ABHISHEK SHARMA

SECTION : I

ROLL NO.: 1

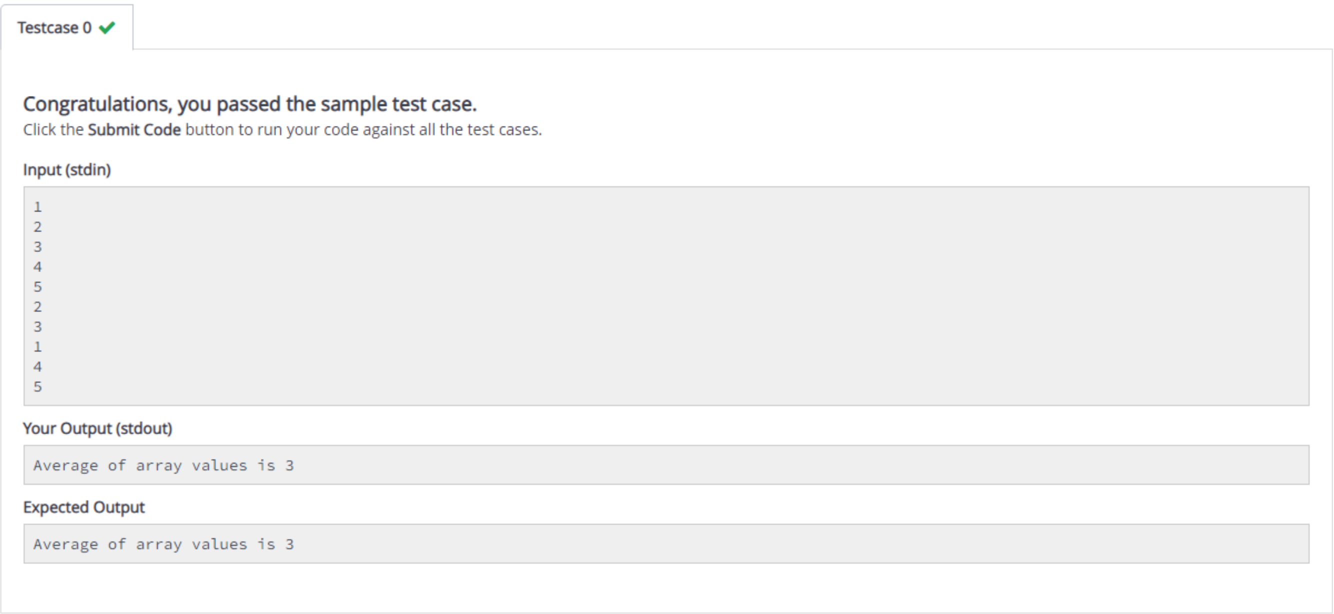
HACKERRANK ID : abhishek00012

1. Write down a program to find out the average of 10 numbers using Array.

Code :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
int main(void) {
    int temp,sum=0;
    for(int i=0;i<10;i++)
    {       scanf("%d",&temp);
            sum+=temp;
    }
    printf("Average of array values is %d",sum/10);
    return 0;
}
```

Output :



Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
1
2
3
4
5
2
3
1
4
5
```

