**DATA STRUCTURE LAB**

**(20MCA135)**

# LAB RECORD

Submitted in partial fulfilment of the requirements for the award of the degree of Master of Computer Applications of A P J Abdul Kalam Technological University, Kerala.

**Submitted by:**

## MERIN BABU (SJC23MCA-2042)



# MASTER OF COMPUTER APPLICATIONS

## ST.JOSEPH’S COLLEGE OF ENGINEERING AND TECHNOLOGY, PALAI

**CHOONDACHERRY P.O, KOTTAYAM**

**KERALA**

**December 2023**

# ST. JOSEPH’ S COLLEGE OF ENGINEERING AND TECHNOLOGY, PALAI

**(An ISO 9001: 2015 Certified College)**

**CHOONDACHERRY P.O, KOTTAYAM, KERALA**



## CERTIFICATE

This is to certify that the Data Structure Lab Record (20MCA135) submitted by Merin Babu, student of First semester MCA at ST. JOSEPH’S COLLEGE OF ENGINEERING AND TECHNOLOGY, PALAI in partial fulfilment for the award of Master of Computer Applications is a bonafide record of the lab work carried out by her under our guidance and supervision. This record in any form has not been submitted to any other University or Institute for any purpose.

|  |  |
| --- | --- |
| **Prof. Anish Augustine K** | **Dr. Rahul Shajan** |
| **Faculty In- Charge** | **(Head of the Department)** |

Submitted for the End Semester Examination held on **Examiner 1:**

**Examiner 2:**

## DECLARATION

I Merin Babu, do hereby declare that the Data Structure Lab Record (20 MCA 135) is a record of work carried out under the guidance of Mr. Anish Augustine, Asst. Professor, Department of Computer Applications, SJCET, Palai as per the requirement of the curriculum of Master of Computer Applications Programme of A P J Abdul Kalam Technology University, Thiruvananthapuram. Further, I also declare that this record has not been submitted, full or part thereof, in any University / Institution for the award of any Degree / Diploma.

Place: Choondacherry MERIN BABU

Date : (SJC23MCA-2042)

## CONTENT

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Program List** | **Page No.** |
| 1 | Array Operation (Insertion, Deletion, Sorting, Merging). | 1 |
| 2 | Searching an array element (Linear Search, Binary Search). |  |
| 3 | Matrix Operations (Addition, Multiplication, Transpose). |  |
| 4 | Using Structure, add two distances in the inch-feet system. |  |
| 5 | Implement Stack Operations. |  |
| 6 | String Operations (Searching, Concatenation, Substring). |  |
| 7 | Sorting an Array (Bubble Sort, Selection Sort, Insertion Sort). |  |
| 8 | Implement Queue operations (Insert, delete, display front &amp; rear values). |  |
| 9 | Implement Circular Queue operations (Insert, delete, display front &amp; rear values). |  |
| 10 | Implement singly linked list (Insert at the head, insert at tail, insert at a position, delete at the  head, delete at tail, delete form a position, search an element). |  |
| 11 | Implement doubly linked list (Insert at the head, insert at tail, insert at a position, delete at the  head, delete at tail, delete form a position, search an element). |  |
| 12 | Implement circular linked list (Insert at the head, insert at tail, insert at a position, delete at the  head, delete at tail, delete form a position, search an element). |  |
| 13 | Implement binary search tree. |  |
| 14 | Implement balanced-binary-search tree. |  |
| 15 | Implement set operations (union, intersection, difference). |  |
| 16 | Implement disjoint set operations. |  |
| 17 | Implement tree traversal methods DFS (In-order, Pre-Order, Post-Order), and BFS. |  |
| 18 | Implement Binomial Heaps and operations (Create, Insert, Delete). |  |
| 19 | Implement B Trees and its operations. |  |
| 20 | Implement Red Black Trees and its operations. |  |
| 21 | Graph Traversal techniques (DFS and BFS) and Topological Sorting. |  |
| 22 | Finding the Strongly connected Components in a directed graph. |  |
| 23 | Prim’s Algorithm for finding the minimum cost spanning tree. |  |
| 24 | Kruskal’s algorithm using the Disjoint set data structure. |  |
| 25 | Single Source shortest path algorithm using any heap structure that supports mergeable heap  operations. |  |