Prasad V. Potluri Siddhartha Institute of Technology, Kanuru, Vijayawada.

1/2 M.Tech. FIRST SEMESTER

CSCS1L1	**DATA STRUCTURES LAB**	**Credits: 2**
Lecture: 4 periods/week		**Internal assessment: 25 marks**
Tutorial: --		**Semester end examination: 50 marks**

Objectives:

1. To implement recursive functions.
2. To implement stack, queue, linked list, tree and graph data structures.
3. To arrange data using different sorting techniques.

Learning Outcome:

1. To describe the usage of various data structures.
2. To explain the operations for maintaining common data structures.
3. To write programs using linked structures such as List, trees, and graphs.
4. To choose, design and apply appropriate data structures for solving computing problems.
5. To analyze algorithms and to determine algorithm correctness and time efficiency.
6. To demonstrate various methods of organizing large amounts of data and arrange the data.
7. Implement the following exercises using ‘C’ Programming language.

Implementation of Data Structures and Algorithms using C.

1. To perform various operations such as insertion, deletion, display on single linked lists.
2. To implement
   (i) Stacks using linked list.
   (ii) Queues using linked list.
3. To perform different types of searching techniques on a given list
   (i) Sequential search
   (ii) Transpose sequential search
   (iii) Binary search
   (iv) Fibonacci search
4. To perform different types of sortings on a given list
   (i) Bubble sort
   (ii) Insertion sort
   (iii) Selection sort
   (iv) Merge sort
5. To perform different types of sortings on a given list
   (i) Quick sort
   (ii) Shell sort
   (iii) Radix sort
   (iv) Topological sort
6.
   (i) To convert the given infix expression to postfix expression.
   (ii) To evaluate the given postfix expression.

7. To perform various operations on graphs
   - (i) Vertex insertion
   - (ii) Vertex deletion
   - (iii) Edge insertion
   - (iv)Edge deletion
   - (v) BFS
   - (vi) DFS

8. To implement dictionaries using hasing technique.

9. To perform various operations on binary heap.

10. To perform various operations on Binary search tree.

11. To perform operations on AVL trees.

12. To perform various operations on B-tree.