Your Output (stdout)

```
Average of array values is 3
```

Expected Output

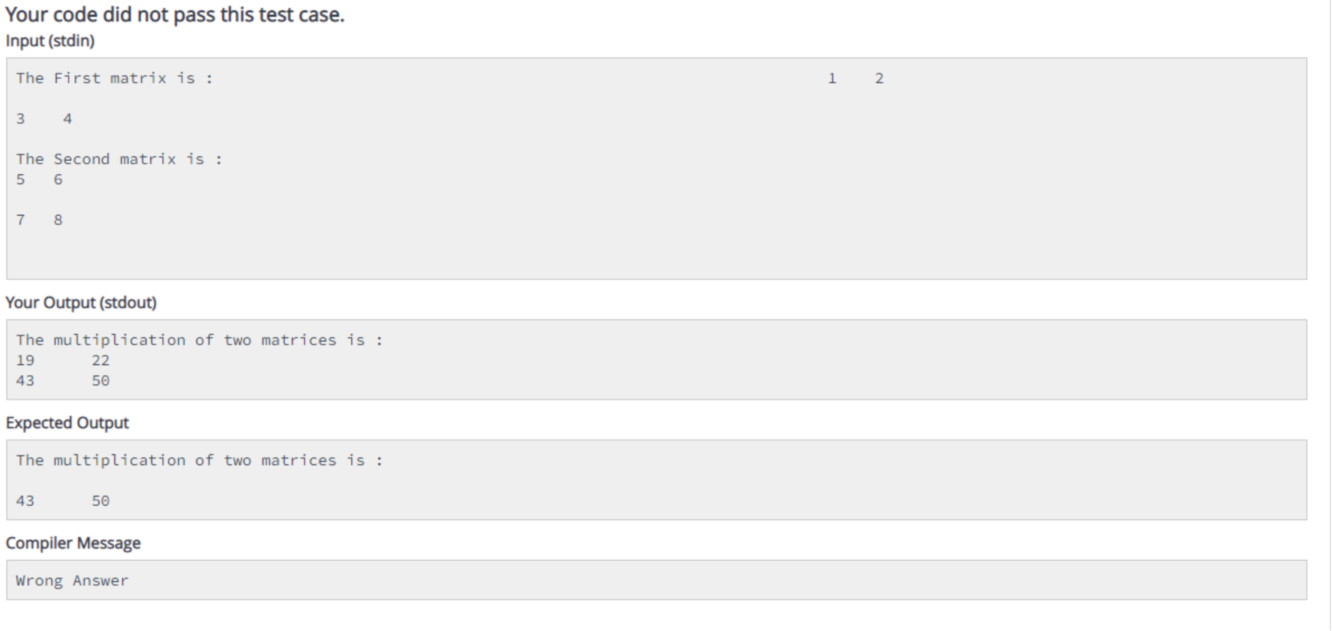
```
Average of array values is 3
```

2. You are given two matrices of dimensions N x N, filled with integers. Your task is to print the product of these matrices.

Code :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
int main()
{
  int i,j, k, sum = 0,a=0;
  int first[2][2], second[2][2], multiply[2][2];
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            a++;
            first[i][j]=a;
        }
    }
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
             a++;
            second[i][j]=a;
        }
    }
    for (i = 0; i < 2; i++) {
      for (j = 0; j < 2; j++) {
        for (k = 0; k < 2; k++) {
          sum = sum + first[i][k]*second[k][j];
        }

        multiply[i][j] = sum;
        sum = 0;
      }
    }
    printf("The multiplication of two matrices is :\n");
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            printf("%d      ",multiply[i][j]);
        }
        printf("\n");
    }
}
```

Output :



Your code did not pass this test case.

Input (stdin)

```
The First matrix is :                                                                         1    2

3    4

The Second matrix is :
5   6

7   8
```

Your Output (stdout)

```
The multiplication of two matrices is :
19      22
43      50
```

Expected Output

```
The multiplication of two matrices is :

43      50
```

Compiler Message

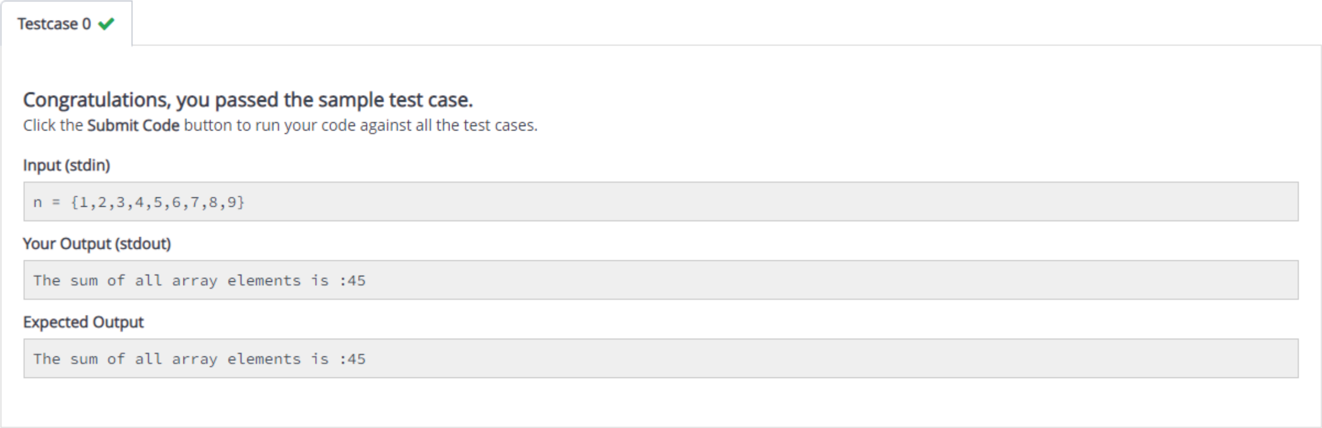
```
Wrong Answer
```

3. Write down a program to calculate the sum of array elements by passing to a function.

Code :

