

Experiment 1

Student Name: Anjali Singh
Branch: CSE
Semester: 5th
Subject Code: 21CSP-314

UID: 20BCS9239
Section/Group: 607/A
Date of Performance: 04/09/22

- 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 to understand various problems and find out a better approach to solve a 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.Declare 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;
}

```

HackerRank Prepare > Tutorials > 30 Days of Code > Day 7: Arrays

Objective
Today, we will learn about the Array data structure. Check out the [Tutorial](#) tab for learning materials and an instructional video.

Task
Given an array, A , of N integers, print A 's elements in reverse order as a single line of space-separated numbers.

Example
 $A = [1, 2, 3, 4]$
Print 4 3 2 1. Each integer is separated by one space.

Input Format
The first line contains an integer, N (the size of our array).
The second line contains N space-separated integers that describe array A 's elements.

Constraints
Constraints
• $1 \leq N \leq 1000$
• $1 \leq A[i] \leq 10000$, where $A[i]$ is the i^{th} integer in the array.

Output Format
Print the elements of array A in reverse order as a single line of space-separated numbers.

Sample Input
4
1 4 3 2

Sample Output

30 Days of Code You have earned 30.00 points!
You are now 4 challenges away from the 2nd star for your 30 days of code badge. 20% 3/7

Congratulations
You solved this challenge. Would you like to challenge your friends? [f](#) [t](#) [in](#) [Next Challenge](#)

Test case 0 [Success](#)
Test case 1 [Success](#)
Test case 2 [Success](#)
Test case 3 [Success](#)
Test case 4 [Success](#)
Test case 5 [Success](#)
Test case 6 [Success](#)

Compiler Message
Success

Hidden Test Case
Unlock this testcase for 5 hacks.
[Unlock](#)

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;  
}
```

the sum of its elements.

[1, 2, 3], $1 + 2 + 3 = 6$, so return 6.

function in the editor below. It must return the sum of the array

ing parameter(s):

er, n , denoting the size of the array.

ce-separated integers representing the array's elements.

ments as a single integer.

elements: $1 + 2 + 3 + 4 + 10 + 11 = 31$

Line: 20 Col: 1

Upload Code as File Test against custom input

Run Code Submit Code

Congratulations
You solved this challenge.
Would you like to challenge
your friends?
Next Challenge

Earn a certificate in Problem Solving
Kudos on your progress! Take the
HackerRank Skills Certification test
and enrich your profile.
Get Certified

