 228	Computer Graphics Lab	0	0	2	1
<b>TOTAL</b>			<b>18</b>	<b>1</b>	<b>10</b>	<b>24</b>



## COURSE CONTENT OF 3<sup>RD</sup> and 4<sup>TH</sup> SEMESTER

### THIRD SEMESTER

#### **ITX 201      OBJECT ORIENTED PROGRAMMING CONCEPTS**

**3 credits [3-0-0]**

**Object oriented thinking:** Need for OOP Paradigm, Procedural programming Vs object oriented programming, object oriented concepts.

**Functions:** Main function, function prototyping, inline functions, reference variables, call by reference, Defaults arguments, function overloading, Math library functions.

**Class:** Difference between C structure and class, specifying a class, Defining member functions: inside and outside class, scope resolution operator, Array within a class, array of objects, Static data members and member functions, Object as function arguments, returning objects, Friend function, memory allocation for objects, pointer to members, pointer to object, this pointer local classes.

**Constructor and destructor:** Constructor, types of constructors: default, parameterized and copy constructor, constructor overloading, constructor with default parameter, dynamic initialization of objects, destructor

**Operator overloading and Type Conversion:** Defining operator overloading, overloading unary and binary operator, Data Conversion: Basic to User Defined , User defined to basic, Conversion from one user-defined type to another.

**Inheritance and polymorphism:** Base class, derived class, visibility modes, derivation and friendship, Types of inheritance, Containership, virtual function binding, pure virtual functions, Abstract class, pointer to derived class.

**Console IO operations:** C++ stream classes, Unformatted IO operations, formatted IO operations, managing output with manipulators.

**Working with files:** Classes for file stream operations, opening and closing files, detectinf cof, File opening modes, file Pointers, Error handling during file operations, command line arguments. Templates: Class template, class template with parameter, function template, function template with parameter.

#### **RECOMMENDED BOOKS**

1. Bjrane Stroustrup, "C++ Programming Language", 3rd Ed., Pearson education Asia, 1997.
2. Lafore R."Object Oriented Programming in C++", 4th Ed. Techmedia, New Delhi, 2002.
3. Yashwant Kenetkar, "Let us C++", 1st Ed., Oxford University Press, 2006.
4. B.A. Forouzan and R.F. Gilberg, "Computer Science: A structured approach using C++" 2nd Ed. Cengage Learning, New Delhi, 2004.
5. E. Balagurusamy, "Object Oriented Programming with C++" 5<sup>th</sup> Ed., Tata McGraw Hill, 2011.

#### **ITX 203      DATA STRUCTURES**

**3 credits [3-0-0]**

**Introduction:** Basic Terminology, Elementary Data Organization, Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Big-Oh, Time-Space trade-off. Abstract Data Types (ADT)

**Arrays:** Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Application of arrays, Sparse Matrices and their representations.

**Linked lists:** Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition, Generalized Linked List

**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi Problem, Simulating Recursion, Principles of recursion, Tail recursion, Removal of recursion Queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.



**Trees:** Basic terminology, Binary Trees, Binary Tree Representation: Array Representation and Dynamic Representation, Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Array and Linked Representation of Binary trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Threaded Binary trees, Traversing Threaded Binary trees, Huffman algorithm.

**Graphs:** Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list, Graph Traversal : Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijkstra Algorithm, Introduction to Activity Networks.

**Searching :** Sequential Search, Binary Search, Comparison and Analysis Internal Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Two Way Merge Sort, Heap Sort, Radix Sort, Practical consideration for Internal Sorting.

**Search Trees:** Binary Search Trees (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, AVL trees, Introduction to m-way Search Trees, B Trees & B+ Trees

**Hashing:** Hash Function, Collision Resolution Strategies Storage Management: Garbage Collection and Compaction.

### RECOMMENDED BOOKS

1. Horowitz and Sahani, "Fundamentals of data Structures", Galgotia Publication Pvt. Ltd., New Delhi, 1984.
2. R. Kruse et al, "Data Structures and Program Design in C", Pearson Education Asia, Delhi, 2002.
3. A. M. Tenenbaum, "Data Structures using C & C++", Prentice-Hall of India Pvt. Ltd., New Delhi, 2004.
4. Bruno R Preiss, "Data Structures and Algorithms with Object Oriented Design Pattern in C++", Jhon Wiley & Sons, Inc., 1999.
5. Gilberg Forozan , "Data Structure – A pseudo code approach with C++", Cengage Learning, New Delhi, 2005.