```
#include <stdio.h>
int main()
{
    int a[9]={1,2,3,4,5,6,7,8,9},i,sum=0;
    for(i=0; i<9; i++)
    {
        scanf("%d",&a[i]);
    }
    for(i=0; i<9; i++)
    {
        sum+=a[i];
    }
     printf("The sum of all array elements is :%d",sum);

    return 0;
}
```

Output :



Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
n = {1,2,3,4,5,6,7,8,9}
```

Your Output (stdout)

```
The sum of all array elements is :45
```

Expected Output

```
The sum of all array elements is :45
```

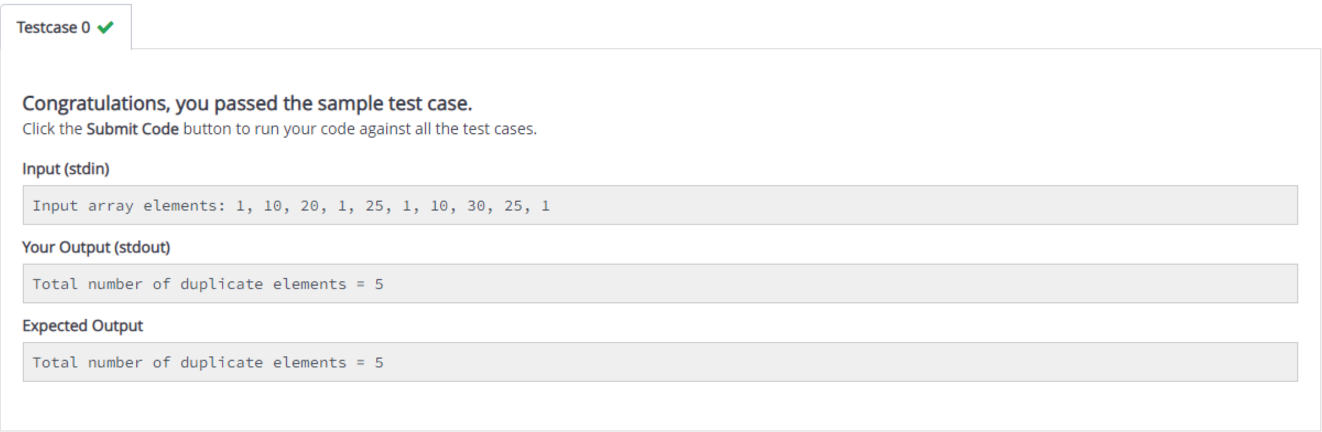
4. Write a program in c to count a total number of duplicate elements in the array using function.

Code :

```
#include <stdio.h>

int main()
{
    int arr[10]={1, 10, 20, 1, 25, 1, 10, 30, 25, 1} , i, j, Size=10 , Count = 0;



    for (i = 0; i < Size; i++)
    {
        for(j = i + 1; j < Size; j++)
        {
            if(arr[i] == arr[j])
            {
                Count++;
                break;
```

```
          }
       }
    }

     printf("Total number of duplicate elements = %d ", Count);

     return 0;
}
```

Output :



Testcase 0 ✓

Congratulations, you passed the sample test case.
Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
Input array elements: 1, 10, 20, 1, 25, 1, 10, 30, 25, 1
```

Your Output (stdout)