Test case 0
Test case 1
Test case 2

Compiler Message
Success

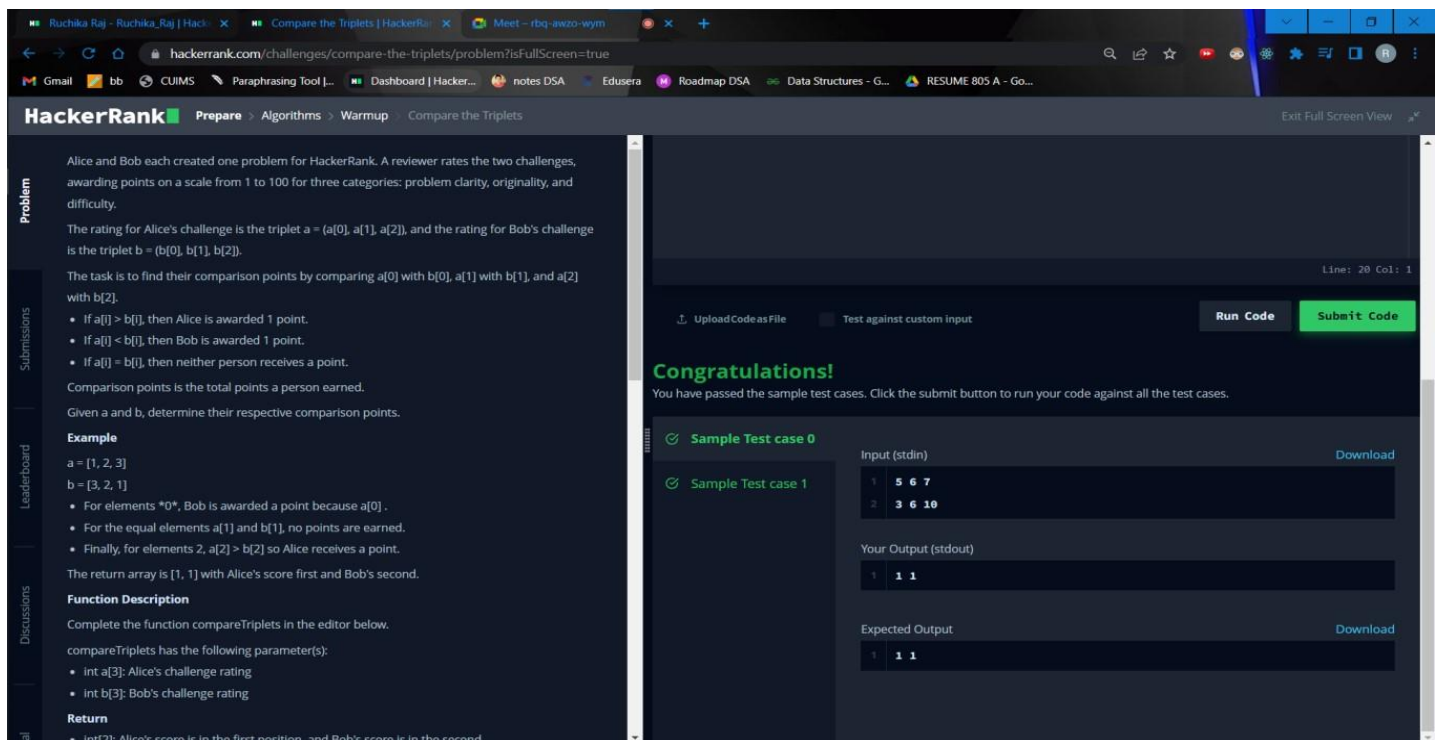
Input (stdin)
6
1 2 3 4 10 11
Download

Expected Output
31
Download

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;
}
```



HackerRank Prepare > Algorithms > Warmup > Compare the Triplets

Problem

Alice and Bob each created one problem for HackerRank. A reviewer rates the two challenges, awarding points on a scale from 1 to 100 for three categories: problem clarity, originality, and difficulty.

The rating for Alice's challenge is the triplet $a = (a[0], a[1], a[2])$, and the rating for Bob's challenge is the triplet $b = (b[0], b[1], b[2])$.

The task is to find their comparison points by comparing $a[0]$ with $b[0]$, $a[1]$ with $b[1]$, and $a[2]$ with $b[2]$.

- If $a[i] > b[i]$, then Alice is awarded 1 point.
- If $a[i] < b[i]$, then Bob is awarded 1 point.
- If $a[i] = b[i]$, then neither person receives a point.

Comparison points is the total points a person earned.

Given a and b , determine their respective comparison points.

Example

$a = [1, 2, 3]$
 $b = [3, 2, 1]$

- For elements $a[0]$, Bob is awarded a point because $a[0] < b[0]$.
- For the equal elements $a[1]$ and $b[1]$, no points are earned.
- Finally, for elements $a[2]$, Alice receives a point.

The return array is $[1, 1]$ with Alice's score first and Bob's second.

Function Description

Complete the function `compareTriplets` in the editor below.

`compareTriplets` has the following parameter(s):

- `int a[3]`: Alice's challenge rating
- `int b[3]`: Bob's challenge rating

Return

- `int[2]`: Alice's score is in the first position, and Bob's score is in the second.

Congratulations!
 You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0

Input (stdin)

