```
mysteryFunction (n - 1);

cout << n << " ";

mysteryFunction (n-1);

}

}
```


Trace the function calls and show the sequence of outputs.

(c) A stack is implemented using an array with a maximum size of 5. The following operations are performed on the stack in sequence: (5)

PUSH (10), PUSH (20), POP (), PUSH (30), PUSH (40)

Explain what happens at each step, the final contents of the stack, and how overflow or underflow conditions are handled.

[This question paper contains 12 printed pages.]

Your Roll No..............

Sr. No. of Question Paper : 4664 **J**

Unique Paper Code : 2344002003

Name of the Paper : DATA STRUCTURES USING C++

Name of the Course : **GENERIC ELECTIVE (COMMON PROGRAM)**

Semester : IV

Duration : 3 Hours Maximum Marks : 90

Instructions for Candidates

1. Write your Roll. No. on the top immediately on receipt of this question paper.

2. **Section A** is compulsory.

3. Attempt **any 4 (four)** questions from **Section B.**

4. All parts of a question must be answered together.

SECTION - A

1. (a) Find the asymptotic notation (Big O) for the following nested loop: (3)

```
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        int x=5;
    }
}
```

(b) Write a function in C++ for inserting an element at the beginning of a singly linked list. (3)

(c) Evaluate the following expression using stacks: (3)

5+2* (3-4+2/3) *4+2

Show stack after each step.

(d) Consider the following array: (3)

(b) Construct a min-heap from the array: **[10, 20, 15, 30, 40]** and show heap at each step. (5)

7. (a) Consider the following array of student scores in descending order of registration: **[90, 70, 50, 80, 60, 85]**

Applying the insertion sort method, execute the sorting process to sort the scores in ascending order. (5)

(b) Given the following recursive function, determine the output when the input is n = 3. (5)

```
#include <iostream>

using namespace std;


void mysteryFunction (int n) {
    if (n > 0) {
```

(c) Write a program to merge two sorted arrays (or lists) into a single sorted list. (5)

Input:

List 1: 1, 3, 5

List 2: 2, 4, 6

6. (a) A hash table of size 10 is implemented using open addressing with linear probing for collision resolution. The hash function used is: (10)

h(k) = k mod10

Insert the following keys into the hash table in the given order:

22, 42, 32, 52, 62

Show the status of the hash table after all insertions. Indicate each step clearly and explain how collisions are handled using linear probing.

A=[5,12,18,23,31,42,56,64,77,89]

Search for 33 using binary search on the array show each step clearly.

(e) An integer array **A[4] [5]** is stored in **row-major order.** The **base address** of the array is 2000 and every element takes 4 bytes of memory. Find the address of element **A [2] [3].** (3)

(f) Consider the following sequence of elements: (3)

40, 20, 60, 10, 30, 50, 70

Construct a **Binary Search Tree (BST)** by inserting the elements one at a time. Draw the tree **after each insertion** to show its construction step by step.

(g) Write a recursive function to calculate factorial of a number. (3)

(h) What will be the output of the following code in C++? (3)

```
#include <iostream>

#include <stack>

using namespace std;

int main() {

    stack<int> s;

    s.push(10);

    s.push (20);

    s.push (30);

    cout << s.top() << endl;

    s.pop();

    cout << s.top() << endl;

    return 0;

}
```



```
        1
       / \
      2   3
     / \   \
    4   5   6
   /
  7
```

Perform the following tree traversals and write down the order of nodes visited for each:

(i) In-order Traversal

(ii) Pre-order Traversal

(iii) Post-order Traversal

(b) Delete node 2 in the above tree and show the resultant tree after deletion. (4)

(i) Describe how **BFS** would assist in identifying the shortest path from the beginning room (root node) to the target room (leaf node) within the maze.

(ii) Describe how **DFS** can be utilized to traverse the maze and obtain a path to the destination room.

(b) Consider the following traversals of a binary tree: (5)

- **Preorder traversal:** 10, 5, 1, 7, 40, 50
- **Inorder traversal:** 1, 5, 7, 10, 40, 50

Construct the Binary Tree using the above traversals.

5. (a) Consider the following tree: (6)

(i) Mansi has a list of recipe names stored in her cooking app. When she wants to find a recipe by checking each name one by one, she uses a basic method that doesn't need the list to be in any order. But when she sorts the list alphabetically and then looks for a specific recipe by skipping over large sections, she uses a faster method. What are the names of these two search methods, and which one is faster for large lists? (3)

(i) Convert the following infix expression to postfix: (A + B)* (C-D) (3)

Section B

2. (a) Perform the following operations on an Array: (10)

(i) Insert the elements: 10, 20, 30, 40, 50.

(ii) Delete the element 30 from the list.

(iii) Display the current list of elements after the insertions and deletions.

Perform the same operations as above on **linked list** (consider inserting the nodes from the end).

After performing the operations on both the array and the linked list, **compare** the time complexity of insertions, deletions, and access in both the array and linked list.

(b) Consider the following circular queue with a capacity of 6: (5)

Front [2, 4, 6, 8, Empty, Empty] Rear

Perform the following operations:

(i) Enqueue 10 into the queue.

(ii) Enqueue 12 into the queue.

(iii) Dequeue an element.

(iv) Enqueue 14 into the queue.

(v) Dequeue an element.

3. (a) Define AVL Trees. What is the balance factor of a node in an AVL tree, and how is it used to keep the tree balanced? (10)

Insert the following sequence of keys into an initially empty AVL tree:

30, 20, 40, 10, 25, 22

Draw the AVL tree after each insertion, clearly indicating any rotations executed to preserve the AVL property.

(b) Consider the following sequence of integers: (5)

A= 4,2,2,8,3,3,1

Perform Counting Sort on the array.

4. (a) Develop a navigation scheme for a maze, modelled as a binary tree. Each of the tree nodes is associated with a room, and edges form potential routes among rooms. The goal is to find the shortest way to cover the maze. (10)