```
Total number of duplicate elements = 5
```

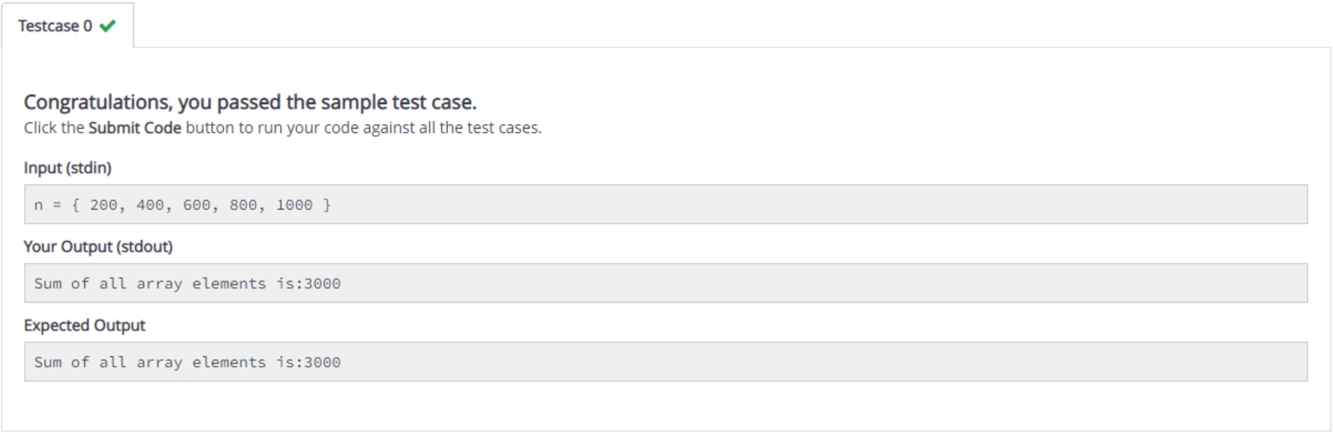
Expected Output

```
Total number of duplicate elements = 5
```

5. C Program to calculate sum of all elements of an array using pointers as arguments.
Code :

```
#include <stdio.h>

int main()
{
    static int array[5] = { 200, 400, 600, 800, 1000 };
    int sum;

      int addnum(int *ptr);
         sum = addnum(array);

        printf("Sum of all array elements is:%d\n", sum);
    return 0;
}

int addnum(int *ptr)
{
    int index, total = 0;
    for (index = 0; index < 5; index++)
        {
        total += *(ptr + index);
    }
    return(total);
 }
```

Output :



Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
n = { 200, 400, 600, 800, 1000 }
```

Your Output (stdout)

```
Sum of all array elements is:3000
```

Expected Output

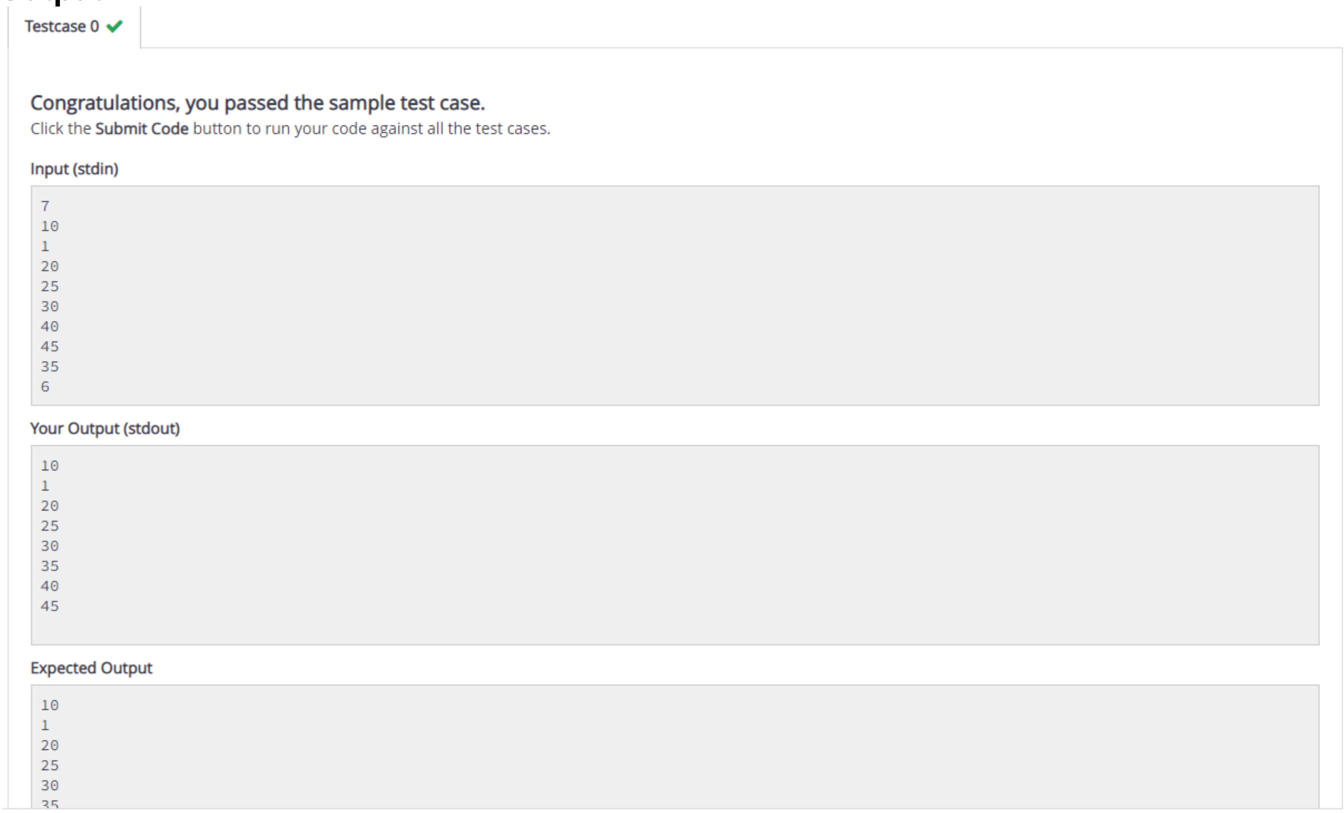
```
Sum of all array elements is:3000
```

