**Title of Subject : Data Structures & Algorithms (SW-212)**

**Discipline :** Software Engineering (3rd Semester)

**Effective :** 18 Batch & onwards

**Pre-requisite :** Programming Fundamentals

**Assessment :** Theory**:** 20% Sessional, 80% Written Semester Examination

## (20% Mid, 60% Final)

Practical: 40% Sessional, 60% Final Examination

**Credit Hours :** 03 + 01 **Marks :** 100 +50

**Minimum Contact Hours:** 45 + 45

# Specific Objectives of course:

* To gain knowledge of a variety of data structures.
* To learn how to efficiently use data structures when storing and retrieving data in a computer.
* To use learned data structures in different applications.

**COURSE LEARNING OUTCOMES:**

Upon successful completion of the course, the student will be able to:

|  |  |  |  |
| --- | --- | --- | --- |
| **CLOs** | **Description** | **Taxonomy level** | **PLO** |
| 1 | Understand various data structures such as arrays, lists, trees, graphs etc. | C2 | 1 |
| 2 | Solving algorithms associated with each data structure. | C3 | 2 |
| 3 | Demonstrate the implementation of algorithms programmatically. | P4 | 3 |

**PROGRAM LEARNING OUTCOMES (PLOs):**

The course is designed so that students will achieve the following PLOs:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Engineering Knowledge: | ☑ | 7 | Environment and Sustainability: | ☐ |
| 2 | Problem Analysis: | ☑ | 8 | Ethics: | ☐ |
| 3 | Design/Development of Solutions: | ☑ | 9 | Individual and Team Work: | ☐ |
| 4 | Investigation: | ☐ | 10 | Communication: | ☐ |
| 5 | Modern Tool Usage: | ☐ | 11 | Project Management: | ☐ |
| 6 | The Engineer and Society: | ☐ | 12 | Lifelong Learning: | ☐ |

**Course outline:**

Abstract Data Types, Complexity analysis, Big O notation, Stacks (Linked lists and array implementation), Recursion and analyzing recursive algorithms, divide and conquer algorithms, sorting algorithms (Selection, insertion, merge, quick, bubble, heap, redix), Queue, Deque, priority queues, linked lists and its various types, searching and sorting in arrays, hashing and indexing, trees and tree traversals, binary search trees, heaps, graphs, breadth first and depth first traversal, shortest path, adjacency matrix.

**Books Recommended:**

1. Robert Lafore, “Data structures and Algorithm analysis in java”, Latest Edition
2. Weiss Mark Allen, “Data structures & Algorithms Analysis”, Latest Edition
3. Herbert Schildt, “The Complete Reference Java2”, Latest Edition
4. Semour Lipschutz, “Data Structure”, Schaum’s outline series, Latest Edition

**Practical Work to be carried out:**

1 To become familiar with conditional and control structures in Java.

2 Implementation of Objects and classes.

3 Implementation of Arrays in Java

4 Implementation of Linear Search Algorithms.

5 Implementation of Binary Search Algorithms.

6 Implementation of Linked list.

7 Implementation of Queues.

8 Implementation of Stacks.

9 Implementation of Merge sort & Heap sort.

10 Implementation of Bubble sort, Quick Sort and insertion sort.

11 Implementation of Recursion.

12 Implementation of Trees.

13 Implementation of Graph.

14 Implementation of Binary tree traversals.

15 Case study

|  |  |  |
| --- | --- | --- |
| **Approval:** |  | |
| **Board of Studies:** | **Resolution No. 02** | **Dated: 29-08-2019** |
| **Board of Faculty:** | **Resolution No. 01** | **Dated: 07-10-2019** |
| **Academic Council:** | **Resolution No. 96.10** | **Dated: 07-10-2019** |