**Worksheet-1**

**Student Name:-** Pushpraj Roy **UID:-** 20BCS9866

**Branch:-** BE- CSE **Section/Group:-** WM\_617 “A”

**Subjetct Code:-** 20CSP-314 **Semester:-** 5th

**Subject Name:-** Competitive Coding Lab

1. **Aim/Overview of the practical:-**
2. Complete the triplets
3. Diagonal Difference
4. **Task to be done/ Which logistics used:-**
5. Alice and Bob each created one problem for Hacker Rank. 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.

1. Given a square matrix, calculate the absolute difference between the sums of its diagonals.

Complete the function in the editor below.

Diagonal Difference takes the following parameter:

* int arr[n][m]: an array of integers

Return

* int: the absolute diagonal difference

1. **Algorithm/Flowchart:-**
2. **Compare the Triplets:**

* Create a class Solution.
* Input the length of array by using scanner class.
* Compare the elements of an array.
* Print the suitable message.

1. **Diagonal Difference:**

* Create a class Solution.
* Input the length of array by using scanner class.
* Calculate the difference of both the diagonals.
* Print the suitable message.

1. **Steps for experiment/practical/Code:**
2. **Compare the Triplets:**

#include <bits/stdc++.h>

template<typename T> T gcd(T a, T b) {

if(!b) return a;

return gcd(b, a % b);

}

template<typename T> T lcm(T a, T b) {

return a \* b / gcd(a, b);

}

template<typename T> void chmin(T& a, T b) { a = (a > b) ? b : a; }

template<typename T> void chmax(T& a, T b) { a = (a < b) ? b : a; }

int in() { int x; scanf("%d", &x); return x; }

using namespace std;

#ifdef ONLINE\_JUDGE

#define debug(args...)

#else

#define debug(args...) fprintf(stderr,args)

#endif

typedef long long Int;

typedef unsigned long long uInt;

typedef unsigned uint;

int A[5], B[5];

int main(void) {

int ia = 0;

int ib = 0;

for (int i = 0; i < 3; i++) {

cin >> A[i];

}

for (int i = 0; i < 3; i++) {

cin >> B[i];

if (A[i] < B[i]) {

ib += 1;

} else if (A[i] > B[i]) {

ia += 1;

}

}

cout << ia << " " << ib << "\n";

return 0;

}

1. **Diagonal Difference:**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

using namespace std;

int main() {

int N;

cin >> N;

int i, j;

int sumdiag1 = 0;

int sumdiag2 = 0;

for(i = 0; i < N; i++){

for(j = 0; j< N; j++)

{

int no;

cin >> no;

if(i == j)

sumdiag1 += no;

if(i+j == N-1)

sumdiag2 += no;

}

}

cout << abs(sumdiag1 - sumdiag2);

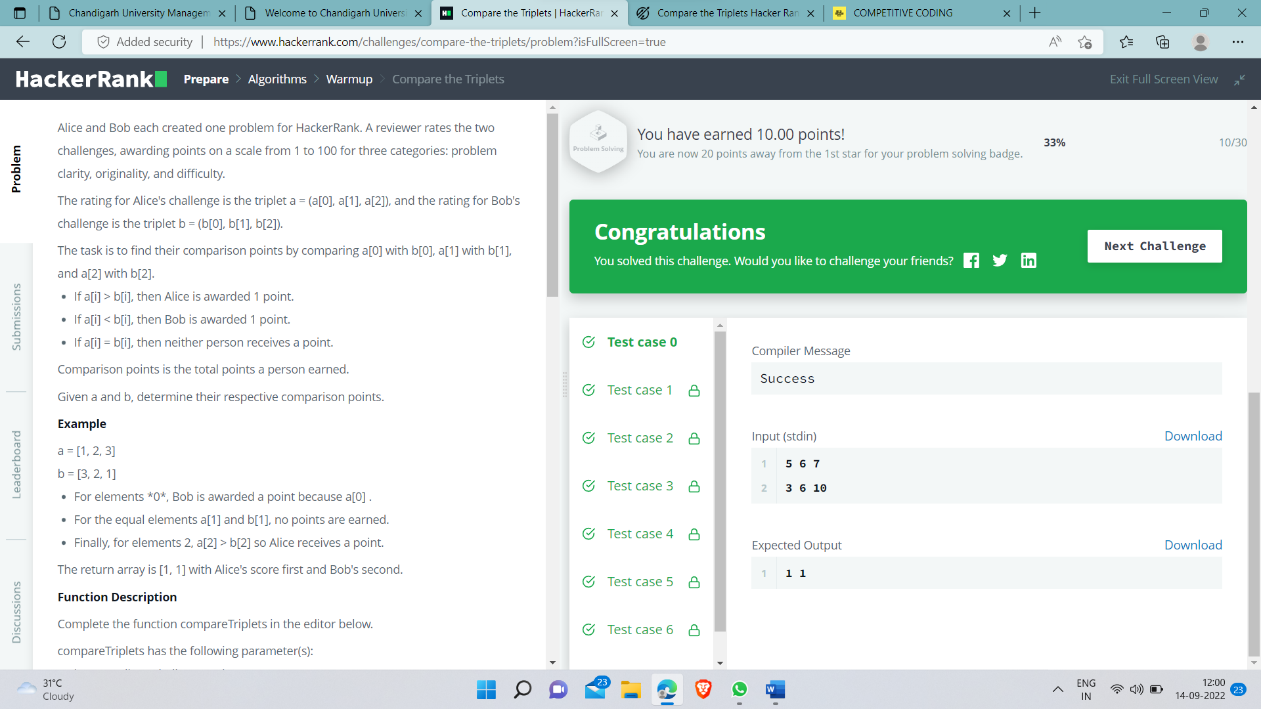
return 0;

