## CHANDIGARH UNIVERSITY

## UNIVERSITY INSTITUTE OF NGINEERING

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



|  |  |
| --- | --- |
| **Submitted By: Submitted To:**  Vivek Kumar(21BCS8129) Mamta Punia(E12337) | |
| **Subject Name** | Competitive Coding - I |
| **Subject Code** | 20CSP-314 |
| **Branch** | Computer Science and Engineering |
| **Semester** | 5th |

**Experiment - 1**

**Student Name: Vivek Kumar UID: 21BCS8129**

**Branch: BE-CSE(LEET) Section/Group: WM-20BCS-616/A**

**Semester: 5th Date of Performance: 12/08/2022**

**Subject Name: Competitive coding - I Subject Code: 20CSP-314**

**1. Aim/Overview of the practical:**

**I.** Given an array of integers, find the sum of its elements.

For example, if the array arr=[1,2,3], 1+2+3=6 , so return 6.

**II.** 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.

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

Make the Changes in the present code and find the Actual output of the given question.

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

**4. Steps for experiment/practical/Code:**

**I. Simple Array Sum:**

import java.io.\*;

import java.util.\*;

public class Solution {

public static int simpleArraySum(int n, int[] ar) {

// Write your code here

int sum=0;

for(int i=0;i<ar.length;i++){

sum=sum+ar[i];

}

return sum;

}

public static void main(String[] args) throws IOException {

Scanner in = new Scanner(System.in);

int n = in.nextInt();

int[] arr = new int[n];

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

arr[i] = in.nextInt();

}

in.close();

int sum=simpleArraySum(n, arr);

System.out.print(sum);

}

}

**II. Compare the Triplets:**

import java.io.\*;

import java.math.\*;

import java.security.\*;

import java.text.\*;

import java.util.\*;

import java.util.concurrent.\*;

import java.util.regex.\*;

class Result {

/\*

\* Complete the 'compareTriplets' function below.

\*

\* The function is expected to return an INTEGER\_ARRAY.

\* The function accepts following parameters:

\* 1. INTEGER\_ARRAY a

\* 2. INTEGER\_ARRAY b

\*/

public static List<Integer> compareTriplets(List<Integer> a, List<Integer> b) {

// Write your code here

int alice = 0;

int bob = 0;

List<Integer> answer = new ArrayList<>();

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

if (a.get(i) > b.get(i)) alice++;

if (a.get(i) < b.get(i)) bob++;

}

answer.add(0,alice);

answer.add(1,bob);

return answer;

}

}

public class Solution {

public static void main(String[] args) throws IOException {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT\_PATH")));

String[] aTemp = bufferedReader.readLine().replaceAll("\\s+$", "").split(" ");

List<Integer> a = new ArrayList<>();

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

int aItem = Integer.parseInt(aTemp[i]);

a.add(aItem);

}

String[] bTemp = bufferedReader.readLine().replaceAll("\\s+$", "").split(" ");

List<Integer> b = new ArrayList<>();

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

int bItem = Integer.parseInt(bTemp[i]);

b.add(bItem);

}

List<Integer> result = Result.compareTriplets(a, b);

for (int i = 0; i < result.size(); i++) {

bufferedWriter.write(String.valueOf(result.get(i)));

if (i != result.size() - 1) {

bufferedWriter.write(" ");

}

}

bufferedWriter.newLine();

bufferedReader.close();