### ITX 205 DATA COMMUNICATION AND NETWORKING

3 credits [3-0-0]

**Data Transmission/The Physical Layer:** Concepts and Terminology, Analog and Digital Data Transmission, Transmission Impairments, Guided Transmission Media, Wireless Transmission, Communication Satellites, The Public Switched Telephone Network, The Mobile Telephone System, Cable Television

**Data Encoding:** Digital Data: Digital and Analog Signals, Analog Data: Digital and Analog Signals, Spread Spectrum

**Data Communication Interface:** Asynchronous and Synchronous Transmission, Line Configurations, Interfacing

**Multiplexing:** Frequency-Division Multiplexing, Synchronous Time-Division Multiplexing, Statistical Time-Division Multiplexing

**Circuit Packet and Switching:** Switched Networks, Circuit-Switching Networks, Switching Concepts, Routing in Circuit-Switched Networks, Control Signaling, Packet-Switching Principles, Routing, Congestion Control, X.25 282

**Frame Relay:** Frame Relay Protocol Architecture, Frame Relay Call Control, User Data Transfer, Network Function, Congestion Control

**LAN Technology and Systems:** LAN Architecture, Bus Tree LANs, Ring LANs, Star LANs, Wireless LANs, Ethernet and Fast Ethernet (CSMA/CD), Token Ring and FDDI, 100VG-AnyLAN, ATM LANs, Fiber Channel, Wireless LANs, Bridge Operation, Routing with Bridges

**Protocols and Architecture:** Protocols, OSI, TCP/IP Protocol Suite

**Examples of networks:** Novell Netware, Arpanet, and Internet. Examples of Data Communication Services: X.25 Networks, Frame relay, Broad band ISDN and ATM. Physical Layer: Transmission media- Narrow band ISDN: Services-Architecture- Interface, Broad band ISDN and ATM- Virtual Circuits versus Circuit Switching – Transmission in ATM networks. FDDI

**Link Layer and Local Area Networks Data link layer:** Service provided by data link layer-Error detection and correction Techniques-Elementary data link layer protocols -Sliding Window protocols - Data link layer in HDLC,



Internet and ATM . Multiple Access protocols: Channel partitioning protocols: TDM-FDM-Code Division Multiple Access(CDMA) .Random Access protocols : ALOHACDMA and CSMA/CD . Local area Network: LAN addresses-Address Resolution Protocol-Reverse Address Resolution Protocol. Ethernet: Ethernet Technologies-IEEE standards- Hubs-Bridges and Switches

#### RECOMMENDED BOOKS:

1. Andrew S. Tanenbaum, "Computer Networks", Ed Pearson Education 4<sup>th</sup> Ed., 2003.
2. James F. Kurose and Keith W. Ross, "Computer Networking", Pearson Education, 2002.
3. William Stallings, "Data and Computer Communication", Pearson Education, 7<sup>th</sup> Ed., 2<sup>nd</sup> Indian Reprint 2004.
4. Miller, "Data and Network Communication", Ed Thomson Learning, 2001.
5. Douglas E Comer, "Computer Networks and Internets", Pearson Education 2<sup>nd</sup> Ed., 5<sup>th</sup> Indian Reprint 2001.

### ITX 207 INTRODUCTION TO DIGITAL LOGIC DESIGN

3 credits [3-0-0]

**Number Systems And Boolean Algebra:** Review of Number systems, Radix conversion, Complements 9's & 10's, Subtraction using 1's & 2's complements, Binary codes, Error detecting and Correcting codes, Theorems of Boolean algebra, Canonical forms, Logic gates.

**Digital Logic Families:** Introduction to bipolar Logic families: RTL, DCTL, DTL, TTL, ECL and MOS Logic families: NMOS, PMOS, CMOS, Details of TTL logic family - Totem pole, open collector outputs, TTL subfamilies, Comparison of different logic families.

**Combinational Logic:** Representation of logic functions, Simplification using Karnaugh map, Tabulation method, Implementation of combinational logic using standard logic gates, Multiplexers and Demultiplexers, Encoders and Decoders, Code Converters, Adders, Subtractors, Parity Checker and Magnitude Comparator.

**Sequential Logic Concepts And Components:** Flip flops - SR, JK, D and T flip flops - Level triggering and edge triggering, Excitation tables - Counters - Asynchronous and synchronous type Modulo counters, design with state equation state diagram, Shift registers, type of registers, circuit diagrams.

