Experiment 1

Student Name: Anjali Singh **UID: 20BCS9239**
Branch: CSE **Section/Group:607/A**
Semester: 5th **Date of Performance:04/09/22**
Subject Code: 21CSP-314

1. **Aim/Overview of the practical:** To implement the concept of Dynamic Array.

2. **Task to be done/ Which logistics used:**

   In this practical we are going understand various problems and find out better approach to solve particular problem related to arrays.

3. **Algorithm/Flowchart** (For programming based labs)**:**

   Reverse Array
   1.START
   2.Declare size of array and take input.
   3.Take input in array.
   4.Reverse the array.
   5.Print the array.
   6.END

Array Sum

1.START
2.Declare size of array and take input.
3.Take input in array.
4.Calculate sum of all elements of array.
5.Print the sum.
6.END

Compare The Triplets

1.START
2.Declare two array a and b with size =3 and an array called result with size 2.
3.Take input of 3 values in each array.
4.For loop (i=0 ; i<3;i++)
5. if (a[i]>b[i])
    ++result[0];
6. else if(a[i]<b[i])
    ++result[1];
7.Print result[0] and result[1].
8.END

Diagonal Difference

1.START
2.Declare variable n and take input of n.
3.Decclare a n size of 2d array.
4.Take input in 2d array using nested loop.
5.Declare sumLeftToRight and sumRightToLeft variable with value = 0.
6.For loop (i=0;i<n;i++)
7. sumLeftToRight += arr[i][i]; sumRightToLeft += arr[i][n-i-1];
8.Declare a res variable and store absolute difference of above both variables in it.
9.Print res.
10.END

4. Steps for experiment/practical/Code:

A. REVERSE AN ARRAY

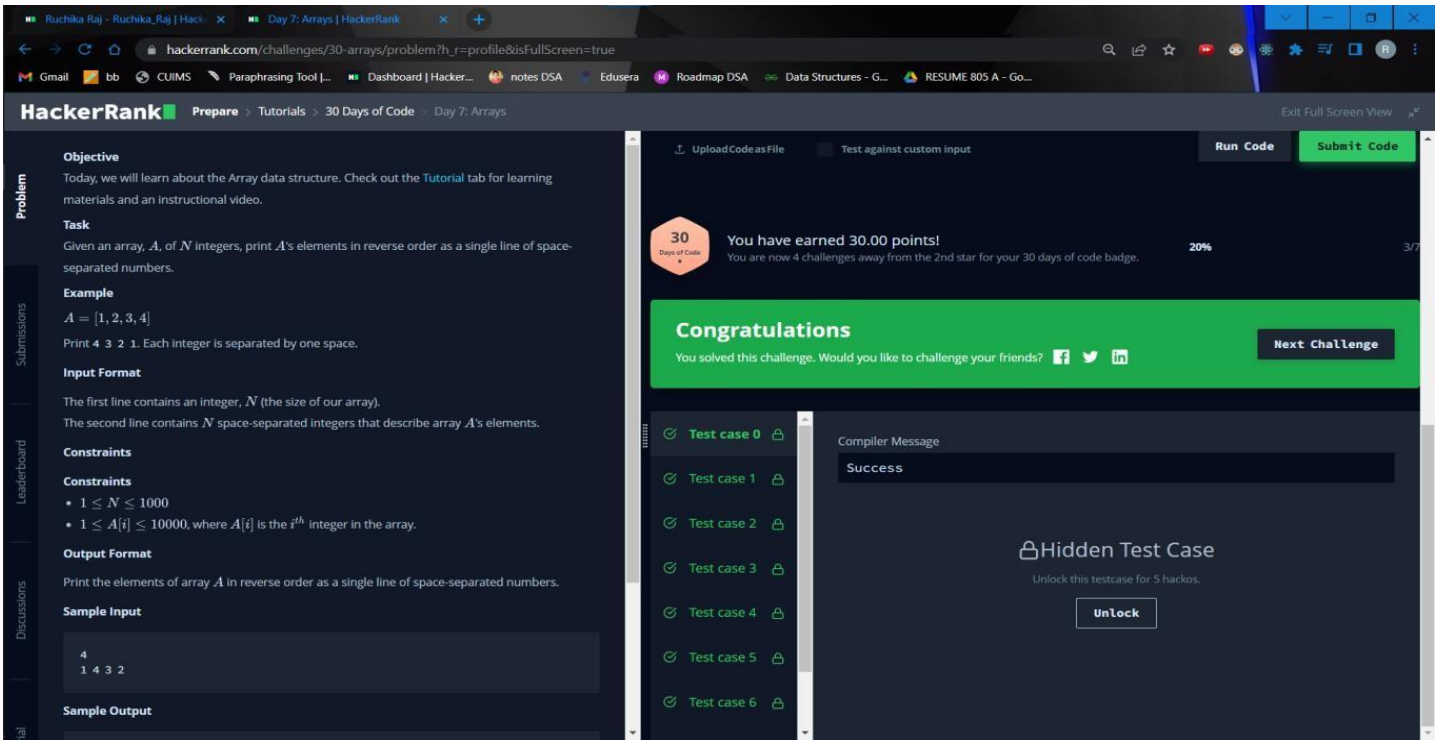
- Reversing array by implementing vector with its reverse function.

```
#include <bits/stdc++.h>
using namespace std;
int main(){
int n;
cin >> n;
vector<int> arr(n);
for(int i = 0;i < n;i++){
      cin >> arr[i];
}
reverse(arr.begin(), arr.end());
for(int i = 0;i < n;i++){
      cout<<arr[i]<<" ";
}
return 0;
}
```