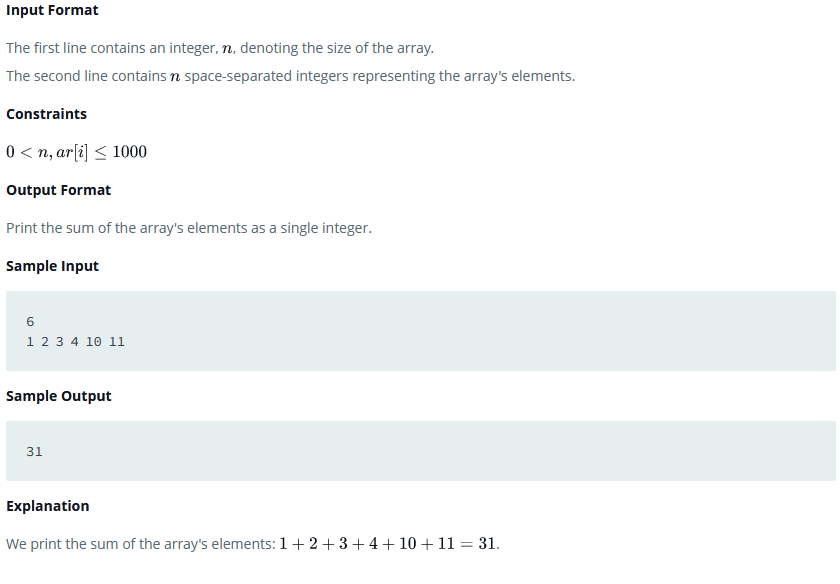
bufferedWriter.close();

}

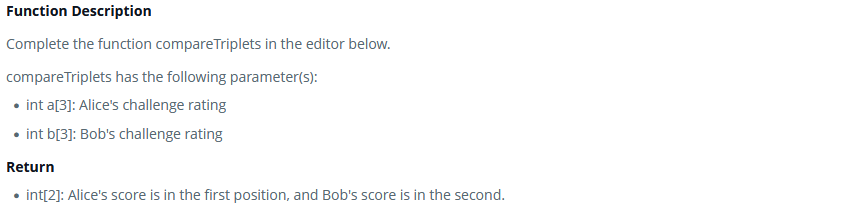
}

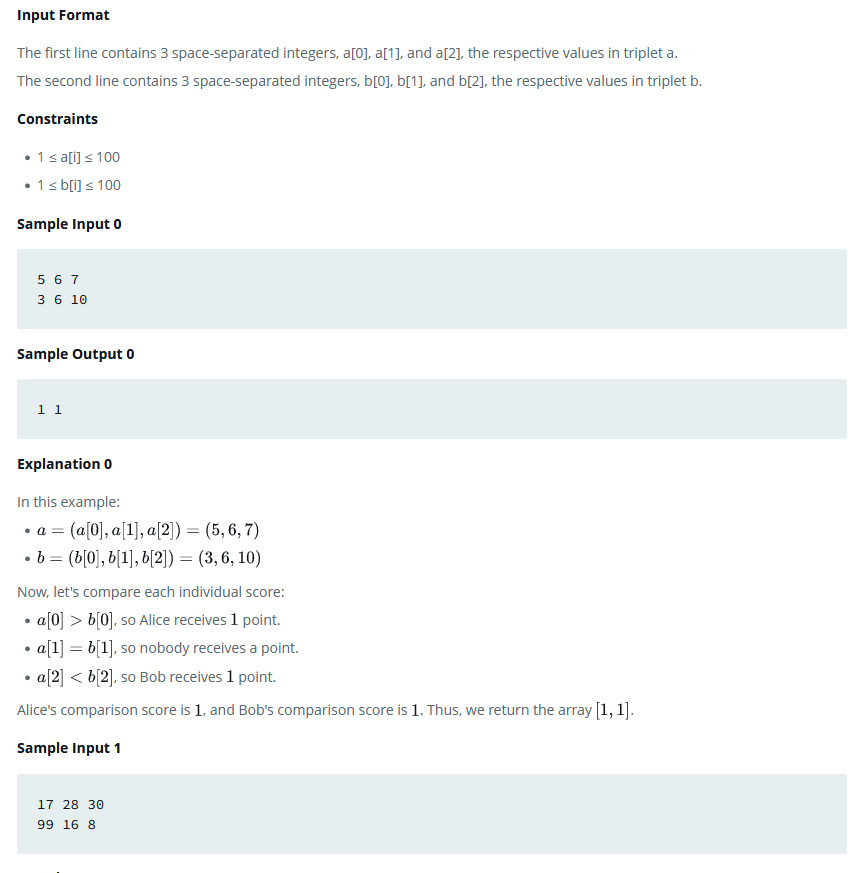
**5. Observations/Discussions/ Complexity Analysis:**