}

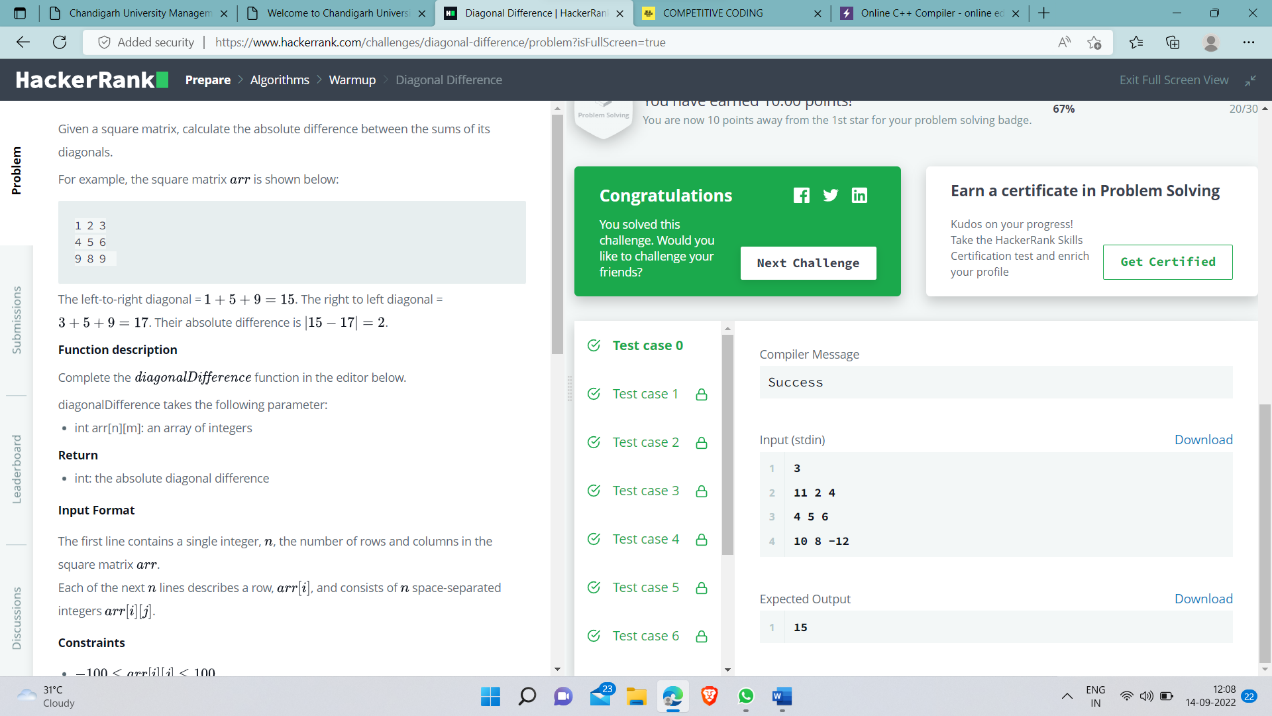
1. **Observations/Discussions/ Complexity Analysis:**

* Calculate the sum of the elements of arrays.
* Compare the elements of the arrays.
* Difference of diagonals of an array.

1. **Result/Output/Writing Summary:**
2. **Compare the Triplets:**

****

1. **Diagonal Difference:**

****

1. **Learning outcomes (What I have learnt):**

* Learn how to calculate the sum of elements of array in c++.
* Learn how to compare the elements in c++.
* Learn how to calculate the difference of diagonals in c++ .

**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. |  |  |  |
|  |  |  |  |