

Data Structures Lab

List of Experiments

1. a) Write a C program to Print 1 to 'n' numbers without using any loops.
b) Write a C program to reverse the digits of a number using recursive function.
c) Write a C program to solve the Towers of Hanoi problem using recursive function.

2. a) Design & Develop a C program to perform linear search for a key value in a given list.
b) Design & Develop a C program to perform Binary search for a key value in a given list.

3. Given a File of N employee records with a set K of Keys(6-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table(HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers.

Design and develop a Program in C that uses Hash function H: KL as $H(K)=K \bmod m$ (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.

4. a) Design, Develop and Implement a C program that implement Selection Sort to sort a given list of integers.
b) Design, Develop and Implement a C program that implement Bubble Sort to sort a given list of integers.

5. a) Design, Develop and Implement C program that implement Quick Sort to sort a given list of integers.
b) Design, Develop and Implement C program that implement Heap Sort to sort a given list of integers.

6. Design, Develop and Implement a menu driven Program in C to represent polynomials using a single linked list and implement functions Polynomials. Each Node: [Coefficient, Exponent, Next]

a) Create Polynomials using Last Insertion of nodes, in a decreasing order of exponents.

Example: Polynomial P: " $7x^3 + 3x^2 + 9x + 6$ " has 4 nodes. 7,3 3,2,9,16,0 → Null

b) Perform Addition of 2 polynomials. (coeff.s of 2 nodes are added only if exp.s match)

Example: Let Polynomials P: " $7x^3 + 3x^2 + 9x + 6$ " & Q: " $5x^4 + 4x^2 + 6x$ ". Then

$P+Q=5x^4+7x^3+7x^2+15x + 6$ ".

c) Perform Subtraction of 2 polynomials.

d) Display the Polynomial (nodes) in it.

e) Exit

7. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)

a) Push an Element onto Stack.

b) Pop an Element from Stack.

c) Demonstrate how Stack can be used to check Palindrome.

d) Display the status (No. of elements, Empty/Full/not) of Stack.

e) Exit

Support the program with appropriate functions for each of the above operations

8. Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^(Power) and alphanumeric operands.

9. Design, Develop and Implement a menu driven Program in C for the following operations on QUEUE of Characters (Array Implementation of Queue with maximum size MAX)

a) Insert an Element into QUEUE

b) Delete an Element from QUEUE

c) Display the status (No. of elements, Empty/Full/not) of QUEUE

d) Exit

Support the program with appropriate functions for each of the above operations

10. a) Design, Develop and Implement a C program to implement Binary tree traversals using iterative functions.

b) Design, Develop and Implement a C program to implement Binary tree traversals using recursive functions.

11. a) Design, Develop and Implement a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers a. Create a BST of N Integers: 89, 38, 24, 40, 95, 76, 57, 11, 59, 18

b) Traverse the BST in In-order, Pre-order and Post-Order

c) Search the BST for a given element (Key) and report the appropriate message

d) Exit

12. a) Design & Develop a Program in C for the following operations on Graph (G) of Cities

a. Create a Graph of N cities using Adjacency Matrix.

b) Print all the nodes reachable from a given starting node in a digraph using DFS or BFS method.

c) Find the shortest paths to all Cities from a given City using Dijkstra's Algorithm.