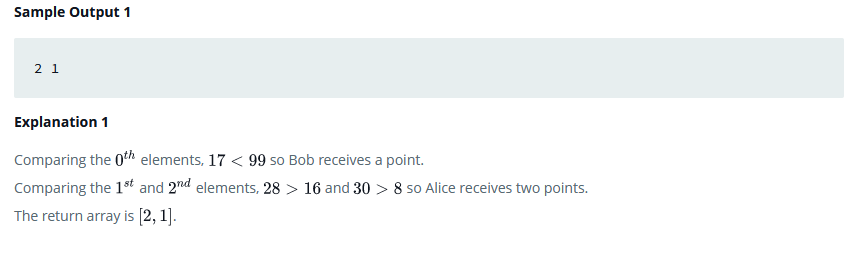
**I. Simple Array Sum:**



**II. Compare the Triplets:**

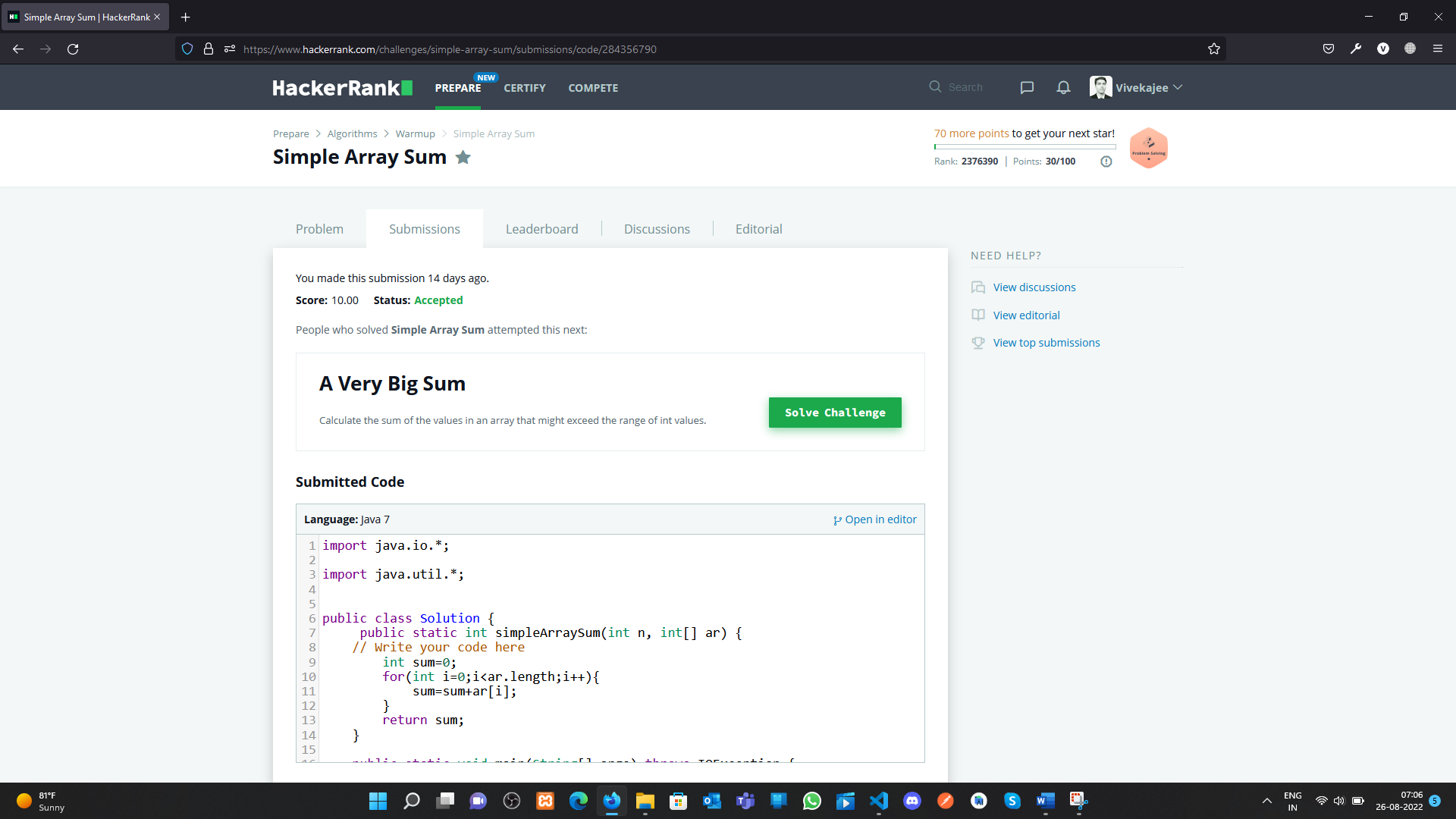