6. Write a program in C to insert an element in an array.

Code :

```
#include <stdio.h>

int main()
{
    int arr[100] = { 0 };
    int i, x, pos, n = 10;

    // initial array of size 10
    arr[0]=7;
    arr[1]=10;
    arr[2]=1;
    arr[3]=20;
    arr[4]=25;
    arr[5]=30;
    arr[6]=40;
    arr[7]=45;
    arr[8]=35;
    arr[9]=6;



    // element to be inserted
    x = 35;

    // position at which element
    // is to be inserted
    pos = 7;

    // increase the size by 1
    n++;

    // shift elements forward
    for (i = n; i >= pos; i--)
        arr[i] = arr[i - 1];

    // insert x at pos
    arr[pos - 1] = x;

    // print the updated array
    for (i = 1; i < n-2; i++)
        printf("%d\n", arr[i]);
    printf("\n");

    return 0;
}
```

Output :



Testcase 0 ✓

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
7
10
1
20
25
30
40
45
35
6
```

Your Output (stdout)

```
10
1
20
25
30
35
40
45
```

Expected Output

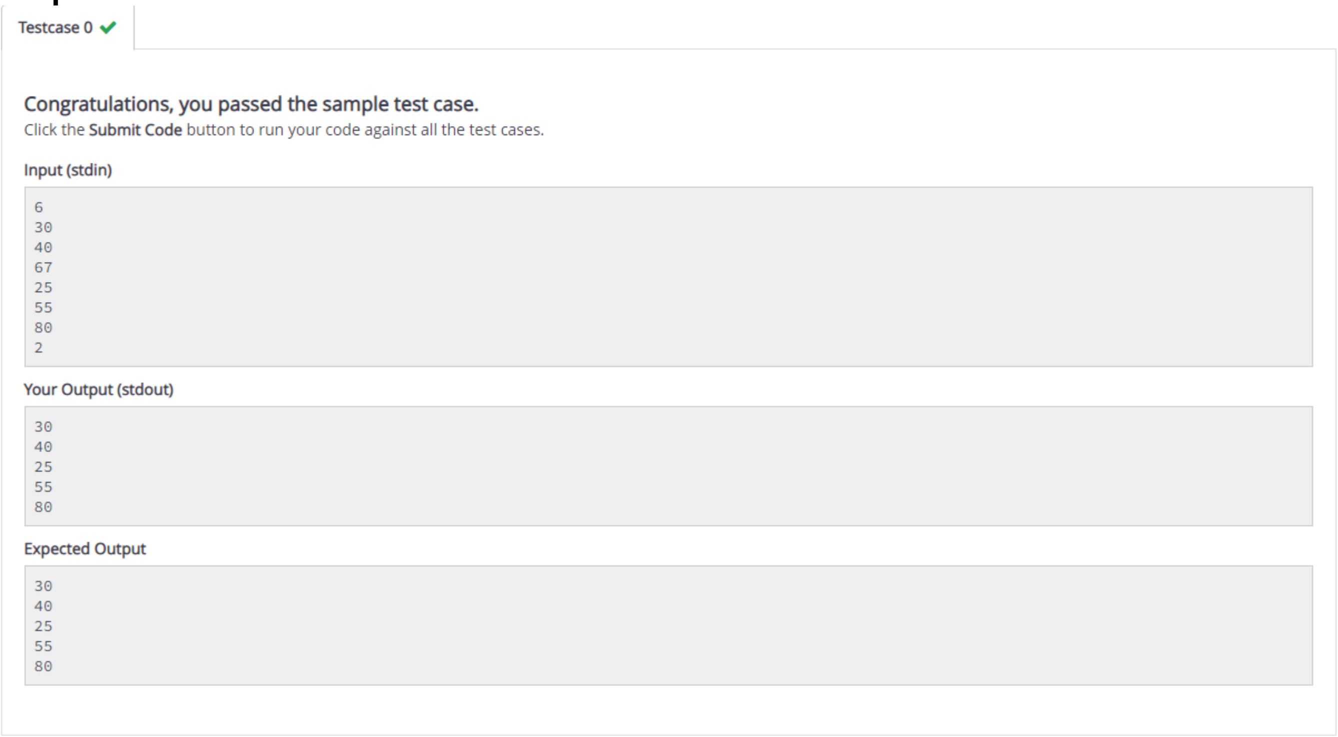
```
10
1
20
25
30
35
```

7. Write a program in C to deleting an element in an array.

Code :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int n, m,i;
    scanf("%d ",&n);
    int a[n];
    for(i=0;i<n;i++)
        scanf("%d ",&a[i]);
    scanf("%d ",&m);
    for(i=m;i<n-1;i++)
    {
            a[i]=a[i+1];
    }
    n--;
    for(i=0;i<n;i++)
        printf("%d\n", a[i]);
    return 0;
}
```

