## CHANDIGARH UNIVERSITY

## UNIVERSITY INSTITUTE OF NGINEERING

## **DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



|  |  |
| --- | --- |
| **Submitted By: Kanishk Soni Submitted To:Mr.Syed Abdul Basit sir** | |
| **Subject Name** | Competitive Coding |
| **Subject Code** | 20CSP-314 |
| **Branch** | BE.CSE. |
| **Semester** | 5th |

**INDEX**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ex. No** | **List of Experiments** | **Conduct (MM: 12)** | **Viva**  **(MM: 10)** | **Record (MM: 8)** | **Total**  **(MM: 30)** | **Remarks/Signature** |
| 1.1 | Array |  |  |  |  |  |
| 1.2 |  |  |  |  |  |  |
| 1.3 |  |  |  |  |  |  |
| 1.4 |  |  |  |  |  |  |
| 2.1 |  |  |  |  |  |  |
| 2.2 |  |  |  |  |  |  |
| 2.3 |  |  |  |  |  |  |
| 2.4 |  |  |  |  |  |  |
| 3.1 |  |  |  |  |  |  |
| 3.2 |  |  |  |  |  |  |
| 3.3 |  |  |  |  |  |  |

**Experiment-1**

**Student Name:** Kanishk Soni **UID:** 20bcs9398

**Branch:** CSE **Section/Group:** 20BCS\_WM\_707 B

**Semester:** 5 **Date of Performance:** 28/08/2022

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

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

**2. Task to be done/ Which logistics used:** The tasks involve finding the sum of elements of an array, comparing triplets, reversing the array and finding the difference of sum of diagonal elements of an array.

**3. Algorithm/Flowchart:**

**A)** **Simple array sum**

Step 1. Read the size of array,

Step 2. Read the array from loop running from 0 to size-1,

Step 3. Again running a loop from 0 to size-1 to store every elements sum in variable s,

Step 4. Display the variable s.

1. **Reverse array**

Step 1. Read the size of array,

Step 2. Read the array from loop running from 0 to size-1,

Step 3. Now run the loop from size-1 to 0 and display the elements.

1. **Compare the Triplets**

Step 1. Read the array A from loop running from 0 to 2,

Step 2. Read the array B from loop running from 0 to 2,

Step 3. Define Ap and Bp variable with 0,

Step 4. Run a loop from 0 to 2 as i and check if value of array A at index i is greater then value of array B at index i then go to Step 5. otherwise check if value of array A at index i is less then value of array B at index i then go to Step 6. after completing the loop go to Step 7.,

Step 5. Increment Ap by 1 then go to Step 4.,

Step 6. Increment Bp by 1 then go to Step 4.,

Step 7. Display Ap and Bp space separated.

1. **Diagonal Difference**

Step 1. Take size of array and array(2D) as n and ar as parameters,

Step 2. define d1 and d2 variable with 0,

Step 3. Run a loop from 0 to n-1 as i,

Step 4. Increment d1 by value of array ar at index [i][i],

Step 5. Increment d2 by value of array ar at index [i][n-1-i],

Step 6. End of loop,

Step 7. return abscissa of d1-d2.

**4. Code & Output:**

**a) Simple array sum:** https://www.hackerrank.com/challenges/simple-array-sum/problem

Code:

#include <bits/stdc++.h>

using namespace std;

int main(){

int n, sum=0, i; cin>>n;

int a[n];

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

cin>>a[i];

sum += a[i];

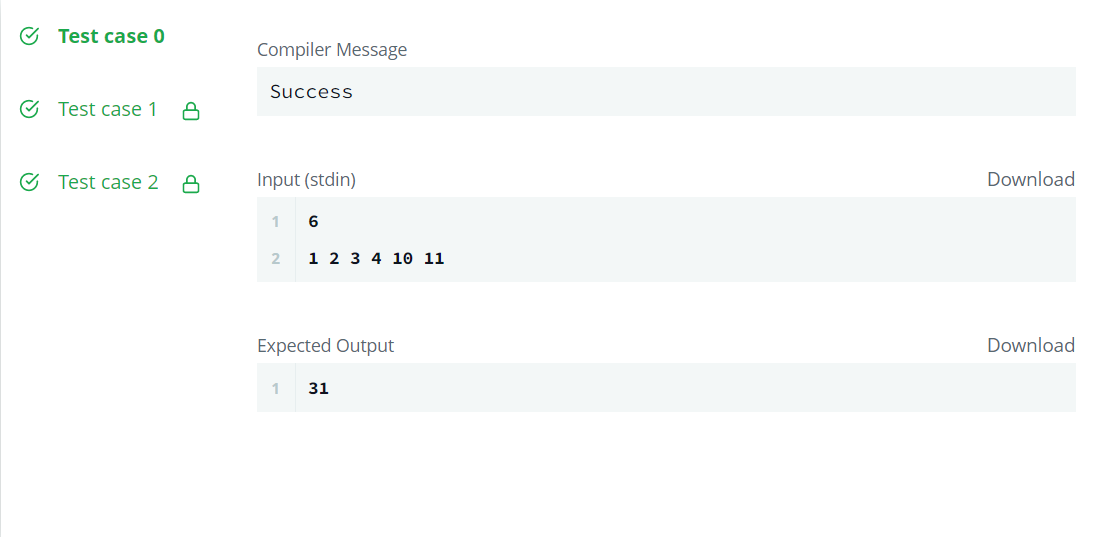
}

cout<<sum;

return 0;

}

Output:



**b) Reverse array:** <https://www.hackerrank.com/challenges/30-arrays/copy-from/284191564>

Code:

