MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Syllabus

(List of Experiments/ Lab & Skill Based mini Projects)

of

B.Tech.

in

Mathematics and Computing

Department of Engineering Mathematics and Computing

Madhav Institute of Technology & Science
Gwalior-474005

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Department of Engineering Mathematics and Computing

Computer Lab (2250306)
Numerical Techniques Using MATLAB

List of Topics in Experiments

1. Introduction to MATLAB and Simple Calculations with MATLAB.
2. Creating Arrays and Mathematical Operations in MATLAB.
3. Two Dimensional Plots in MATLAB.
4. User Define function and function file.
5. Loops and Conditional Statements in MATLAB.
6. Polynomial and Interpolation.
7. Application on Numerical Methods :
   a. Solving Algebraic Equations of one variables
   b. Finding Maxima & Minima
   c. Numerical Integration.
   d. Ordinary Differential Equation.
8. Three dimensional Plots.

List of Experiments

1. If X=[1 4; 8 3], find

a) the inverse matrix of X .
b) the diagonal of X.
c) the sum of each column and the sum of whole matrix X.
d) the transpose of X.
2. Plot Sinc function In matlab , where Sinc $(x) = \sin(x) / x$, and $-2\pi \leq x \leq 2\pi$
3. 3-D Plot of function: y=xcos(x); z=exp(x/5)cos(x)+1 for $0 \leq x \leq 6\pi$.
4. Root Finding
   a. Program for roots of f(x)=0 by Newton Raphsonm method
   b. Program for roots of f(x)=0 by Bisection method
   c. Program for roots of f(x)=0 by Regula-Falsi method.
5. Solution of a system of simultaneous algebraic equations using the Gaussian Elimination procedure.
6. Determination of Eigenvalues and Eigenvectors of a square matrix.
7. Solution of a system of simultaneous algebraic equations using the Gauss-Seidel iterative method.
8. Program for solving to integral of a given function using Trapezoidal Rule
9. Program for solving numerical integration by Simpson's 1/3 rule.
10. Program for solving numerical integration by Simpson's 3/8 rule.
11. Program for solving numerical solution of an ordinary differential equation using the Euler's method.
12. Program for solving numerical solution of an ordinary differential equation using the Runge-Kutta -4th order method.

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Department of Mathematics & Computing
Data Structures and Algorithms
(MAC-2250324)

Implement the following in C/C++

1. Linked List Insertion
2. Linked List Deletion (Deleting a given key)
3. Linked List Deletion (Deleting a key at given position)
4. A Programmer's approach of looking at Array vs. Linked List
5. Find Length of a Linked List (Iterative and Recursive)
6. How to write C functions that modify head pointer of a Linked List?
7. Swap nodes in a linked list without swapping data
8. Reverse a linked list
9. Merge two sorted linked lists
10. Merge Sort for Linked Lists
11. Reverse a Linked List in groups of given size
12. Detect and Remove Loop in a Linked List
13. Add two numbers represented by linked lists | Set 1
14. Rotate a Linked List
15. Circular Linked List Introduction and Applications,
16. Circular Singly Linked List Insertion<
17. Circular Linked List Traversal
18. Split a Circular Linked List into two halves
19. Sorted insert for circular linked list
20. Doubly Linked List Introduction and Insertion
21. Delete a node in a Doubly Linked List
22. Reverse a Doubly Linked List
23. QuickSort on Doubly Linked List
24. Merge Sort for Doubly Linked List
25. Introduction to Stack
26. Infix to Postfix Conversion using Stack
27. Evaluation of Postfix Expression
28. Reverse a String using Stack
29. Implement two stacks in an array
30. Check for balanced parentheses in an expression
31. Next Greater Element
32. Reverse a stack using recursion
33. Sort a stack using recursion
34. Design and Implement Special Stack Data Structure
35. Implement Stack using Queues
36. Design a stack with operations on middle element
37. How to efficiently implement k stacks in a single array?
38. Sort a stack using recursion
39. Queue Introduction and Array Implementation
40. Linked List Implementation of Queue
41. Applications of Queue Data Structure
42. Priority Queue Introduction
43. Deque (Introduction and Applications)
44. Implementation of Deque using circular array
45. Implement Queue using Stacks
46. Linear Search, Binary Search, Jump Search, Interpolation Search, Exponential Search
47. Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Heap Sort, Quicksort, Radix Sort, Counting Sort, Bucket Sort, Shell Sort.

48. Tree Traversals
49. BFS vs DFS for Binary Tree
50. Level Order Tree Traversal
51. Diameter of a Binary Tree
52. Inorder Tree Traversal without Recursion
53. Inorder Tree Traversal without recursion and without stack!
54. Threaded Binary Tree
55. Maximum Depth or Height of a Tree
56. If you are given two traversal sequences, can you construct the binary tree?
57. Clone a Binary Tree with Random Pointers
58. Construct Tree from given Inorder and Preorder traversals
59. Maximum width of a binary tree
60. Print nodes at k distance from root
61. Print Ancestors of a given node in Binary Tree
62. Check if a binary tree is subtree of another binary tree
63. Connect nodes at same level
64. Search and Insert in BST
65. Deletion from BST
66. Minimum value in a Binary Search Tree
67. Inorder predecessor and successor for a given key in BST
68. Check if a binary tree is BST or not
69. Lowest Common Ancestor in a Binary Search Tree.
70. Inorder Successor in Binary Search Tree
71. Binomial Heap
72. Fibonacci Heap
73. Heap Sort
74. Separate Chaining for Collision Handling
75. Open Addressing for Collision Handling
76. Breadth First Traversal for a Graph
77. Depth First Traversal for a Graph
78. Applications of Depth First Search
79. Applications of Breadth First Traversal
80. Detect Cycle in a Directed Graph
81. Detect Cycle in a an Undirected Graph
82. Detect cycle in an undirected graph