- Using Two Pointer Method for Reverse an Array

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    int a[n];
    for(int i=0;i<n;i++){
        cin>>a[i];
```

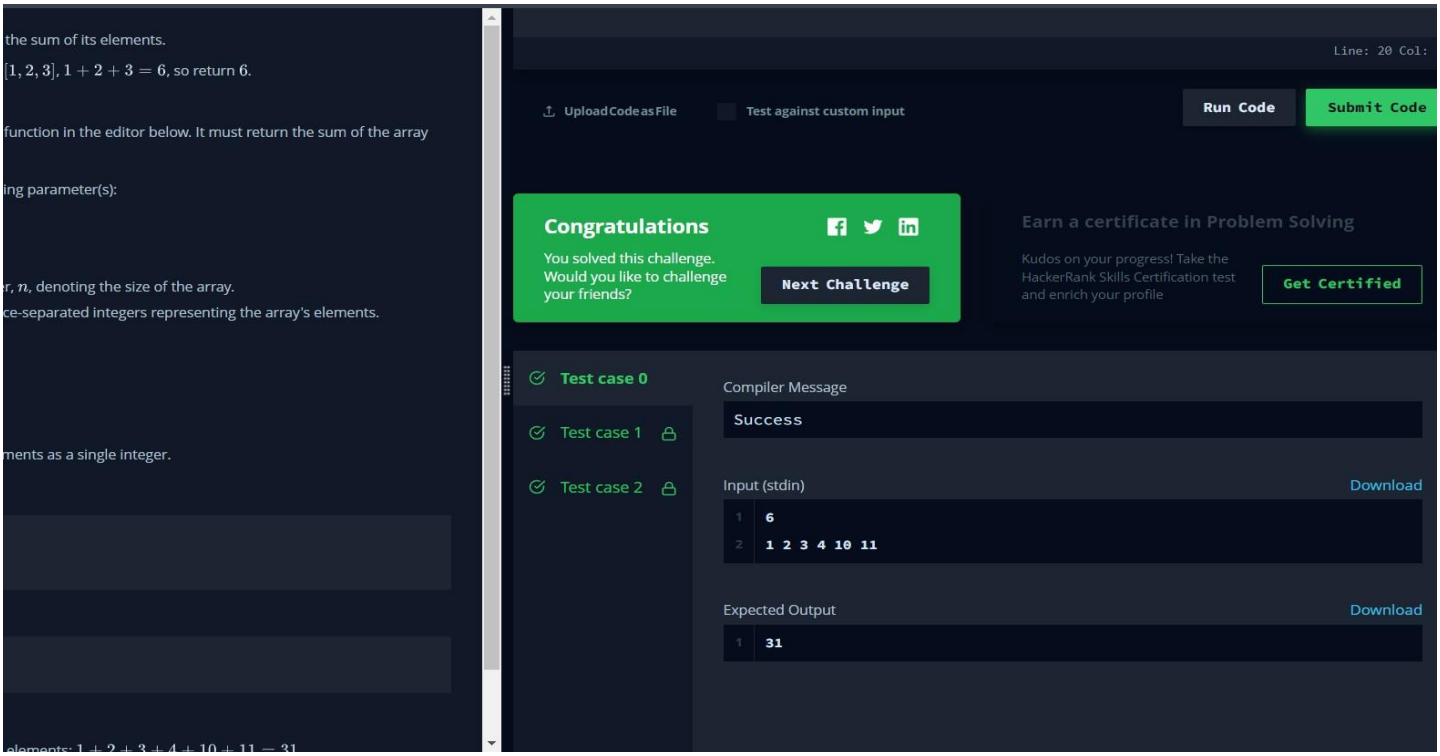
```
    }
    int start=0,end=n-1;
    while(start<end){
        swap(a[start],a[end]);
        start++;
        end--;
    }
    for(int i=0;i<n;i++){
        cout<<a[i]<<" ";
    }
    cout<<endl;
}
```



B. Sum of array