Output :



Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
6
30
40
67
25
55
80
2
```

Your Output (stdout)

```
30
40
25
55
80
```

Expected Output

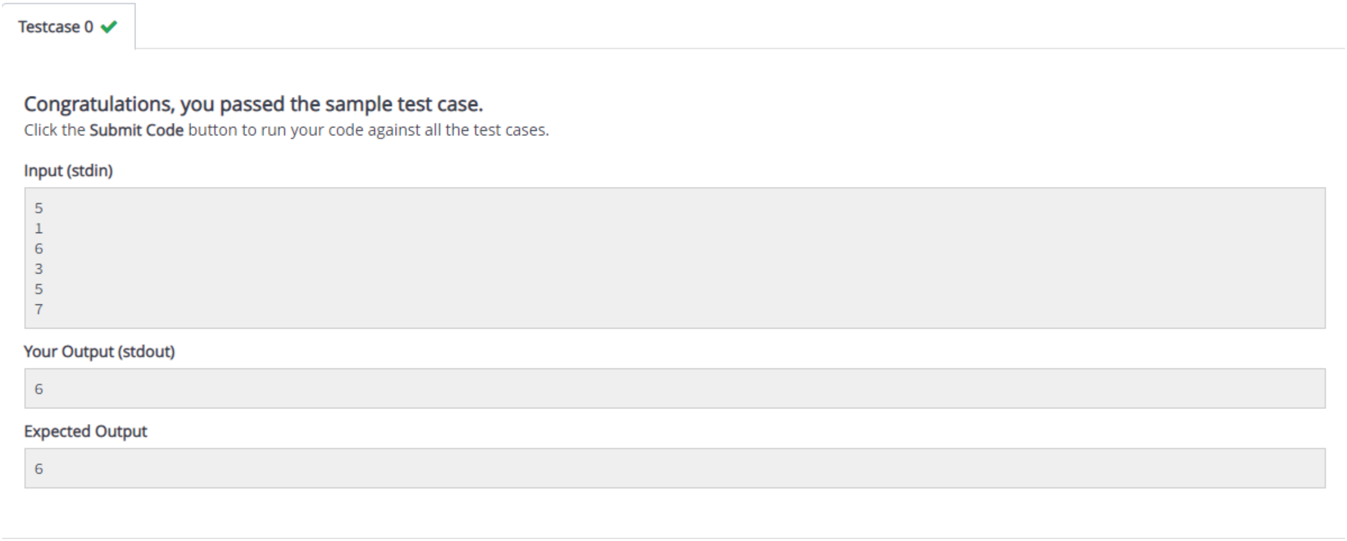
```
30
40
25
55
80
```

9. Write a program in C to find the second highest number in an array.

Code :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int n, i,j,max;
    scanf("%d",&n);
    int a[n];
    for(i=0;i<n;i++)
        scanf("%d\n ",&a[i]);
    for(i=0;i<n;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(a[i]>a[j])
            {
                max=a[i];
                a[j]=a[i];
                a[j]=max;
            }
        }
    }
    printf("%d ",a[n-2]);

    return 0;
}
```

Output :



Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
5
1
6
3
5
7
```

Your Output (stdout)

