**B.TECH 3rd  SEMESTER CSE**

**Discipline : CSE**

**Semster : 3rd**

**Subject : Data Structures and Algorithms**

**Subject Code : CSE-201T**

**Lesson Plan Duration : 15 Weeks**

**Work Load(Per week) : 3 Hours/Week& 1 Practical**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Week | Theory | | Practical | |
|  | Lecture Day | Topic | Practical Day | Topic |
| 1 | 1 | Data Structures: Definition and its types, Abstract Data Types | 1 | Program to perform following operations on tables using functions only a) Addition b) Subtraction |
| 2 | Static and dynamic memory storage. |
| 3 | Arrays, matrices, sparse matrices |
| 2 | 4 | multi-dimensional arrays, operations on arrays | 2 | Programto perform following operations on tables using functions only a)Multiplication b) transpose |
| 5 | Linear search, Binary search, |
| 6 | Insertion sort, selection sort |
| 3 | 7 | Bubble sort, Merge sort | 3 | Program to find the element in an array using linear Search Method. |
| 8 | Linked Lists: List Types (singly, doubly) |
| 9 | Linked Lists: (singly circular, header, doubly circular) |
| 4 | 10 | Operations on Lists – create, insert | 4 | Program to find the element in an array using Binary Search Method. |
| 11 | Operations on Lists -delete, search, display; Applications of linked list |
| 12 | TEST of UNIT-I |
| 5 | 13 | Stacks: Definition, Array implementation of stacks | 5 | Program to implement the insertion operations on Singly linked list |
| 14 | Linked implementation of stacks |
| 15 | Applications of Stacks: Infix, Postfix and prefix expression |
| 6 | 16 | conversions and evaluation of expressions | 6 | Program to implement the deletion operations on Singly linked list |
| 17 | Recursion, Quick Sort |
| 18 | Queues: Definition, Array implementation of queues |
| 7 | 19 | Linked implementation of queues | 7 | Program to implement an integer stack using arrays |
| 20 | Circular queues |
| 21 | Priority queues, Double-ended queues |
| 8 | 22 | Assignment-1 | 8 | Program to implement an integer stack using singly linked list. |
| 23 | Trees: Binary Trees and their properties |
| 24 | Static representation of binary trees |
| 9 | 25 | Linked representation of binary trees | 9 | Program to implement an integer queue using arrays |
| 26 | Complete Binary Tree, Threaded Binary tree |
| 27 | Different tree traversal algorithms(non-recursive) |
| 10 | 28 | Binary Search Tree (create, delete) | 10 | Program to implement an integer circular queue using singly linked list |
| 29 | Binary Search Tree ( search, insert, display) |
| 30 | Heap Sort and its complexity analysis |
| 11 | 31 | AVL Trees, Balanced multi-way search trees. | 11 | Program to implement Binary tree. |
| 32 | TEST of UNIT-II |
| 33 | Graphs: Definition, Array representation of graphs |
| 12 | 34 | Linked representation of graphs | 12 | Program to implement traversal pre-order, in-order and post-order. |
| 35 | Graph Traversal (BFS and DFS) |
| 36 | Adjacency matrix and adjacency lists |
| 13 | 37 | path matrix | 13 | Program to implement the following sorting techniques a) Bubble sort b) Quick sort |
| 38 | Finding Shortest Path - Warshall's Algorithm |
| 39 | Hashing, Hash table |
| 14 | 40 | Hash functions | 14 | Program to implement the following sorting techniquesc) Selection sort d) Merge sort. |
| 41 | Running time: Time Complexity, Big – Oh - notation |
| 42 | Best Case, Worst Case, Average Case |
| 15 | 43 | Factors depends on running time, Evaluating time Complexity | 15 | Internal Viva |
| 44 | Assignment-2 |
| 45 | Problems |