**D/A And A/D Converters:** Weighted resistor type D/A Converter, Binary ladder D/A converter, Steady state accuracy test, D/A accuracy and resolution, Parallel A/D Converter, counter type A/D converter, Successive approximation A/D converter, Single and Dual slope A/D converter, A/D accuracy and resolution.

**Semiconductor Memories:** Memory organization, Classification, and characteristics of memories, Sequential memories, ROMs, R/W memories, Content Addressable memories, Charged-Coupled Device memory, PLA, PAL and Gate Array.

#### RECOMMENDED BOOKS:

1. M.Morris Mano, Pearson Education, "DIGITAL DESIGN" 3<sup>rd</sup> Ed. ,PHI, 2007.
2. Donald D.Givone, "Digital Principles and Design", Tata McGraw Hill Ed., 2002.
3. John F Wakerly, "Digital Design Principles and Practices 3<sup>rd</sup> Ed.", Pearson Education 2001.
4. J P. Hayes, "Introduction to Digital Logic Design", Addison-Wesley Publishing, 2002.
5. Charles H. Roth Jr., "Fundamentals of logic design", 6<sup>th</sup> Ed., Cengage Learning, New Delhi, 2012.



Linear dependence of vectors and rank of matrices, linear transformations and inverse of matrices, reduction to normal form, bilinear form and quadratic form, consistency and solution of linear algebraic system of equations, eigen values, eigen vectors and their applications to system of ordinary differential equations, Cayley Hamilton theorem, orthogonal, unitary, hermitian and similar matrices.

Differential calculus of functions of several variables, partial differentiation, homogeneous functions and Euler's theorem, Taylor's and Maclaurin's series, Taylor's theorem for functions of two variables, functions of several variables, Lagrange's method of multipliers.

Double and triple integrals, change of order of integration, change of variables, applications to evaluation of area, surface area and volume.

Scalar, and vector fields, differentiation of vectors, velocity and acceleration, vector differential operators Del, Gradient, Divergence and Curl and their physical interpretations, formulae involving these operators, line, surface and volume integrals, solenoidal and irrotational vectors, Green's theorem, Gauss divergence theorem, Stoke's theorem and their applications.

Formulation and classification of partial differential equations, solution of first order linear equations, standard forms of non-linear equations, Charpit's method, linear equations with constant coefficients, non-homogenous linear equations, Monge's method for non-homogenous equations of second order, separation of variables method for solution of heat, wave and Laplace equation.

#### RECOMMENDED BOOKS:

1. B.V.Ramana, "Higher Engineering Mathematics", Tata Mc Graw-Hill Publishing Company Ltd., 2008.
2. R.K.Jain & S.R.K.Iyenger, "Advance Engineering Mathematics", Narosa Publishing House, 2002.
3. B.S.Grewal, "Engineering Mathematics", Khanna Publishers, 2004.
4. E.Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons, 2005.
5. G.F.Simmons, "Differential Equations", Tata Mc Graw-Hill Publishing Company Ltd. 1981.

**Definition and Scope of Engineering Economics:** Concept of revenue and costs, break-even analysis. Law of demand & supply, time value of money, present and future worth methods.

**Decision Making:** Decision making process, decision making under risk certainty, uncertainty and conflict.

**Replacement and maintenance Analysis:** Types of maintenance, determination of economic life of an asset, replacement of items that fail suddenly and that fail over a period of time.

**Methods of depreciation:** straight line method, sum-of-the year's digest method, declining balance method, sinking fund method and service output method of depreciation.

**Inventory control:** Introduction and objective of inventory control, purchase model with instantaneous replenishment, model with shortages, price break model, ABC analysis.

**Forecasting:** Demand forecasting by quantitative and qualitative techniques, applications of demand forecasting.

**Make or Buy Decision:** Criteria for make or buy, approaches for make or buy decision.

**Value Engineering Analysis:** Value analysis vs. value engineering function, aims and value engineering procedure, advantages & applications.

**Linear Programming:** Linear programming as a tool of decision making, graphical and Simplex Methods and applications in decision making.



## RECOMMENDED BOOKS:

1. Panaeerselvam, R., 'Engineering Economics', Prentice Hall of India: New Delhi, 2001.
2. Grant, E.L., Irevan, W.G. and Leanenworh, R.S., 'Principles of Engineering Economy'. Ronald Press: New York, 1976.
3. Jaha, H.A, 'Operations Research: An Introduction', Prentice-Hall of India: New Delhi, 2005.
4. Vohra , N.D., 'Quantitative Techniques in Managerial Decision Making'. Tata McGraw Hill: New Delhi, 2006.
5. Dougherty, Christopher, 'Introduction to Econometrics', Oxford University Press: New Delhi, 2007.

