|  |
| --- |
| **SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES** |
| **COMPUTER SCIENCE AND ENGINEERING PROGRAMME** |

**CSA0369- DATA STRUCTURES WITH TOPOLOGICAL SORTING**

**LIST OF PROGRAMS**

1. Write a C program to perform Matrix Multiplication
2. Write a C program to find Odd or Even number from a given set of numbers
3. Write a C program to find Factorial of a given number without using Recursion
4. Write a C program to find Fibonacci series without using Recursion
5. Write a C program to find Factorial of a given number using Recursion
6. Write a C program to find Fibonacci series using Recursion
7. Write a C program to implement Array operations such as Insert, Delete and Display.
8. Write a C program to search a number using Linear Search method
9. Write a C program to search a number using Binary Search method
10. Write a C program to implement Linked list operations
11. Write a C program to implement Stack operations such as PUSH, POP and PEEK
12. Write a C program to implement the application of Stack (Notations)
13. Write a C program to implement Queue operations such as ENQUEUE, DEQUEUE and Display
14. Write a C program to implement the Tree Traversals (Inorder, Preorder, Postorder)
15. Write a C program to implement hashing using Linear Probing method
16. Write a C program to arrange a series of numbers using Insertion Sort
17. Write a C program to arrange a series of numbers using Merge Sort
18. Write a C program to arrange a series of numbers using Quick Sort
19. Write a C program to implement Heap sort.
20. Write a program to perform the following operations:

a) Insert an element into a AVL tree.

b) Delete an element from a AVL tree.

c) Search for a key element in a AVL tree.

1. Write a C program to Graph traversal using Breadth First Search
2. Write a C program to Graph traversal using Depth First Search
3. Implementation of Shortest Path Algorithms using Dijkstra’s Algorithm
4. Implementation of Minimum Spanning Tree using Prim’s Algorithm
5. Implementation of Minimum Spanning Tree using Kruskal Algorithm

INTERNAL EXAMINER EXTERNAL EXAMINER