#### **DEPARTMENT OF COMPUTER SCIENCE**

#### **RAJAGIRI COLLEGE OF SOCIAL SCIENCES**

#### **(Autonomous)**

****

### MASTER OF COMPUTER APPLICATIONS

2025 - 2027

### MCA102

**DATA STRUCTURES USING C**

**NAME : ALBIN MAMMEN MATHEW**

**SEMESTER : FIRST**

**REGISTER NO. :**

#### 

#### **DEPARTMENT OF COMPUTER SCIENCE**

#### **RAJAGIRI COLLEGE OF SOCIAL SCIENCES**

#### **(Autonomous)**

KALAMASSERY - KOCHI – 683104

###### CERTIFICATE

NAME : ALBIN MAMMEN MATHEW

**SEMESTER : FIRST**

REGISTER NO. :

Certified that this is a bonafide record of work done by **ALBIN MAMMEN MATHEW** in the Software Laboratory of MCA102 subject in the First semester Master of Computer Application examinations <YEAR>, Department of Computer Science, Rajagiri College of Social Sciences (Autonomous), Kalamassery.

Mr. Ananthakrishnan K V Dr. Bindiya M Varghese

Faculty in Charge Dean, Computer Science

Internal Examiner External Examiner

Place : Kalamassery

Date :

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| Si. No | Activity | Page No |
| 1. | Storage Classes in C | 1 |
| 2. | Array Operations using Global Variables | 2 |
| 3. | Array Operations using Local Variables | 8 |
| 4. | Searching for Occurrences of an Element in an Array | 15 |
| 5. | Sorting Array Elements in Ascending Order | 17 |
| 6. | Displaying Array Elements Using Recursion | 20 |
| 7. | Displaying Array Elements in Reverse Order Using Recursion | 21 |
| 8. | Matrix Addition and Subtraction | 22 |
| 9. | Matrix Multiplication | 25 |
| 10. | Finding the Transpose of a Matrix | 28 |
| 11. | Determinant of a Matrix (2×2 and 3×3) | 30 |
| 12. | Implementation of Stack Operations using Arrays | 32 |
| 13. | Display a String in Reverse Order | 35 |
| 14. | Reverse a String in the Same Array | 36 |
| 15. | Sorting N Strings in Ascending Order | 37 |
| 16. | Reversing a String using Stack | 39 |
| 17. | Conversion of Infix Expression to Postfix using Stack | 43 |
| 18. | Conversion of Infix Expression to Prefix using Stack | 47 |
| 19. | Evaluation of an Infix Expression using Stack | 51 |
| 20. | Evaluation of Expression by Converting Infix to Postfix | 53 |
| 21. | Sequence of Pop Operations for a Given Stack Operation Sequence | 57 |
| 22. | Read and Display a Sparse Matrix | 59 |
| 23. | Addition of Two Sparse Matrices | 62 |
| 24. | Multiplication of Two Sparse Matrices | 66 |
| 25. | Reading and Displaying a Polynomial | 70 |
| 26. | Addition of Two Polynomials | 72 |
| 27. | Subtraction of Two Polynomials | 75 |
| 28. | Multiplication of Two Polynomials | 78 |
| 29. | Implementation of Queue | 81 |
| 30. | Implementation of Circular Queue | 86 |
| 31. | Implementation of Heapsort | 91 |
| 32. | Implementation of Priority Queue | 93 |
| 33. | Date Structure: Read, Display and Compare Two Dates | 98 |
| 34. | Employee Structure Operations (Search, Sort, Delete) | 100 |
| 35. | Reading and Displaying a Polynomial (Using Structure Array) | 108 |
| 36. | Addition of Two Polynomials (Using Structures) | 111 |
| 37. | Subtraction of Two Polynomials (Using Structures) | 115 |
| 38. | Multiplication of Two Polynomials (Using Structures) | 119 |
| 39. | Demonstration of malloc(), calloc(), and free() | 123 |
| 40. | Finding Mean of N Integers using malloc() | 125 |
| 41. | Finding Mode of N Numbers using calloc() | 127 |
| 42. | Array of Books using Pointers (Read and Display Functions) | 129 |
| 43. | Implementation of varchar using realloc() | 132 |
| 44. | Creation and Display of a Singly Linked List | 134 |
| 45. | Insertion Operations in Singly Linked List | 137 |
| 46. | Deletion Operations in Singly Linked List | 141 |
| 47. | Display Linked List in Reverse Order | 145 |
| 48. | Sorting a Linked List (By Value Swap and Address Swap) | 148 |
| 49. | Polynomial Addition and Multiplication using Linked List | 152 |
| 50. | Linked List of Names – Insert, Delete, Display, Sort, Reverse, Count | 157 |
| 51. | Implementation of Stack using Linked List | 165 |
| 52. | Implementation of Queue using Linked List | 169 |
| 53. | Implementation of Circular Linked List | 173 |
| 54. | Implementation of Circular Linked Queue | 179 |
| 55. | Implementation of Doubly Linked List | 183 |
| 56. | Implementation of Circular Doubly Linked List (String Data) | 191 |
| 57. | Linear Search in an Array | 198 |
| 58. | Binary Search in an Array of Integers | 200 |
| 59. | Binary Search in an Array of Strings | 202 |
| 60. | BST Insertion and Recursive Traversals (Inorder, Preorder, Postorder) | 204 |
| 61. | BST Insertion and Inorder Traversal (Without Recursion) | 207 |
| 62. | BST Insertion and Preorder Traversal (Without Recursion) | 209 |
| 63. | BST Insertion and Postorder Traversal (Without Recursion) | 211 |
| 64. | BST Insertion using Names and Display in Ascending Order (Inorder Traversal) | 214 |
| 65. | Graph Representation using Adjacency Matrix (Using Arrays) | 216 |
| 66. | Graph Representation using Adjacency List (Using Linked Lists) | 218 |
| 67 | Project: Minesweeper | 222 |