```
6
```

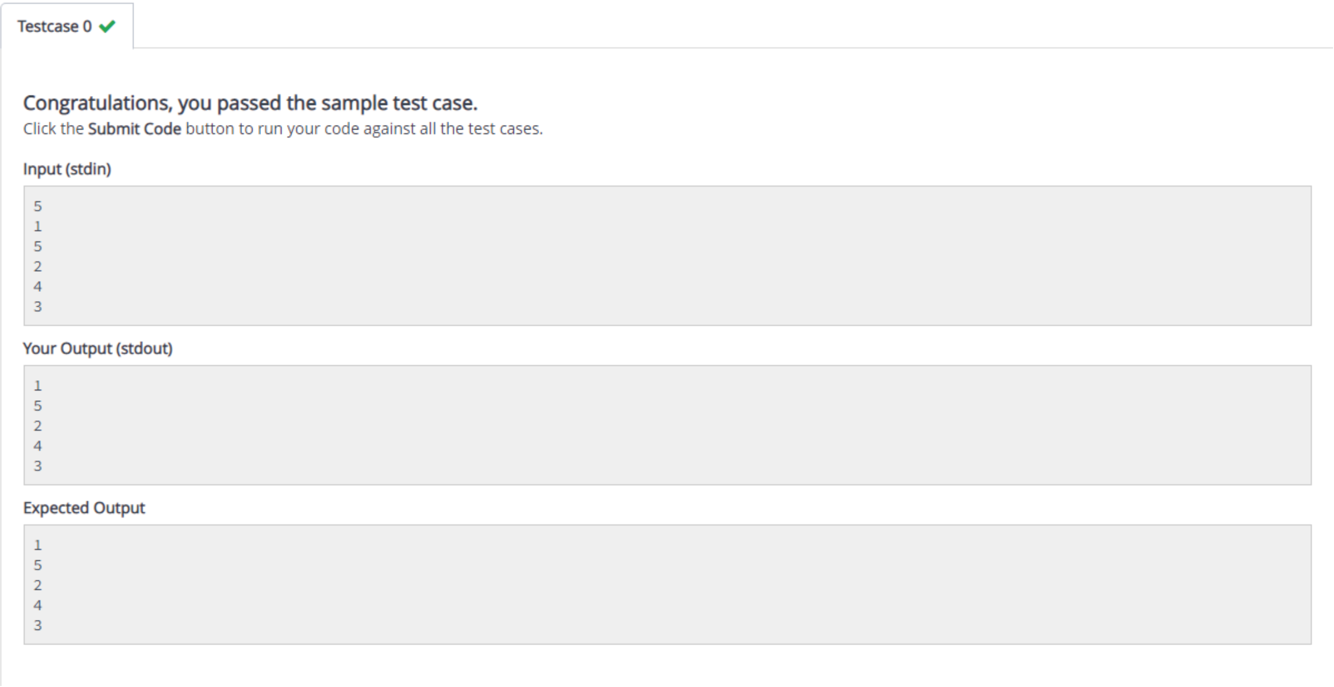
Expected Output

```
6
```

10. C program to copy an array into another using a pointer.
Code :

```
#include <stdio.h>

#define MAX_SIZE 100 // Maximum array size


/* Function declaration to print array */
void printArray(int arr[], int size);


int main()
{
    int source_arr[MAX_SIZE], dest_arr[MAX_SIZE];
    int size, i;

    int *source_ptr = source_arr;   // Pointer to source_arr
    int *dest_ptr   = dest_arr;     // Pointer to dest_arr

    int *end_ptr;


    /*
     * Input size and elements in source array
     */

    scanf("%d", &size);

    for (i = 0; i < size; i++)
    {
        scanf("%d", (source_ptr + i));
    }


    // Pointer to last element of source_arr
    end_ptr = &source_arr[size - 1];
    /*
     * Run loop till source_ptr exists in source_arr
     * memory range.
     */
    while(source_ptr <= end_ptr)
    {
        *dest_ptr = *source_ptr;

        // Increment source_ptr and dest_ptr
        source_ptr++;
        dest_ptr++;
    }




    printArray(dest_arr, size);


    return 0;
}


/**
 * Function to print array elements.
 *
 * @arr     Integer array to print.
 * @size    Size of array.
 */
void printArray(int *arr, int size)
{
    int i;

    for (i = 0; i < size; i++)
    {
        printf("%d\n", *(arr + i));
    }
}
```

Output :



Congratulations, you passed the sample test case.
Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
5
1
5
2
4
3
```