**Discipline : CSE**

**Semster : 3rd**

**Subject : Object Oriented Programming Using C++**

**Subject Code : CSE-202T**

**Lesson Plan Duration : 15 Weeks**

**Work Load(Per week) : 3 Hours/Week**

|  |  |  |
| --- | --- | --- |
| Week | Theory | |
|  | Lecture Day | Topic |
| 1 | 1 | Object Oriented Programming, C++ Standard Library, Basics of a Typical C++ Environment, Illustrative Simple C++ Programs |
| 2 | Pre-processors Directives, Macro Programs, Header Files and Namespaces |
| 3 | library files, new features of ANSI C++ standard, Introduction to VC++, Dev C++ etc. |
| 2 | 4 | Encapsulation (Information Hiding), Access Modifiers: Controlling access to a class method, or variable (public, protected, private, block level and scope), Other Modifiers. |
| 5 | Block structure of Class and Struct in memory, Accessing Members of Structures |
| 6 | Class Scope and Accessing Class Members |
| 3 | 7 | Separating Interface from Implementation |
| 8 | Default Constructors, Chained Constructor, Default Arguments with Constructors |
| 9 | Constant Object and Const Member Functions |
| 4 | 10 | Object as Member of Class, Using Destructors. |
| 11 | **Function:** Controlling Access Function and Utility Functions, |
| 12 | Function overloading |
| 5 | 13 | Introduction, Inline Function |
| 14 | Friend Function and Friend Classes, |
| 15 | Using This Pointer, Dynamic Memory Allocation with New and Delete, |
| 6 | 16 | Static Class Members |
| 17 | Container And Iterators, algorithm andfunctional Classes, Proxy Class. |
| 18 | Overloading, Inheritance, Overriding Methods, Abstract Classes |
| 7 | 19 | Reusability, Class’s Behaviors. |
| 20 | **Inheritance:** Base Classes And Derived Classes, Protected Members, |
| 21 | Casting Base- Class Pointers to Derived- Class Pointers, Using Member Functions |
| 8 | 22 | Overriding Base –Class Members in a Derived Class, Public, Protected and Private Inheritance, |
| 23 | Using Constructors and Destructors in derived Classes, Implicit Derived –Class Object to Base- Class Object Conversion, Composition Vs. Inheritance. |
| 24 | **Virtual Functions and Polymorphism:** Introduction to Virtual Functions |
| 9 | 25 | Abstract Base Classes and Concrete Classes, |
| 26 | Polymorphism, New Classes and Dynamic Binding |
| 27 | Virtual Destructors, Polymorphism, Dynamic Binding. |
| 10 | 28 | Files and Streams |
| 29 | Creating a Sequential Access File, Reading Data From a Sequential Access File, Updating Sequential Access Files |
| 30 | Random Access Files, Creating a Random Access File, Writing Data Randomly to a Random Access File, |
| 11 | 31 | Reading Data Sequentially from a Random Access File |
| 32 | **Managing Console I/O:** Stream Input/output Classes and Objects, Stream Output, Stream Input |
| 33 | Unformatted I/O (with read and write), Stream Manipulators |
| 12 | 34 | Stream Format States, Stream Error States. |
| 35 | Introduction, Basics of C++ Exception Handling |
| 36 | Try Throw, Catch, Throwing an Exception, Catching an Exception, |
| 13 | 37 | Rethrowing an Exception, Exception specifications, Processing Unexpected Exceptions, |
| 38 | Stack Unwinding |
| 39 | Constructors, Destructors and Exception Handling |
| 14 | 40 | Exceptions and Inheritance. |
| 41 | Introduction, Function Templates, Overloading Template Functions |
| 42 | Class Template, Class Templates and Non-Type Parameters |
| 15 | 43 | Templates and Inheritance, |
| 44 | Templates and Friends, |
| 45 | Templates and Static Members. |

**Discipline : CSE**

**Semster : 3rd**

**Subject : Discrete Structures**

**Subject Code : CSE-203T**

**Lesson Plan Duration : 15 Weeks**

**Work Load(Per week) : 3 Hours/Week**

|  |  |  |
| --- | --- | --- |
| Week | Theory | |
|  | Lecture Day | Topic |
| 1 | 1 | Introduction to set theory. |
| 2 | Set operations, Algebra of sets |
| 3 | Duality, Finite and Infinite sets |
| 2 | 4 | Classes of sets, Power Sets |
| 5 | Multi sets, Cartesian Product |
| 6 | Representation of relations, Types of relation |
| 3 | 7 | Equivalence relations and partitions, Partial ordering relations and lattices |
| 8 | Function and its types, Composition of function and relations |
| 9 | Cardinality and inverse relations |
| 4 | 10 | TEST of UNIT-1 |
| 11 | Basic operations: AND(^), OR(v), NOT(~), |
| 12 | Truth value of a compound statement |
| 5 | 13 | propositions, tautologies |
| 14 | contradictions |
| 15 | **Assignment-1** |
| 6 | 16 | Permutations with and repetition |
| 17 | Permutations without repetition |
| 18 | Combination |
| 7 | 19 | Polynomials and their evaluation, Sequences |
| 20 | Introduction to AP, GP and AG series |
| 21 | partial fractions |
| 8 | 22 | linear recurrence relation with constant coefficients |
| 23 | Homogeneous solutions |
| 24 | Particular solutions |
| 9 | 25 | Total solution of a recurrence relation using generating functions |
| 26 | **TEST Of UNIT-2** |
| 27 | Definition and examples of a monoid |
| 10 | 28 | Semigroup, Groups and rings |
| 29 | Homomorphism, Isomorphism and Automorphism |
| 30 | Subgroups and Normal subgroups |
| 11 | 31 | Cyclic groups, Integral domain and fields |
| 32 | Cosets, Lagrange’s theorem |
| 33 | **Assignment-2** |
| 12 | 34 | Introduction to graphs, Directed and Undirected graphs |
| 35 | Homomorphic and Isomorphic graphs, Subgraphs |
| 36 | Cut points and Bridges, Multigraph and Weighted graph |
| 13 | 37 | Paths and circuits, Shortest path in weighted graphs |
| 38 | Eurelian path and circuits, Hamilton paths and circuits |
| 39 | Planar graphs, Euler’s formula, Trees, |
| 14 | 40 | Spanning trees, |
| 41 | Binary trees and its traversals |
| 42 | Problems |
| 15 | 43 | TEST Of UNIT-3 |
| 44 | Problems |
| 45 | TEST Of UNIT-4 |