**ITX 221**

**OBJECT ORIENTED PROGRAMMING CONCEPTS LAB**

**1 credits [0-0-2]**

### Objectives:

- To make the student learn a object oriented way of solving problems.
- To teach the student to write programs in C++ to solve the problems.

### LIST OF PRACTICALS

1. Write a program to read a matrix of size m x n form the keyboard and display the same using function.
2. Program to make the use of inline function.
3. Write a function power () which raise a number m to a power n. The function takes double value of m and integer value of n and returns the result. Use a default value of n is 2 to make the function to calculate squares when this argument is omitted.
4. Program to show that the effect of default arguments can be alternatively achieved by overloading.
  - a) Write a class ACCOUNT that represents your bank account and then use it.
  - b) The class should allow you to deposit money, withdraw money, calculate interest,
  - c) send you a message if you have insufficient balance.
5. Write a class STRING that can be used to store strings, add strings, equate string, output strings.
6. Create the class TIME to store time in hours and minutes. Write a friend function to add two TIME objects.
7. Create two classes DM and DB. DM stores the distance in meter and centimeters and DB stores the distance in feet and inches. Write a program two add object of DM with the object of DB class.
8. Write a program to create an abstract class named Shape containing an empty method named number\_of\_Sides. Provide three classes named Trapezoid, Triangle and Hexagon such that each one of the classes inherits the class Shape. Each one of the classes contains only the method number Of Sides ( ) that shows the number of sides in the given geometrical figures.
9. Program to demonstrate the concept of:
  - a) Default constructor
  - b) Parameterized constructor
  - c) Copy constructor
  - d) Constructor overloading
10. Program to demonstrate the concept of destructor.
11. Program to show multiple inheritance
12. Program to show multilevel inheritance
13. Program to show hybrid inheritance
14. Program to show the concept of containership.
15. Program to overload unary operator.
16. Program to overload binary operator
17. Program to show the concept of run time polymorphism using virtual function.
18. Program to work with formatted and unformatted IO operations.
19. Program to read the name and roll numbers of students from keyboard and write them into a file and then display it.
20. Program to copy one file onto the end of another, adding line numbers
21. Write a function template for finding the minimum value contained in an array.



22. Write a class template to represent generic vector (a series of float values). Include member function to perform following tasks.
- Create vector
  - Modify the value of a given element
  - To multiply by a scalar value
  - To display vector in the form of (10, 20, 30,.....)

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*

**ITX 223                      DATA STRUCTURE LAB**

**2 credits [0-0-4]**

**Objectives:**

- To understand how the choice of data structures can lead to efficient implementations of algorithms.
- To familiarize how certain applications can benefit from the choice of data structures.

**LIST OF PRACTICALS**

Write Program in C or C++ for following.

- Program to demonstrate the use of Linear search to search a given element in an array.
- Program to demonstrate the use of Binary search to search a given element in an array.
- Write a program that implement following operations on a linear array:
  - Insert a new element at end as well as at a given position
  - Delete an element from a given whose value is given or whose position is given
- Write a program that implement following operations on a Linked List:
  - Insert a new element
  - Delete an existing element
  - Search an element
- Program to demonstration the implementation of various operations on a Stack represented using a linear array.
- Program to demonstration the implementation of various operations on a Queue represented using a linear array.
- Program to demonstration the implementation of various operations on a Circular queue represented using a linear array.
- Program to illustrate the implementation of different operations on a Binary Tree.
- Program to illustrate the traversal of graph using breadth-first search.
- Program to illustrate the traversal of graph using depth-first search.
- Program to sort an array of integers using bubble sort.
- Program to sort an array of integers using selection sort.
- Program to sort an array of integers using Insertion sort.
- Program to sort an array of integers using radix sort.
- Program to sort an array of integers using merge sort.
- Program to sort an array of integers using quick sort.

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*



**Objectives:**

- To understand the working of the fundamental elements of communications systems hardware.
- Study of different modulation techniques.

**LIST OF PRACTICALS**

1. Making Straight, Rollover and Cross-Over cables
2. Cable & RJ-45 Jack outlet installation
3. Installation of NIC Card & using TCP/IP
4. Design, build & test a simple communication system
5. Overview and basic Configuration of Router
6. Router show Command
7. Basic LAN Setup
8. Designing & Implementing LAN using subnetting
9. Study of Amplitude Modulation
10. Study of frequency Modulation
11. Study of ASK Modulation
12. Study of FSK Modulation
13. Simple point-to-point communication & error detection
14. Implementation of STOP and Wait protocol
15. Implementation of Sliding Window protocol

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents*

**Objectives:**

- To make the students experimentally familiar with the operation of various digital circuits and logic families.