```
5 6 7
3 6 10
```

Your Output (stdout)

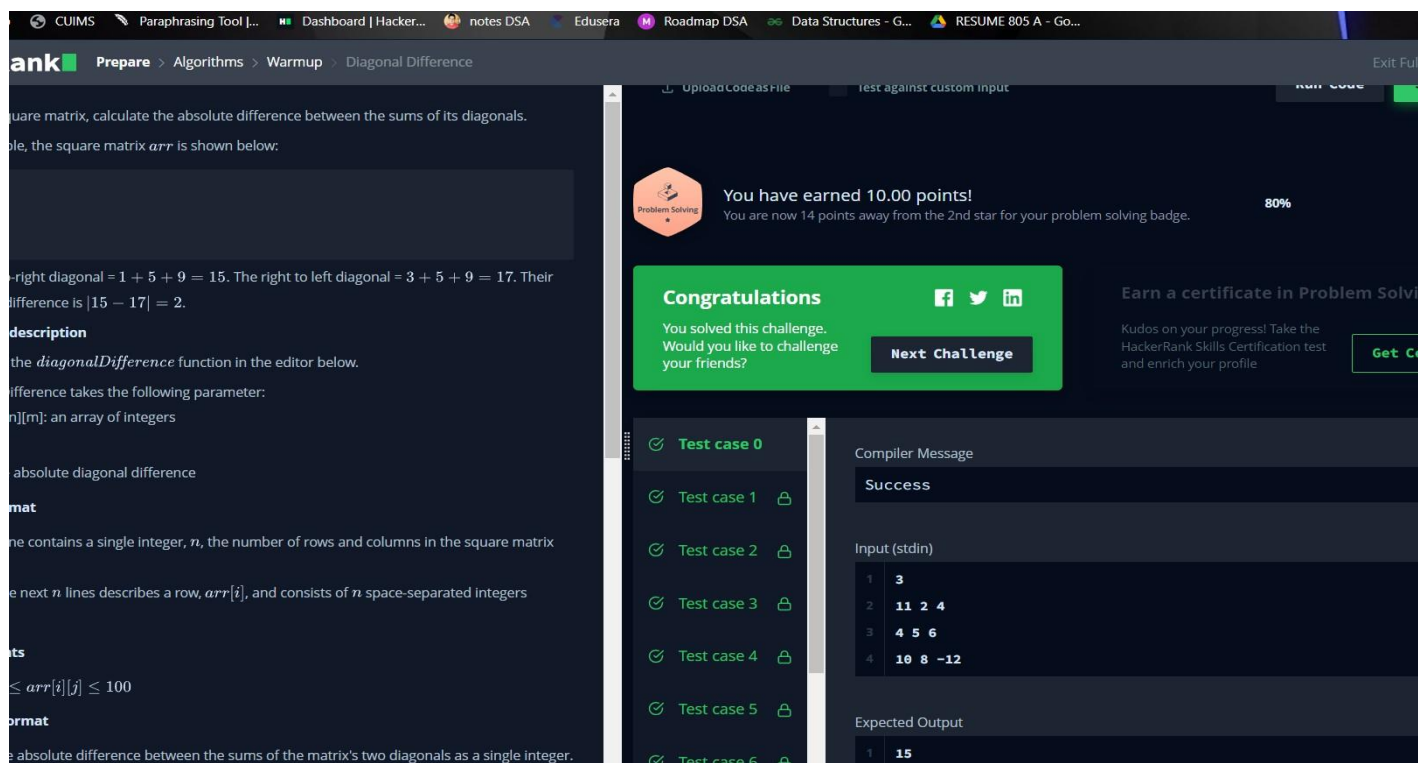
```
1 1
```

Expected Output

```
1 1
```

D. Diagonal Difference

```
#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;
}
```

The screenshot shows the HackerRank interface for the 'Diagonal Difference' problem. The problem description on the left explains that for a square matrix, the absolute difference between the sums of its diagonals must be calculated. It provides an example matrix and the formula for the absolute difference. The solution code is written in Python in the editor. On the right, a green banner celebrates the user for solving the challenge, showing they have earned 10.00 points and are 14 points away from the 2nd star. Below this, a list of test cases (0 to 6) is shown, all of which are passed. The 'Compiler Message' section displays 'Success'. The 'Input (stdin)' section shows a 4x4 matrix, and the 'Expected Output' is 15.

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.			