Your Output (stdout)

```
1
5
2
4
3
```

Expected Output

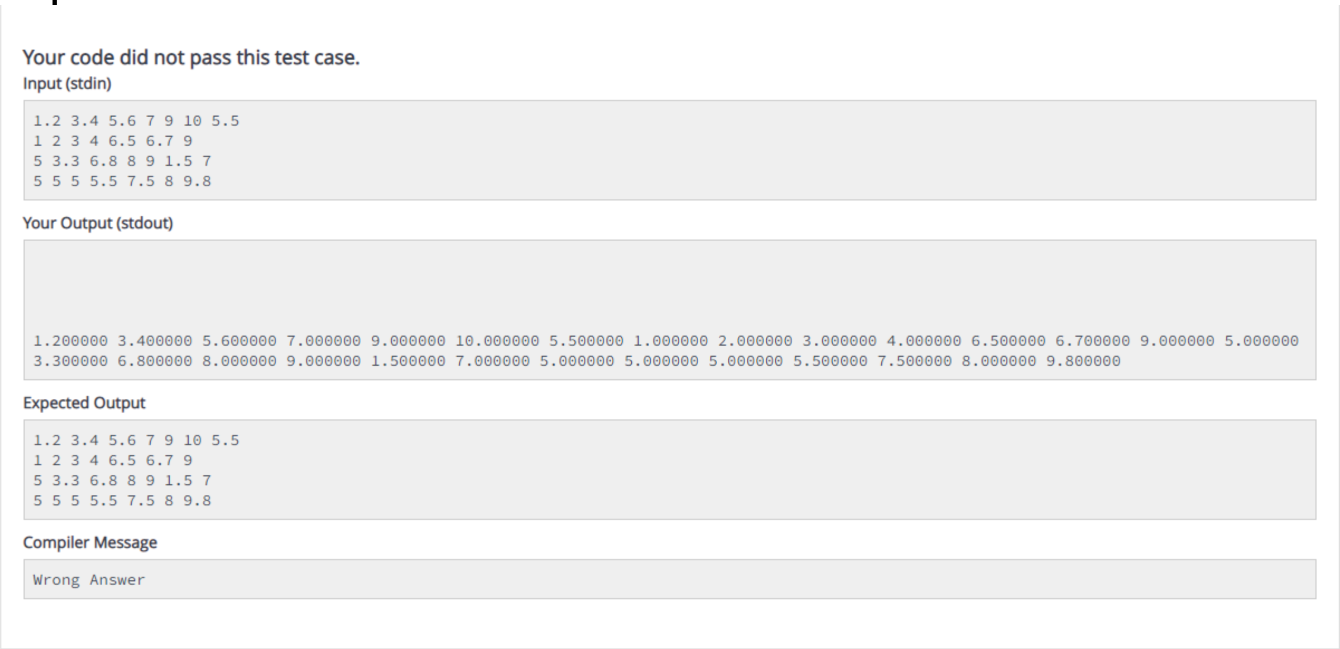
```
1
5
2
4
3
```

8. Write down a program to store humidity of four different cities of a week and display its value.

Code :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    float hum[4][7];
    int i,j;
    for(i=0;i<4;i++)
    {
        for(j=0;j<7;j++)
        {
            scanf("%f ",&hum[i][j]);
        }
        printf("\n");
    }
    for(i=0;i<4;i++)
    {
        for(j=0;j<7;j++)
        {
            printf("%f ",hum[i][j]);
        }
    }
    return 0;
}
```

Output :



Your code did not pass this test case.

Input (stdin)

```
1.2 3.4 5.6 7 9 10 5.5
1 2 3 4 6.5 6.7 9
5 3.3 6.8 8 9 1.5 7
5 5 5 5.5 7.5 8 9.8
```

Your Output (stdout)

```




1.200000 3.400000 5.600000 7.000000 9.000000 10.000000 5.500000 1.000000 2.000000 3.000000 4.000000 6.500000 6.700000 9.000000 5.000000
3.300000 6.800000 8.000000 9.000000 1.500000 7.000000 5.000000 5.000000 5.000000 5.500000 7.500000 8.000000 9.800000
```

Expected Output

```
1.2 3.4 5.6 7 9 10 5.5
1 2 3 4 6.5 6.7 9
5 3.3 6.8 8 9 1.5 7
5 5 5 5.5 7.5 8 9.8
```

Compiler Message

```
Wrong Answer
```