**LIST OF PRACTICALS**

1. Verification of the truth tables of TTL gates, e.g., 7400, 7402, 7404, 7408, 7432, 7486.
2. Verify the NAND and NOR gates as universal logic gates.
3. Verification of the truth table of the Multiplexer 74150.
4. Verification of the truth table of the De-Multiplexer 74154.
5. Design and verification of the truth tables of Half and Full adder circuits.
6. Design and verification of the truth tables of Half and Full subtractor circuits.
7. Design and test of an S-R flip-flop using NOR/NAND gates.
8. Verify the truth table of a J-K flip-flop (7476)
9. Verify the truth table of a D flip-flop (7474)
10. Operate the counters 7490, 7493. Verify the frequency division at each stage and with a low frequency clock (say 1 Hz) display the count on LEDs.
11. Operate the universal shift register 74194.
12. Verify the truth table of decoder driver 7447/7448. Hence operate a 7 segment LED display through a counter using a low frequency clock.
13. Design and test D/A converter using R-2R Ladder Network

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*





## FOURTH SEMESTER

### ITX 202    COMPUTER SYSTEM ARCHITECTURE

3 credits [3-0-0]

**Introduction:** Number representation; fixed and floating point number representation, IEEE standard for floating point representation. Error detection and correction codes: Hamming code.

Digital computer generation, computer types and classifications, functional units and their interconnections, buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer.

**Central Processing Unit:** Addition and subtraction of signed numbers, look ahead carry adders. Multiplication: Signed operand multiplication, Booths algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation, Processor organization, general register organization, stack organization and addressing modes.

**Control Unit:** Instruction types, formats, instruction cycles and subcycles ( fetch and execute etc) , micro-operations, execution of a complete instruction.

Hardwire and micro programmed control: microprogramme sequencing, wide branch addressing, microinstruction with next address field, pre-fetching microinstructions, concept of horizontal and vertical microprogramming.

**Memory:** Basic concept and hierarchy, semiconductor RAM memories, 2D & 2 1/2D memory organization. ROM memories.

Cache memories: concept and design issues (performance, address mapping and replacement)

Auxiliary memories: magnetic disk, magnetic tape and optical disks

Virtual memory: concept implementation.

**Input / Output:** Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions.

Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors.

Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.

#### RECOMMENDED BOOKS:

1. M Moris Mano, "Computer System Architecture", Pearson Education, 3<sup>rd</sup> Ed. 1993.
2. David A. Patterson and John L. Hennessy, "Computer Organization & Design-The Hardware/Software Interface", Morgan Kaufmann, 2<sup>nd</sup> Ed. 1997.
3. William Stallings, "Computer Organisation and Architecture, Designing for Performance", Pearson Education Asia, 6<sup>th</sup> Ed. 2003.
4. Harry F. Jordan and Gita Alaghband, "Fundamentals of Parallel Processing", Pearson Education, 1<sup>st</sup> Ed. 2003.
5. Barry Wilkinson Michael Allen, "Parallel Programming", Prentice Hall, 1999.

### ITX 204    INTRODUCTION TO DESIGN AND ANALYSIS OF ALGORITHMS

3 credits [3-0-0]

**Introduction :** Algorithms, Analyzing algorithms, Complexity of algorithms, Growth of functions, Performance measurements, Sorting and order Statistics - Shell Sort, Quick Sort, Merge Sort, Heap Sort, Comparison of sorting algorithms, Sorting in linear time.

**Advanced Data Structures:** Red-Black trees, B – trees, Binomial Heaps, Fibonacci Heaps.

**Divide and Conquer:** Divide and Conquer with examples such as Sorting, Matrix Multiplication, Convex hull and Searching.

**Greedy methods:** Greedy methods with examples such as Optimal Reliability Allocation, Knapsack, Minimum Spanning trees – Prim's and Kruskal's algorithms, Single source shortest paths - Dijkstra's and Bellman Ford algorithms.



**Dynamic programming:** Dynamic programming with examples such as Kanpsack, All pair shortest paths – Warshal’s and Floyd’s algorithms, Resource allocation problem. Backtracking, Branch and Bound with examples such as Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of subsets.

**Selected Topics:** Algebraic Computation, Fast Fourier Transform, String Matching, Theory of NP-completeness, Approximation algorithms and Randomized algorithms.

