DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment 1(a)

Student Name: Karanvir Singh
UID: 22BCS16269
Branch: CSE
Section/Group: NTPP-602/A
Semester: 5th
Date of Performance: 22/01/25
Subject Name: Advanced Programming Lab-1
Subject Code: 22CSP-314

1. **Title:** Array Reversal

2. **Aim:** Given an array, of size, reverse it. Example: If array, arr= {1,2,3,4,5}, after reversing it, the array should be, arr= {5,4,3,2,1}.

3. **Objective:** The objective of the problem "Given an array of size n , reverse it" is to write a function that takes an array of a given size as input and returns a new array where the order of the elements is reversed.

4. **Implementation/Code:**

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int num, *arr, i;
    scanf("%d", &num);
    arr = (int*) malloc(num * sizeof(int));
    for(i = 0; i < num; i++) {
        scanf("%d", arr + i);
    }

int temp;
    for(i = 0; i < num / 2; i++) {
        temp = arr[i];
        arr[i] = arr[num - i - 1];
```

```
            arr[num - i - 1] = temp;
        }
        for(i = 0; i < num; i++)
            printf("%d ", *(arr + i));
        return 0;
    }
```

5. **Output:**

6. **Time Complexity:** O(n)
7. **Space Complexity:** O(n)

DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
Discover. Learn. Empower.

Experiment 1(D)

Student Name: Karanvir Singh
UID: 22BCS16269
Branch: CSE
Section/Group: NTPP-602/A
Semester: 5th
Date of Performance: 22/01/25
Subject Name: Advanced Programming Lab-1
Subject Code: 22CSP-314

1. Title: Diagonal Difference

2. Aim:

Given a square matrix, calculate the absolute difference between the sums of its diagonals.
For example, the square matrix is shown below:
1 2 3
4 5 6
9 8 9
The left-to-right diagonal =. The right to left diagonal =. Their absolute difference is.

3. Objective:

The objective of this program is to compare the ratings of two challenges created by Alice and The objective of this code is to compute the absolute difference between the sums of the diagonals of a square matrix. Given a square matrix of size $n \times n$ the program performs the following tasks:

Input Handling: It reads the size of the matrix $n$xn and the matrix elements from the standard input.

Diagonal Sum Calculation: It calculates the sum of the elements in the left-to-right diagonal and the sum of the elements in the right-to-left diagonal.

Absolute Difference: It computes the absolute difference between these two diagonal sums.

4.Implementation/Code:

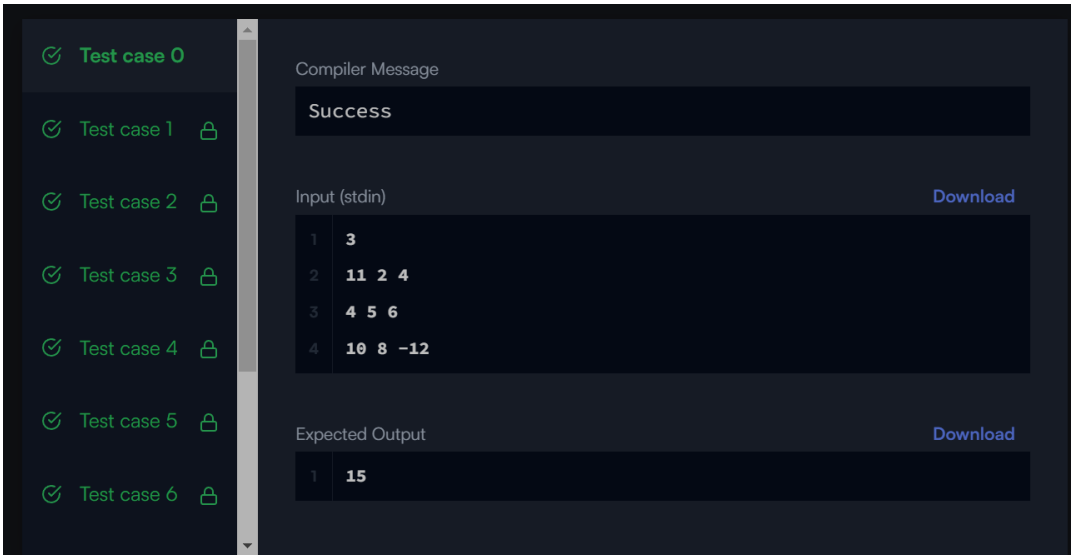
```cpp
#include <iostream>
#include <cmath>
using namespace std;

int main() {
    int n;
    cin >> n;
    int matrix[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrix[i][j];
        }
    }
    int LRDiagSum = 0;
    int RLDiagSum = 0;
    for (int i = 0; i < n; i++) {
        LRDiagSum += matrix[i][i];
        RLDiagSum += matrix[i][n - 1 - i];
    }
    int difference = abs(LRDiagSum - RLDiagSum);
    cout << difference;

    return 0;
}
```

5. Output:



6. **Time Complexity:** $O(n^2)$
7. **Space Complexity:** $O(n^2)$