#include <bits/stdc++.h>

using namespace std;

int main(){

int n, i; cin>>n;

int a[n];

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

cin>>a[i];

}

for(i=n-1; i>=0; --i){

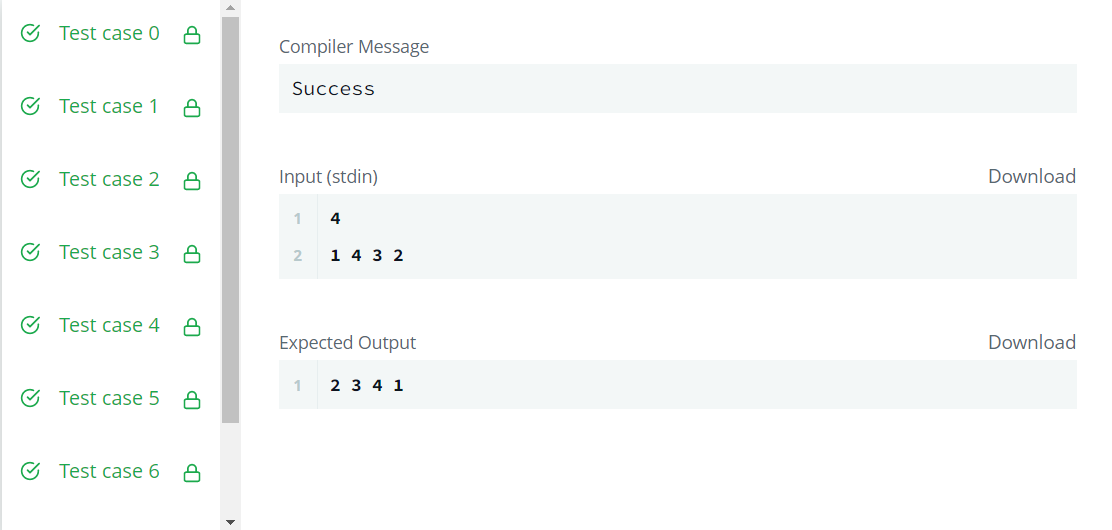
cout<<a[i]<<" ";

}

return 0;

}

Output:



**c) Compare the triplets:** <https://www.hackerrank.com/challenges/compare-the-triplets/problem>

Code:

#include <bits/stdc++.h>

using namespace std;

int main() {

int i, a[3], b[3], ap=0, bp=0;

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

cin>>a[i];

}

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

cin>>b[i];

if(a[i] > b[i]) {

ap++;

} else if(a[i] < b[i]) {

bp++;

}

}

cout<<ap<<" "<<bp;

return 0;

}

Output:



**d) Diagonal difference:** https://www.hackerrank.com/challenges/diagonal-difference/problem

Code:

#include <bits/stdc++.h>

using namespace std;

#define X 1000

int abs(int x){

if(x<0){

return (-1) \* x;

}

return x;

}

int diagDiff(int n, int ar[X][X]){

int d1=0, d2=0, i;

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

d1 += ar[i][i];

d2 += ar[i][n-1-i];

}

return abs(d1 - d2);

}

int main() {

int n, i, j, d1=0, d2=0; cin>>n;

int a[X][X];

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

for(j=0; j<n; j++){

cin>>a[i][j];

}

}

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

d1 += a[i][i];

d2 += a[i][n-1-i];

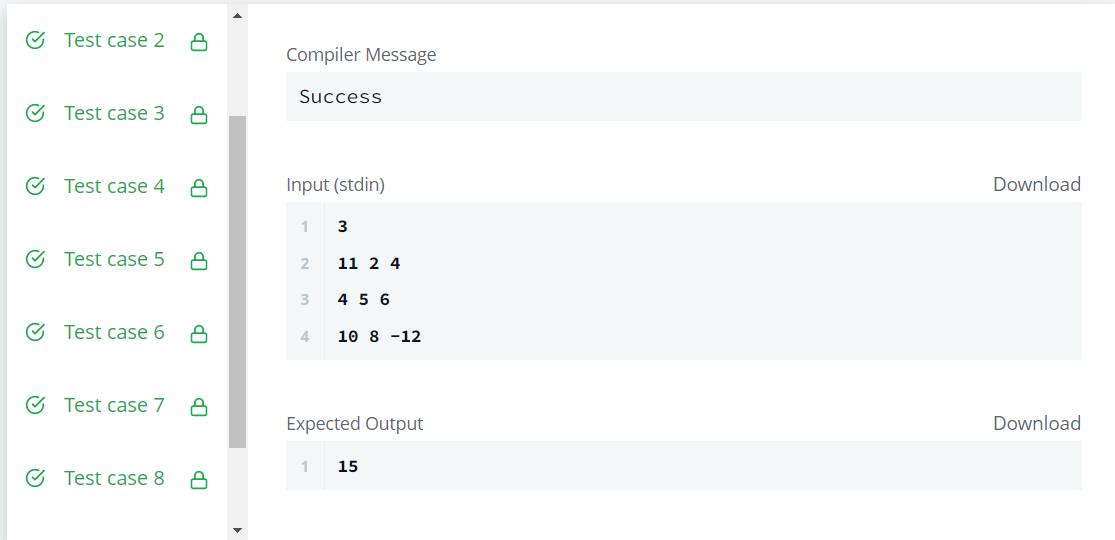
}

cout<<diagDiff(n, a);

return 0;

}

Output:



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

**1.** Understood the concept of arrays.

**2.** Solved questions based on arrays to get a better understanding of the data structure.

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