#### RECOMMENDED BOOKS:

1. Cormen, leiserson and Rivest , “Introduction to Algorithms”, McGraw-Hill, 2001.
2. Sartaj Sahni and Ellis Horwitz, “Design and Analysis of Algorithms”, Galgotia Publications, 1999.
3. Ullman and Hopcroft, “Design and Analysis of Algorithms”, Pearson Education, 2006.
4. J. Kleinberg and E. Tardos, “Algorithm Design”, Addison-Wesley, 2005.
5. Michael T. Goodrich and Roberto Tamassia, “Algorithm Design: Foundations, Analysis, and Internet Examples”, Wiley, 2002.

### ITX 206 FUNDAMENTALS OF DATABASE MANAGEMENT SYSTEMS

3 credits [3-0-0]

**Introduction:** An overview of database management system, database system Vs file system, Database system concepts and architecture, data models schema and instances, data independence and data base language and interfaces, Data definitions language, DML, Overall Database Structure.

**Data Modeling using the Entity Relationship Model:** ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationships of higher degree.

**Relational data Model and Language:** Relational data model concepts, integrity constraints: entity integrity, referential integrity, Keys constraints, Domain constraints, relational algebra, relational calculus, tuple and domain calculus.

**Introduction to SQL:** Characteristics of SQL. Advantage of SQL. SQL data types and literals. Types of SQL commands. SQL operators and their procedure. Tables, views and indexes. Queries and sub queries. Aggregate functions. Insert, update and delete operations. Joins, Unions, Intersection, Minus, Cursors in SQL.

**Data Base Design & Normalization:** Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependences, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design.

**Transaction Processing Concepts:** Transaction system, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling.

**Crash Recovery:** Failure classification, recovery concepts based on deferred update, recovery concepts based on intermediate update, shadow paging, check points, on-line backup during database updates

**Concurrency Control Techniques:** Concurrency control, locking Techniques for concurrency control, Time stamping protocols for concurrency control, validation based protocol, multiple granularity, Multi version schemes, Recovery with concurrent transaction.

**Client/Server Databases:** Client/Server concepts, approach, Client/Server environments, characterization of Client/Server computing, application partitioning, the two-layer, and the Three layer architecture, Client/Server communication, APIs in Client/Server computing, middleware technology, application developments, design concepts, Client application development tools, and database servers.

**Integrity, Security and Repositories:** Needs for database integrity, integrity constraints, non-procedural integrity constraints, integrity constraints specifications in SQL, introduction to database security mechanism, security specification in SQL, system catalogues



### Case Studies:

**Oracle:** Database Design and Querying Tools; SQL Variations and Extensions; Storage and Indexing; Query Processing and Optimization; Concurrency Control and Recovery; System Architecture; Replication, Distribution and External Data; Database Administration Tools.

**IBM DB2:** Universal database; Database Design and Querying Tools; SQL Variations and Extensions Storage and Indexing; Query Processing and Optimization; Concurrency Control and Recovery; System Architecture; Replication, Distribution and External Data; Database Administration Tools.

### RECOMMENDED BOOKS:

1. Date C J, “An Introduction To Database System”, 8<sup>th</sup> Ed., Addison Wesley, 2003.
2. Korth, Silbertz, Sudarshan, “Database Concepts”, 6<sup>th</sup> Ed., McGraw Hill, 2010.
3. Elmasri, Navathe, “Fundamentals Of Database Systems”, 5<sup>th</sup> Ed., Addison Wesley, 2007.
4. Bipin C. Desai, “An introduction to Database Systems”, 8<sup>th</sup> Ed., Galgotia Publication, 2006.
5. Rob and Coronel, “Database Systems” 5<sup>th</sup> Ed., Cengage Learning, 2008.

## ITX 208 PROGRAMMING IN JAVA

3 credits [3-0-0]

**Overview of Basic OOP Concepts:** Need for object-oriented paradigm: Agents, responsibility, messages, methods, classes and instances, class hierarchies (Inheritance), method binding, datatypes, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, classes and objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling, inheritance, super keyword, polymorphism- method overriding, abstract classes.

**Packages and Interfaces:** Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring packages – Java.io, Java.util.

**Exception handling and multithreading:** Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

**Event Handling:** Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, lists panels – scroll pane, dialogs, menubar, graphics, layout manager – layout manager types – boarder, grid, flow, card and grid bag.

