|  |  |
| --- | --- |
| **Course-20 Title: Computer Programming Contest – I** |  |
| **Course No.:CCE-124 Credit : 0.00 Contact Hours: 1** | **Total Marks:** |

**11.1 Rationale:**

**11.2 Objectives:**

|  |  |
| --- | --- |
| **Course-21 Title: Data Structure and Algorithms** |  |
| **Course No.: CIT 211 Credit : 3 Contact Hours: 3** | **Total Marks: 100** |

**11.1 Rationale:**

To become a successful computer professional, one needs to have in depth knowledge of data structures to apply them in problem solving and algorithms to analyze different solutions to problems.

**11.2 Objectives:**

1. Apply different data structures to problem solving
2. Explain different algorithms of solutions to problems.
3. Analyze different problem solutions

|  |  |  |  |
| --- | --- | --- | --- |
| **11.3 Learning Outcomes** | **11.4 Course Content** | **11.5**  **Teaching Strategy/ Learning Experience** | **11.6 Assessment Strategy** |
| * Define basic terminologies of data structures | **Introduction:** Basic Terminology, Elementary Data Organization, Data Structures | **Lecture** | **Quiz**  **Short Answer** |
| * Analyze algorithms | Algorithms, and Complexity of Algorithms | **Lecture**  **Exercise** | **Exercise**  **Assignment** |
| * Define elementary data structures * Apply arrays * Calculate memory requirements | elementary data structures, arrays | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Apply stacks, queues and recursion * Differentiate between stacks and queues * Evaluate expressions using stacks * Analyze recursive functions | **Stacks, Queues and Recursion:** Fundamentals, Different types of stacks and queues: Circular, dequeues etc.; evaluation of expressions, multiple stacks and queues; Recursion: Direct and indirect recursion, depth of recursion; Simulation of Recursion: Removal of Recursion; Towers of Hanoi. | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Define and identify graphs and trees * Construct graphs and trees * Differentiate between graphs and trees | **Elements of Graphs and Trees:** Graph Terminology, Paths and Circuits, Connectedness, Matrix Representation of Graph and Isomorphism of graphs. Trees, Rooted trees, Path Lengths in Rooted Trees. | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Construct linked lists * Apply linked lists to stacks and queues * Analyze memory allocation | **Linked Lists:** Single linked lists, Linked stacks and queues, the storage pool, polynomial addition, equivalence relations, sparse matrices, doubly linked lists and dynamic storage management, generalized lists, garbage collection and compaction | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Identify extended binary trees * Construct and apply extended binary trees | **Extended binary trees:** 2-trees, internal and external path lengths, Huffman codes/algorithm; Threaded binary trees, binary tree representation of trees; Application of Trees: Set representation, decision trees, game trees; Counting binary trees | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Apply sorting algorithms * Distinguish among sorting algorithms | **Sorting:** Searching, bubble sort, shell sort, insertion sort, selection sort, quick sort, heap sort, 2-way merge sort, sorting on several keys, practical considerations of internal sorting. searching, hash techniques | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |
| * Apply algorithmic techniques * Analyze algorithms | **Algorithms:** Techniques for analysis of algorithms; Algorithmic Techniques: divide-and-conquer, greedy method, dynamic programming, backtracking, branch and bound; Flow algorithms. Topological sorting; Connected components; spanning trees; Shortest paths; Algebraic simplification and transformations; Lower bound theory; NP-completeness; NP-hard and NP-complete problems; Approximation Algorithms; Introduction to parallel algorithms | **Lecture**  **Exercise**  **Demonstration** | **Quiz**  **Assignment**  **Practical exam** |

**RECOMMENDED BOOKS AND PERIODICALS**

**Text Books**:

1. 1. S. Lipschutz : “ Theory and Problem of Data Stuctures”

2. E. Horowitz : “Data Structure”

3. D. E. Knuth :“The Art of Computer Programming, Vol. 1, Fundamental Algorithms”

4. D. E. Knuth :“The Art of Computer Programming, Vol. 2, Seminumerical Algorithms”

5. D. E. Knuth :“The Art of Computer Programming, Vol. 3, Sorting and Searching”

6. Goodman : “Introduction to Design and Analysis of Algorithms”

7. Robert Sedgewick :"Algorithms"

8. Ellis Horowitz & Sartaj Sahni : "Fundamentals of Computer Algorithms"