****

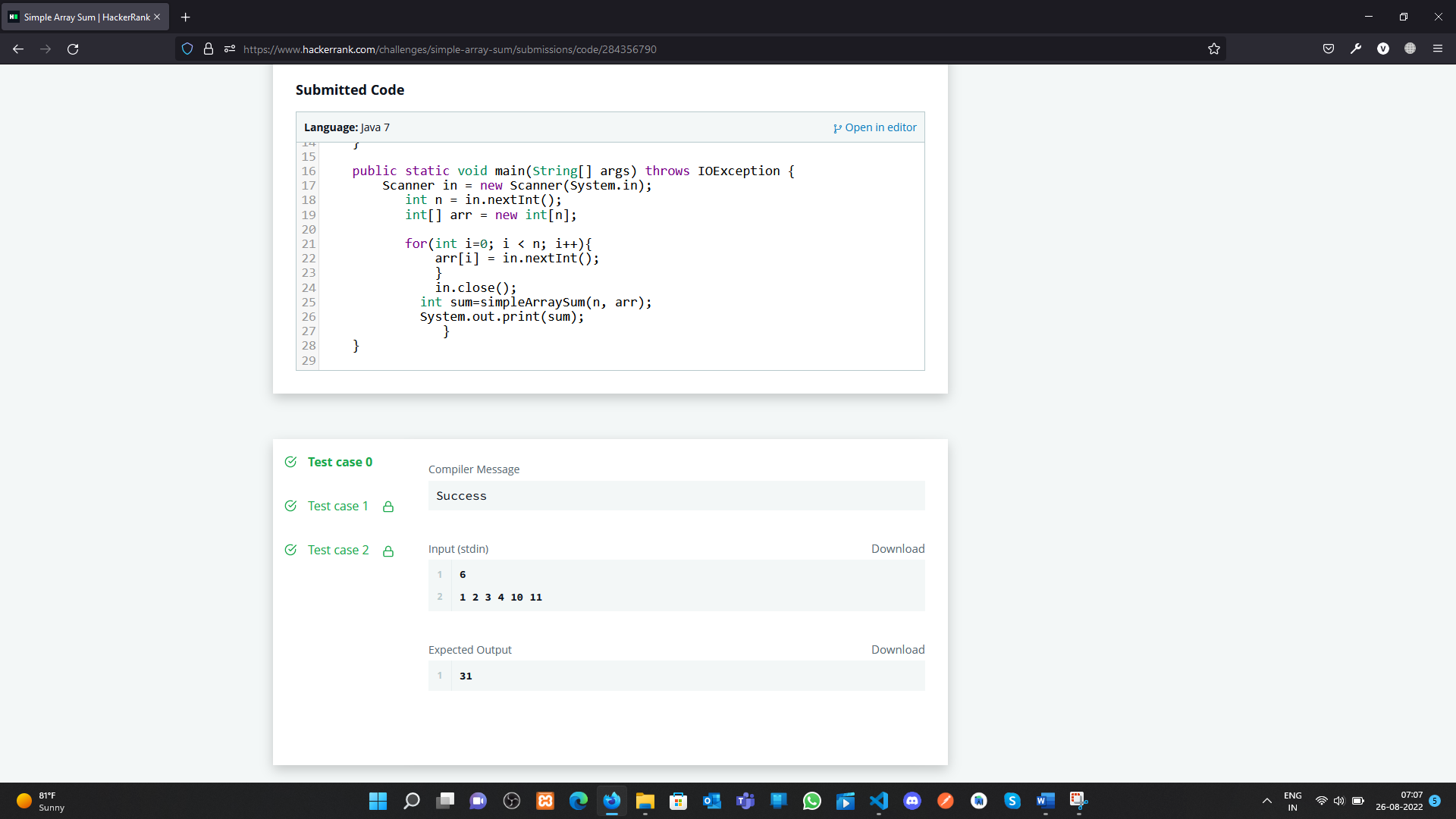
****

****

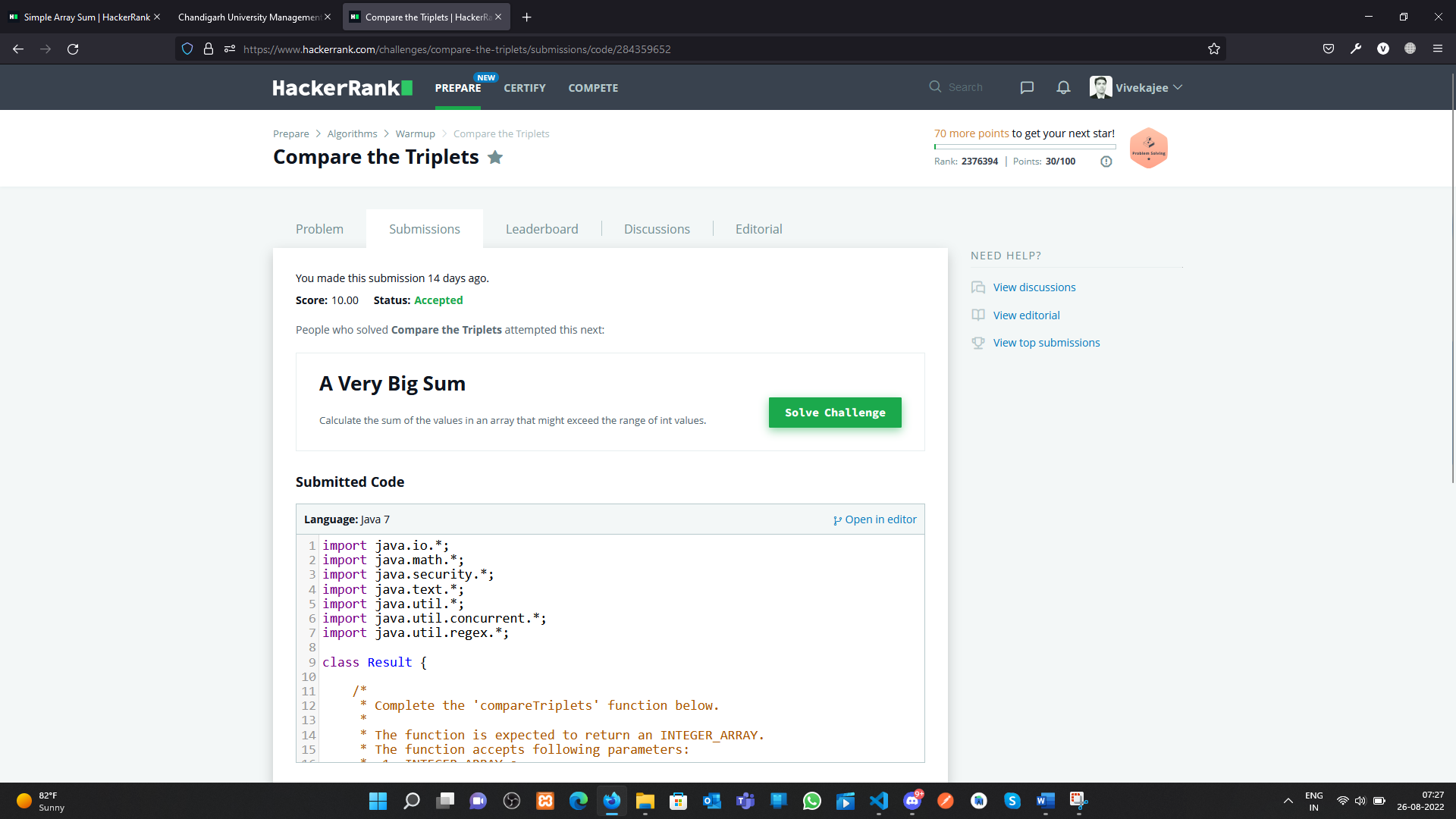
**6. Result/Output/Writing Summary:**

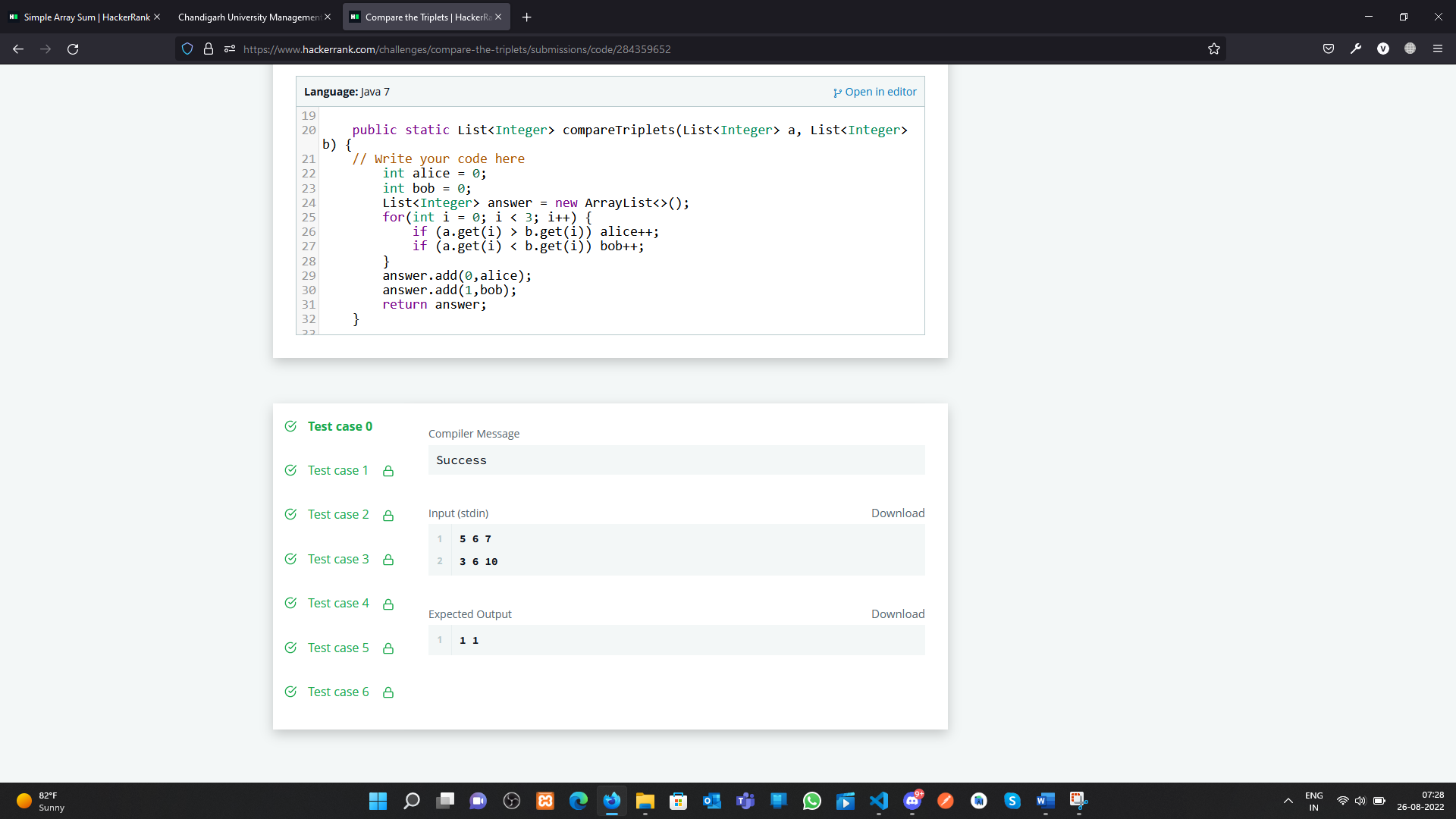
**I. Simple Array Sum:**





**II. Compare the Triplets:**





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

**1.** Array concept in Java

**2.** Sum of the all-item present in an Array

**3.** Compare the triplets and show the results.

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