**Applets:** Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.

**Swing:** Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing-JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

**Networking:** Basics of network programming, addresses, ports, sockets, simple client server program, multiple clients, Java .net package Packages – java.util,

### RECOMMENDED BOOKS:

1. J.Nino and F.A. Hosch, “An Introduction to programming and OO design using Java”, 2<sup>nd</sup> Ed., John Wiley & sons, 2005.
2. T. Budd, “An Introduction to OOP”, 3<sup>rd</sup> Ed., Addison-Wesley, 2002.
3. Y. Daniel Liang, “Introduction to Java programming”, 6<sup>th</sup> Ed., Prentice Hall, 2006.
4. R.A. Johnson-Thomson , “An introduction to Java programming and object oriented application development”, 1<sup>st</sup> Ed., Cengage Learning, 2006.
5. Cay.S.Horstmann and Gary Cornell, “Core Java 2, Vol 1, Fundamentals”, 5<sup>th</sup> Ed., Prentice Hall, 2000.



**Introduction and Line Generation:** Types of computer graphics, Graphic Displays- Random scan displays, Raster scan displays, Frame buffer and video controller, Points and lines, Line drawing algorithms, Circle generating algorithms, Mid point circle generating algorithm, and parallel version of these algorithms.

**Transformations:** Basic transformation, Matrix representations and homogenous coordinates, Composite transformations, Reflections and shearing. Windowing and Clipping: Viewing pipeline, Viewing transformations, 2-D Clipping algorithms- Line clipping algorithms such as Cohen Sutherland line clipping algorithm, Liang Barsky algorithm, Line clipping against non rectangular clip windows; Polygon clipping – Sutherland Hodgeman polygon clipping, Weiler and Atherton polygon clipping, Curve clipping, Text clipping.

**Three Dimensional:** 3-D geometric primitives, 3-D Object representation, 3-D Transformation, 3-D viewing, projections, 3-D Clipping.

**Curves and Surfaces:** Quadric surfaces, Spheres, Ellipsoid, Blobby objects, Introductory concepts of Spline, B-spline and Bezier curves and surfaces. Hidden Lines and Surfaces: Back Face Detection algorithm, Depth buffer method, A- buffer method, Scan line method, basic illumination models – Ambient light, Diffuse reflection, Specular reflection and Phong model, Combined approach, Warn model, Intensity Attenuation, Color consideration, Transparency and Shadows.

#### BOOKS RECOMMENDED

1. Donald Hearn and M Pauline Baker, “Computer Graphics C Version”, Pearson Education
2. Amrendra N Sinha and Arun D Udai,” Computer Graphics”, TMH 16
3. Donald Hearn and M Pauline Baker, “Computer Graphics with OpenGL”, Pearson education
4. Steven Harrington, “Computer Graphics: A Programming Approach” , TMH
5. Rogers, “ Procedural Elements of Computer Graphics”, McGraw Hill

Introduction to Financial Accounting, Accounting as an information system, Basic accounting concepts and conventions. Transaction analysis, journal posting, Corporate financial statements (Trading Account, Profit and Loss Account and Balance Sheet). Cost types and its management, cost behaviour, products costing systems, valuation of materials and labour cost computation, job, batch, and contract costing, standard costing, variance analysis activity based costing, allocation of overheads, joint product and by-product costing, cost-volume-profit analysis, marginal costing, cost control systems, budgetary analysis.

#### RECOMMENDED BOOKS:

1. Sharma, Subhash and M.P. Vithal, "Financial Accounting for Management Text and Cases", McMillan, 1989.
2. Jawahar Lal and Seema Srivastava, “Cost Accounting”, Tata McGraw Hill, 2008.
3. M.Y. Khan & P.K. Jain, Cost Accounting, Tata McGraw Hill, 2003.
4. Robert and Anthony, “Management Accounting”, Pearsons Education India, 2007.
5. Gupta, R.L., and Radha Swamy, “Advanced Accounting”, Sultan Chand & Sons, 1990.



**Objectives:**

- To introduce mathematical aspects and analysis of algorithms.
- To introduce algorithm design methods

**LIST OF PRACTICALS**

1. Program to implement Quick sort using the Divide and Conquer technique and analyze its Time Complexity.
2. Program to implement Merge sort using the Divide and Conquer technique and analyze its Time Complexity.
3. Program to perform Binary Search using the Divide and Conquer technique and analyze its Time Complexity.
4. Program to implement Strassen's Matrix Multiplication Algorithm and analyze its Time Complexity.
5. Program to find the Minimum Spanning tree using prim's algorithm.
6. Program to find the Minimum Spanning tree using kruskal's algorithm.
7. Program to solve the knapsack problem using greedy method.
8. Program to find the shortest path of the multistage graph using dynamic programming.
9. Program to solve the Traveling salesman problem using the dynamic programming approach.
10. Program to solve the Optimal Binary Search Tree problem using the dynamic programming approach.
11. Program to find the solution to the N queen's problem using backtracking.
12. Program to find the shortest path using Floyd's algorithm.
13. Program to solve Graph Coloring problem.
14. Program to solve Hamiltonian Cycle Problem.
15. Program to implement Knuth Morris Pratt algorithm and analyze its time complexity.