**Discipline : CSE**

**Semster : 3rd**

**Subject : Computer Organization and Architecture**

**Subject Code : CSE-204T**

**Lesson Plan Duration : 15 Weeks**

**Work Load(Per week) : 3 Hours/Week**

|  |  |  |
| --- | --- | --- |
| Week | Theory | |
|  | Lecture Day | Topic |
| 1 | 1 | Basic principles of Boolean Algebra & Logic Gates |
| 2 | Laws of Boolean Algebra |
| 3 | Minimization of Boolean Function |
| 2 | 4 | Minimization using k maps for two and three variable |
| 5 | Minimization using k maps for four variable |
| 6 | Adder: Half adder and Full adder |
| 3 | 7 | Subtractor |
| 8 | Multiplexers and Demultiplxers |
| 9 | Encoder and Priority Encoder |
| 4 | 10 | Decoder |
| 11 | Sequential Logic Blocks Flip Flops |
| 12 | Flip Flops |
| 5 | 13 | Counters |
| 14 | Registers |
| 15 | Flynn’s Classification |
| 6 | 16 | Performance Metrics |
| 17 | Types of CPU Architectures |
| 18 | Test of Unit-1 |
| 7 | 19 | Stored Program Concept |
| 20 | Timing and Control Unit |
| 21 | Instruction codes and types of instructions |
| 8 | 22 | Basic of Logic Design: Accumulator Logic |
| 23 | Control Memory |
| 24 | Micro-programmed Control :address sequencing |
| 9 | 25 | Micro-instruction format |
| 26 | Micro-Program sequencer |
| 27 | Implementation of control unit |
| 10 | 28 | Test of unit-2 |
| 29 | RISC & CISC Processors |
| 30 | Stack Organization |
| 11 | 31 | Instruction formats |
| 32 | Addressing modes |
| 33 | Operations in the instruction set |
| 12 | 34 | Introduction to parallelism and its goals |
| 35 | Instruction level Parallelism:Pipelining |
| 36 | Processor level Parallelism |
| 13 | 37 | Amdahl’s Law and types of interrupts |
| 38 | Test of unit-3 |
| 39 | **The need of memory hierarchy & Memory Parameters** |
| 14 | 40 | Main memory and its expansion and its types |
| 41 | Cache memory and its mapping :Direct and associative |
| 42 | Input-Output Interface |
| 15 | 43 | Modes of transfer |
| 44 | Direct memory access(DMA) |
| 45 | Test of unit-4 |

**Discipline**: Mathematics

**Semester**: 3rd  Semester

**Subject**: Mathematics-III (Theory)

**Lesson Plan Duration**: 12 weeks (From August 2019 to November, 2019)

**Word Load (Lecture) per week (in hours):** Lectures-04

|  |  |  |
| --- | --- | --- |
| **Week** | **Lecture** | **Syllabus** |
| 1st | 1 | Fourier series |
| 2 | Euler's formulae |
| 3 | Conditions for a Fourier expansion |
|  | 4 | Change of interval |
| 2nd | 5 | Problems |
| 6 | Fourier expansion of odd and even functions |
| 7 | Fourier expansion of square wave |
|  | 8 | Fourier expansion of rectangular wave |
| 3rd | 9 | Fourier expansion of saw toothed wave |
| 10 | Half rectfied wave |
| 11 | Full rectified wave |
|  | 12 | Half range sine series |
| 4th | 13 | Half range cosine series |
| 14 | Fourier integrals |
| 15 | Fourier transforms, shifting therom |
|  | 16 | Fourier transform of derivatives |
| 5th | 17 | Fourier transform of integrals |
| 18 | Numerical Problems |
| 19 | Doubts |
|  | 20 | Test |
| 6th | 21 | Convolution therom |
| 22 | Fourier transform of Dirac delta's function |
| 23 | Complex functions definition |
|  | 24 | Complex functions questions |
| 7th | 25 | Exponential functions and questions |
| 26 | Trigonometric functions and questions |
| 27 | Hyperbolic and logarithmic functions |
|  | 28 | Limit and continuity of function |
| 8th | 29 | Differentiability and analyticity |
| 30 | Problems |
| 31 | Test |
|  | 32 | Necessary and sufficient condition for a function to be analytic |
| 9th | 33 | Polar form of C-R equation |
| 34 | Harmonic functions |
|  | 35 | Complex integral |
|  | 36 | Cauchy gaursat therom, cauchy integral formula |
| 10th | 37 | Power series, radius and circle of convergence |
| 38 | Taylor's maclaurin's and Laurent's series |
| 39 | Revision |
|  | 40 | Test |
| 11th | 41 | Zeroes and singularity of complex functions |
| 42 | Residue |
| 43 | Evaluation of real integrals using residues |
|  | 44 | Revision |