Experiment 10

StudentName: Shubham **UID:22BCS15853**

Branch: CSE **Section/Group:** KRG 2 B

Semester: 6th **Date of Performance:** 18/04/25

Subject Name: Advanced Programming Lab-2 **Subject Code:** 22CSP-351

Problem -1

1. **Aim:** Number of 1 Bits
2. **Objective:** Given a positive integer n, write a function that returns the number of set bits in its binary representation (also known as the Hamming weight).
3. **Implementation/Code:**

```
class Solution {
public:
    int hammingWeight(int n) {
        int count=0;
        while(n){
            n=n&(n-1);
            count++;
        }
        return count;
    }
};
```

4. **Output:**

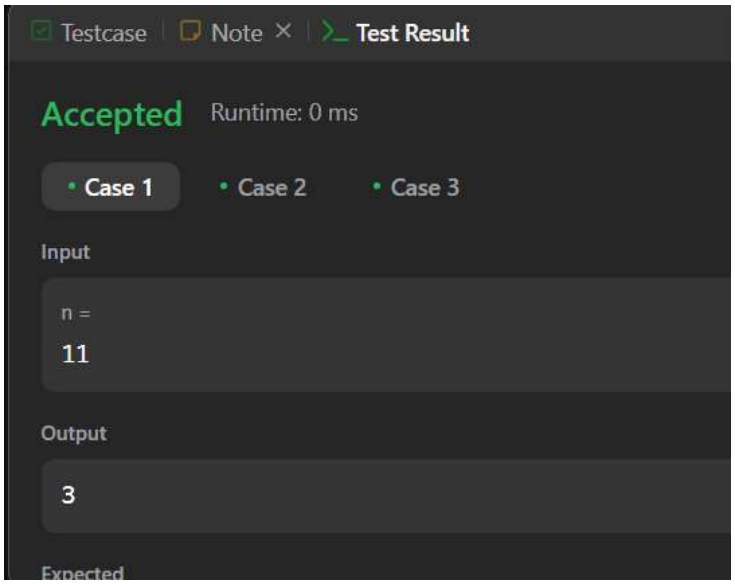

Figure 1

Problem-2

1. **Aim:**Pascal's Triangle
2. **Objectives:**Given an integer numRows, return the first numRows of Pascal's triangle.

3. **Implementation/Code:**

```
class Solution {
public:
    vector<vector<int>> generate(int numRows) {
        vector<vector<int>> result;
        for(int i=0;i<numRows;i++){
            vector<int> row(i+1,1);
            for(int j=1;j<i;j++){
                row[j]=result[i-1][j-1]+result[i-1][j];
            }
            result.push_back(row);

        }
        return result;
    }
};
```

4. **Output:**

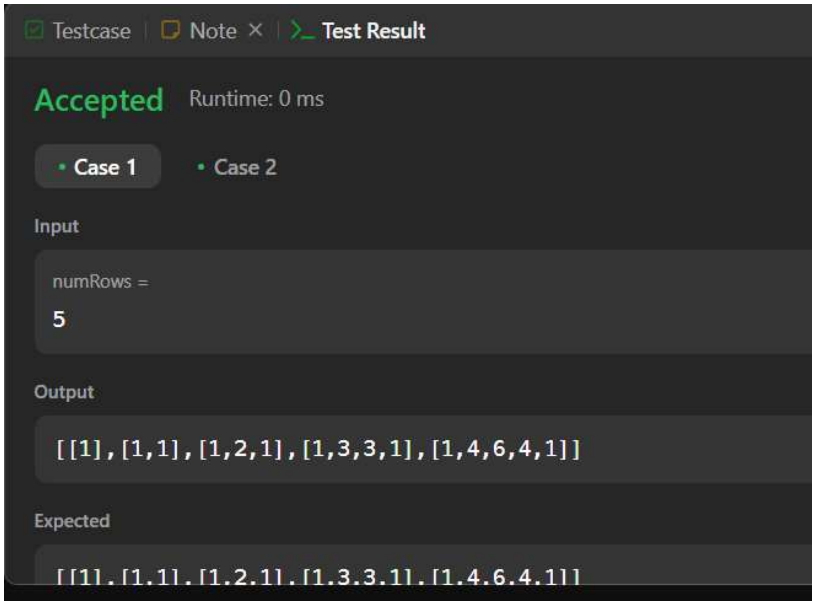

Figure 2

5. Learning Outcomes:

- **Time Complexity Optimization** – Learning approaches for problems that could have been solved with brute force .
- **Efficient Memory Usage** – Using constant space (O(1)) instead of additional arrays or recursion stacks.
- **Iterative Problem-Solving** – Implementing loops effectively to avoid unnecessary computations.