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*

**Objectives:**

- To provide a strong formal foundation in database concepts, technology and practice to the participants to groom them into well-informed database application developers.

**LIST OF PRACTICALS**

1. Write the queries for Data Definition and Data Manipulation language.
2. Write SQL queries using Logical operators (=,<,>,etc.).
3. Write SQL queries using SQL operators (Between.... AND, IN(List), Like, IS NULL and also with negating expressions ).
4. Write SQL query using character, number, date and group functions.
5. Write SQL queries for Relational Algebra (UNION, INTERSECT, and MINUS, etc.).
6. Write SQL queries for extracting data from more than one table (Equi-Join, Non-Equi-Join , Outer Join)
7. Write SQL queries for sub queries, nested queries.
8. Write programs by the use of PL/SQL.
9. Concepts for ROLL BACK, COMMIT & CHECK POINTS.
10. Create VIEWS, CURSORS, and TRIGGERS & write ASSERTIONS.
11. Create FORMS and REPORTS.
12. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.



13. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
14. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
15. Queries using Conversion functions (to\_char, to\_number and to\_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next\_day, add\_months, last\_day, months\_between, least, greatest, trunc, round, to\_char, to\_date)
16.
  - a) Creation of simple PL/SQL program which includes declaration section, executable section and exception Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
  - b) Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
17. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
18. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
19. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
20. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
21. Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.
22. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
23. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

\* Students are advised to use **Developer 2000/Oracle-10i** or higher version or other latest version for above listed experiments. However depending upon the availability of software's, students may use **Power Builder /SQL SERVER**. Mini Project may also be planned & carried out through out the semester to understand the important various concepts of Database.

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*

## ITX 226 PROGRAMMING IN JAVA LAB

2 credits[0-0-4]

### Objectives:

- To make the student learn the application of advanced object oriented concepts for solving problems.
- To teach the student to write programs using advanced Java features to solve the problems

### LIST OF PRACTICALS

1. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.
2. Write a Java program to multiply two given matrices.
3. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
4. Write a Java program for sorting a given list of names in ascending order.
5. Write a java program to create an abstract class named Shape that contains an empty method named number Of Sides ( ). Provide three classes named Trapezoid, Triangle and Hexagon such that each one of the classes extends the class Shape. Each one of the classes contains only the method number Of Sides ( ) that shows the number of sides in the given geometrical figures.
6. Develop an applet that displays a simple message.
7. Develop an applet that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked.
8. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits



- and for the +, -, \*, % operations. Add a text field to display the result.
9. Write a Java program for handling mouse events.
  10. Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds
  11. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
  12. Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the textfields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.
  13. Write a Java program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. (Use java.net)
  14. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts.
  15. Write a Java program that allows the user to draw lines, rectangles and ovals.
  16. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using JTable component.

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*

## **ITX 228    COMPUTER GRAPHICS LAB**

**1 credits[0-0-2]**

### **Objectives:**

- To learn algorithmic development of graphics primitives like: line, circle, ellipse, polygon etc.
- To learn the representation and transformation of graphical images and pictures

### **LIST OF PRACTICALS**

1. To draw a line using DDA Algorithm.
2. To draw a line using Bresenham’s Algorithm.
3. To draw a circle using trigonometric Algorithm.
4. To draw a circle using Bresenham’s Algorithm.
5. To draw a circle using Midpoint Algorithm.
6. To draw an ellipse using Trigonometric Algorithm.
7. To draw an ellipse using Midpoint Algorithm.
8. To translate an object with translation parameters in X and Y directions.
9. To scale an object with scaling factors along X and Y directions.
10. To rotate an object with a certain angle.
11. To perform composite transformations of an object.
12. To clip line segments against windows.
13. Demonstrate the properties of Bezier Curve.
14. Run a sample session on Microsoft Windows including the use of Paintbrush.

*This is only the suggested list of Practicals. Instructor may frame additional Practicals relevant to the course contents.*