```
#include <iostream>
using namespace std;
int main() {
    int N, Sum = 0, i;
    cin>>N;
    int arr[N];
    for (i = 0 ; i < N ; i++) {
        cin>> arr[i];
        Sum += arr[i];
```

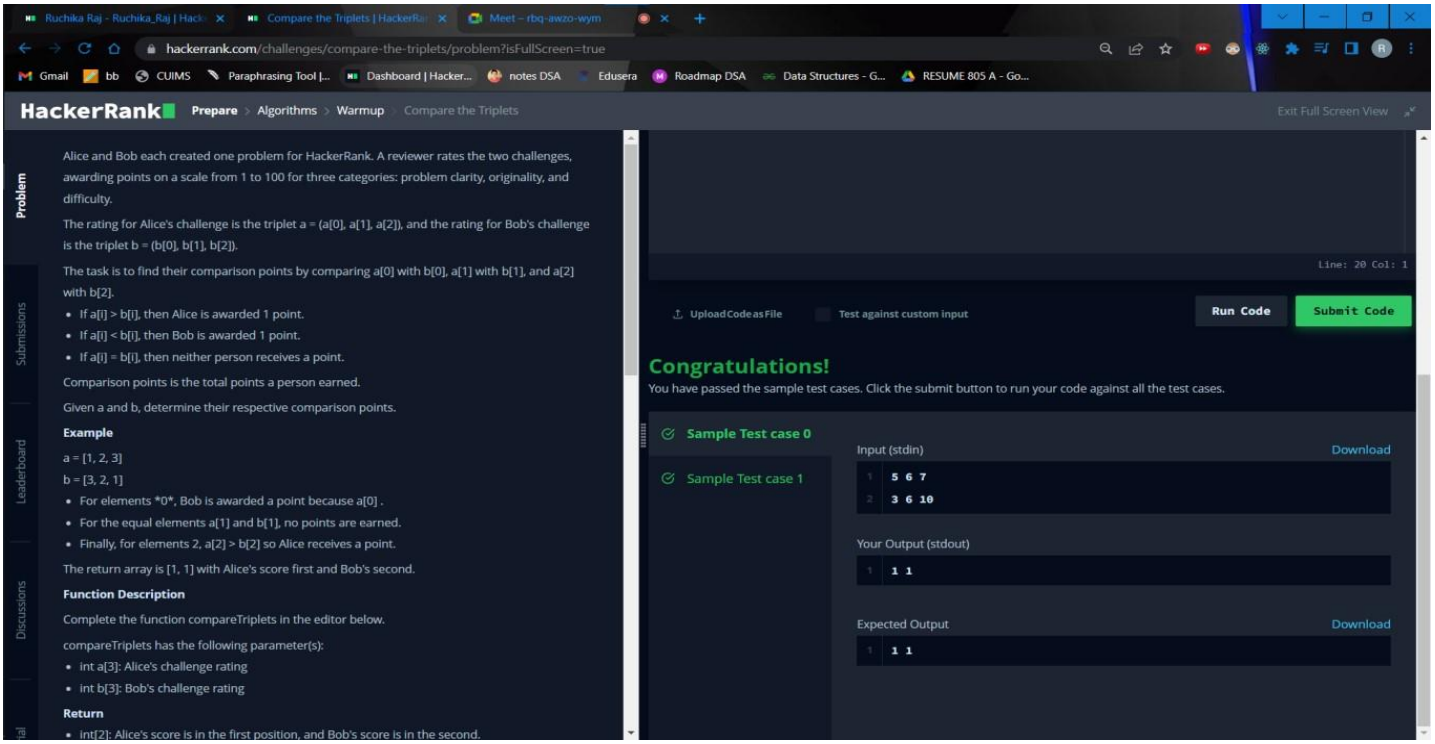
```
    }
    cout<<Sum<<endl;
       return 0;
}
```



C. Compare the Triplets

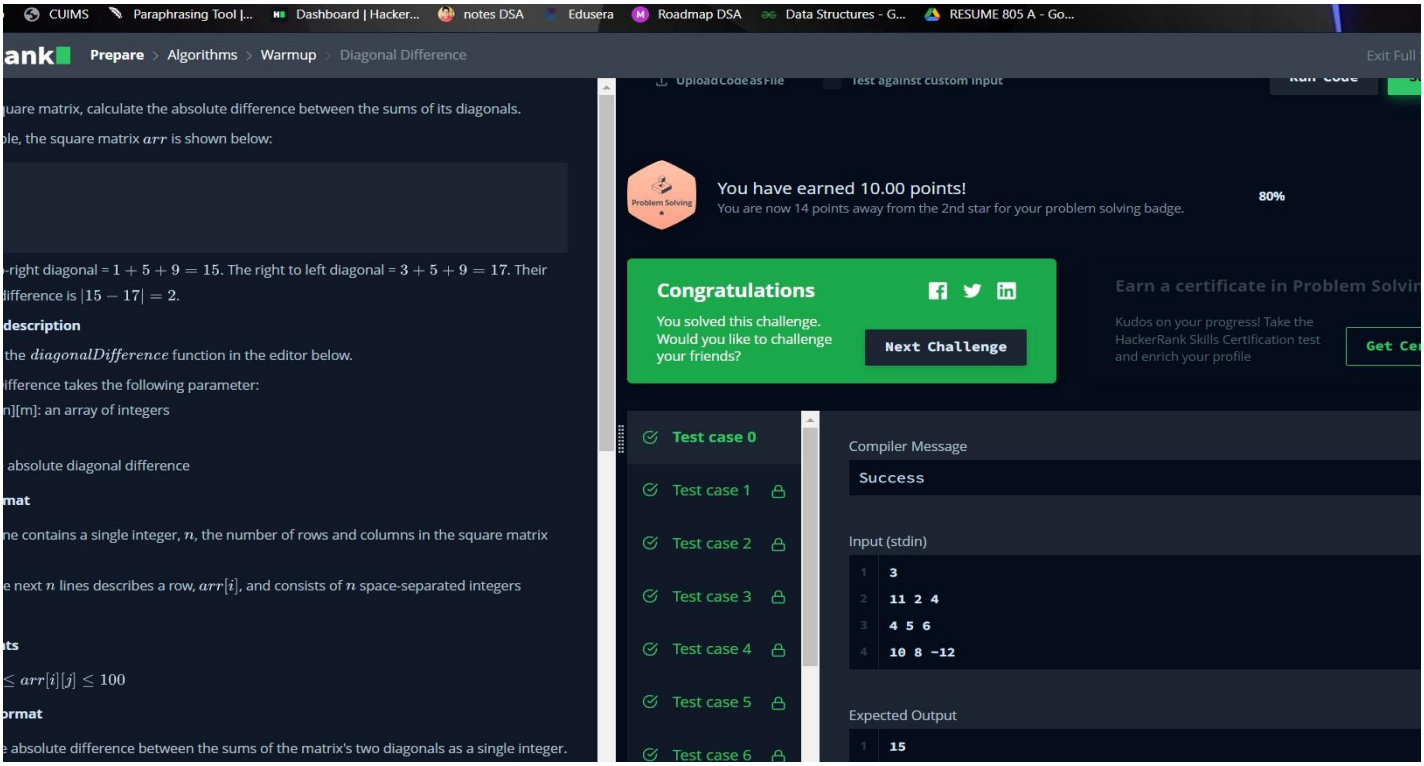
```
#include <iostream>
using namespace std;

int main()
{
    int a[3];
    int b[3];
```

```
    int result[2] = {0, 0};
    cin >> a[0] >> a[1] >> a[2];
    cin >> b[0] >> b[1] >> b[2];
    for (int i = 0; i < 3; i++){
        if (a[i] > b[i]){
            ++result[0];
        }
        else if (a[i] < b[i]){
            ++result[1];
        }
    }
    cout << result[0] << " " << result[1];
    return 0;
}
```



D. Diagonal Difference

```cpp
#include <bits/stdc++.h>
using namespace std; int main()
{
    int n, i, j;
    cin >> n;
    int arr[n][n];
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            cin >> arr[i][j];
        }
    }
    int sumLeftToRight = 0, sumRightToLeft = 0;
    for (int i = 0; i < n; i++)
    {
        sumLeftToRight += arr[i][i];
        sumRightToLeft += arr[i][n - i - 1];
    }
    int res = abs(sumRightToLeft - sumLeftToRight);
    cout << res << endl;
    return 0;
}
```

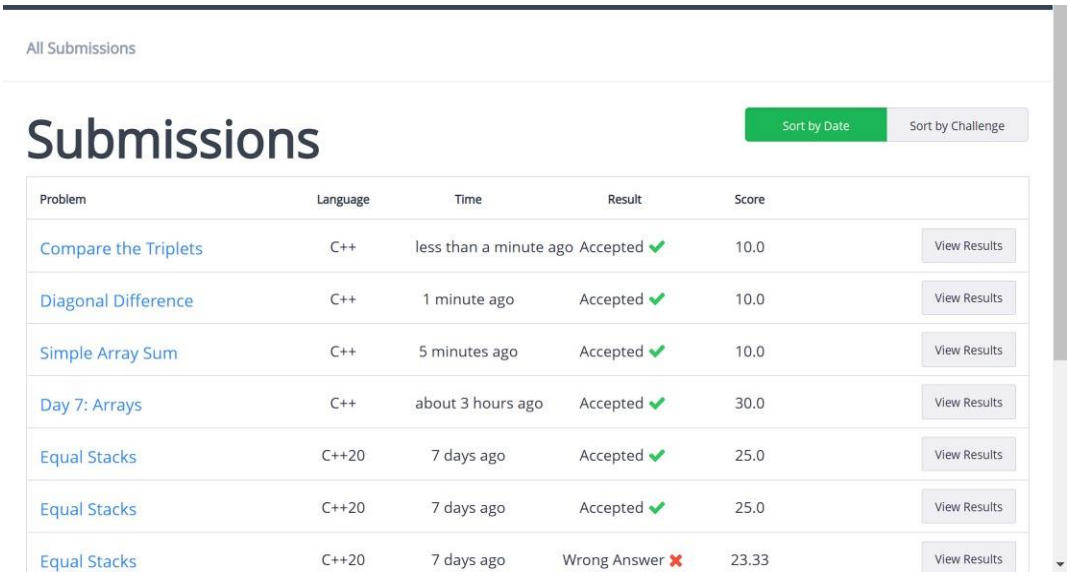


5. Observations/Discussions/ Complexity Analysis:

As per different types of question in question number:-

1. In reverse array, we reverse the array and print it,
2. In simple array sum ,we take an array and calculate sum of all the elements of that array and then print it.
3. In compare the triplets, we take two different array of size 3, we store 3 elements in each array and then compare elements of the array index wise.
4. In diagonal difference, we take a 2d array having 3 rows and 3 columns and first we find sum of right diagonal elements and then left diagonal elements and later we calculate difference between both right and left diagonal sum.

6. Result/Output/Writing Summary:



All Submissions

Submissions

Sort by Date | Sort by Challenge

Problem	Language	Time	Result	Score	
Compare the Triplets	C++	less than a minute ago	Accepted ✔	10.0	View Results
Diagonal Difference	C++	1 minute ago	Accepted ✔	10.0	View Results
Simple Array Sum	C++	5 minutes ago	Accepted ✔	10.0	View Results
Day 7: Arrays	C++	about 3 hours ago	Accepted ✔	30.0	View Results
Equal Stacks	C++20	7 days ago	Accepted ✔	25.0	View Results
Equal Stacks	C++20	7 days ago	Accepted ✔	25.0	View Results
Equal Stacks	C++20	7 days ago	Wrong Answer ✖	23.33	View Results

Learning outcomes (What I have learnt):

1. I have learnt how to use different functions and library of c++.
2. I have learnt how to deal with real time problems.
3. All 4 for questions helps me to build different logic and concept.
4. Learnt how to implement array and do various types of functions with it.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			