Question **1**

Correct

Marked out of 1.00

Flag question

Write a program to find the largest and second largest elements with in the elements of the given one dimensional array.

For example, if the user gives the **input** as:
5

Next, the program should print the messages one by one on the conscle.

If the user gives the **input** as:
10
50
30
20
25

then the program should **print** the result as:

```
The largest element of the array =
The second largest element of the a
```

Hints
Let us assume first element it self as the **large**, **second_large** and then compare both with all the other elements.
If any one found as **large** then change the value of the large, otherwise compare it with **second_large** and exchange it when necessary condition is satisfied.

For example:

Input	Result
5 2 5 9 7 3	The largest element of the a The second largest element o
6 64 87 34 58 62 18	The largest element of the a The second largest element o

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
#include <limits.h>

int main() {
    int v;
    scanf("%d", &v);
    int arr[v];

    if (v < 2) {
        printf("Number of
        return 0;
    }

    for (int i = 0; i < v
        scanf("%d", &arr[
    }

    int lNum = INT_MIN, s
    for (int i = 0; i < v
        if (lNum < arr[i]
            sNum = lNum;
            lNum = arr[i]
        } else if (sNum <
            sNum = arr[i]
        }
    }

    printf("The largest e

    return 0;
}
```

	Input	Expected
✓	5 2 5 9 7 3	The largest element o The second largest el
✓	6 64 87 34 58 62 18	The largest element o The second largest el

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

Write a program to find the minimum and second minimum elements with in the elements of one dimensional array.

Constraints:

- 1 <= N <= 10^3
- 1 <= Elements of the array <= 10^6

Instruction: To run your custom test cases strictly map your input and output layout with the visible test cases.

Hints
Let us assume that first element it self as the **minimum**, **second_minimum** and then compare both with all the other elements.
If any one found as **minimum** then change the value of the minimum, otherwise compare it with **second_minimum** and exchange it when necessary condition is satisfied.

For example:

Input	Result
4 65 32 85 96	Min element = 32 Second min element =

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
#include <limits.h>

int main() {
    int v;
    scanf("%d", &v);

    int arr[v];
    for (int i = 0; i < v
        scanf("%d", &arr[
    }

    int n1 = INT_MAX, n2
    for (int i = 0; i < v
        if (arr[i] < n1)
            n2 = n1;
            n1 = arr[i];
        } else if (arr[i]
            n2 = arr[i];
        }
    }

    printf("Min element =
}
```

	Input	Expected
✓	4	Min element =

Question 3

Correct

Marked out of 1.00

Flag question

Write a program to read a student n subjects marks in an array and find the total, average of the marks.

For example, if the user gives the **input** as:
3

Next, the program should print the messages one by one on the console.
If the user gives the **input** as:
75
80
85

then the program should **print** the result as:

```
The total marks = 240
The average marks = 80.000000
```

Hints
marks are integers, **total** is also an integer but **average** is a float value, so typecast it.

For example:

Input	Result
5 45 65 55 75 85	The total marks = 325 The average marks = 65.00000
4 36 45 38 56	The total marks = 175 The average marks = 43.75000

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main() {
    int v;
    scanf("%d", &v);
    int t = 0;

    for (int i = 0; i < v
        int x;
        scanf("%d", &x);
        t = t + x;
    }

    float x = (t * 1.0) /
    printf("The total mar
}
```

	Input	Expected
✓	5 45 65 55 75 85	The total marks = 325 The average marks = 6
✓	4 36 45 38 56	The total marks = 175 The average marks = 4

Passed all tests! ✓

Question 4

Correct

Marked out of 1.00

Flag question

The below sample code finds the addition of two matrices.

In the main() function read a two two-dimensional array of elements and then find the addition of two matrices.

The logic is

- First checks the **row sizes** and **column sizes** of two two-dimensional arrays are equal or not.
- If the sizes are not equal then print "Addition is not possible" and stop the process.
- If the sizes are equal then use **two for loops** to add each corresponding elements of two matrices and finally print the result.

Fill in the missing code so that it produces the desired output.

For example:

Input	Result
2 2 1 2 3 4 2 2 4 5 6 7	The given matrix-1 is 1 2 3 4 The given matrix-2 is 4 5 6 7 Addition of two matrices 5 7 9 11

Answer: (penalty regime: 0 %)

Reset answer

```
    for ( j=0;j<n;j++)
    { // Complete the c
        scanf("%d", &a[
    }
}
scanf("%d %d", &p, &q);
for (i=0;i<p;i++ )
{ // Complete the code
    for (j=0;j<q;j++ )
    { // Complete the c
        scanf("%d", &b[
    }
}
printf("The given matri
for(i=0;i<m;i++){
    for(j=0;j<n;j++){
        printf("%d ",a[
    }
    printf("\n");
}
// Write the code to di
printf("The given matri
for(i=0;i<p;i++){
    for(j=0;j<q;j++){
        printf("%d ",b[
    }
    printf("\n");
}

// Write the code to di
if (p==m && q==n ){
printf("Addition of two
// Write the condition
    for (i=0;i<m;i++ )
    { // Complete the c
        for ( j=0;j<n;j
        { // Complete t
            c[i][j] = a
            printf("%d
        }
        printf("\n");
    }


    // Write the code t
}
else
{
    printf("Addition is
}
return 0;
```

	Input	Expected
✓	2 2 1 2 3 4 2 2 4 5 6 7	The given matrix-1 1 2 3 4 The given matrix-2 4 5 6 7 Addition of two ma 5 7 9 11

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code which counts the number of vowels, consonants, digits and spaces are presented in a given string.

Initially, the variables vowels, consonants, digits and spaces are initialized to 0.

Iterate the string from the **first** character to **last** character to find all vowels, consonants, digits and spaces.

When a vowel character is found, vowel variable is incremented by 1. Similarly, consonants, digits and spaces are incremented when these characters are found in the string.

Finally, the count is displayed on the screen.

For example:

Input
kohli hits 100 in every cricket ma

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>
#include <ctype.h>

int main() {
    char line[100];
    int i, vowels, conson

    vowels = consonants =

    fgets(line, sizeof(li

    for (i = 0; line[i] !
        if (line[i] == 'a
            line[i] == 'A
            ++vowels;
        } else if (isalph
            ++consonants;
        } else if (line[i
            ++digits;
        } else if (line[i
            ++spaces;
        }
    }

    printf("Vowels = %d\n
    printf("Consonants =
    printf("Digits = %d\n
    printf("White spaces

    return 0;
}
```

	Input
✓	kohli hits 100 in every cri

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code which copies a given string into another string.

Initially, read a string from the standard input device and write a loop to copy each character of given string into another string till the end of the string is reached.

Place '\0' at the end of the copied string.

Finally, the copied string is displayed on the screen.

For example:

Input	Result
GangaRiver	The copied string = Ga

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main() {
    char str1[50], str2[5
    int i;

    scanf("%s", str1);

    for (i = 0; str1[i] !
        str2[i] = str1[i]
    }
    str2[i] = '\0';

    printf("The copied st
    return 0;
}
```

	Input	Expected
✓	GangaRiver	The copied stri

Passed all tests! ✓

Passed all tests! ✓

Question 3

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code which concatenates two given strings and store the result in another string.

Read two strings from the standard input device and write a loop to copy each character of the first string into third string till the end of the first string.

Write another loop to copy each character of the second string into third string till the end of second string.

Now place '\0' at the end of the third string.

Finally, display the third string.

For example:

Input	Result
Narendra Modi	NarendraModi

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main() {
    char a[20], b[20], c[
    int i, j;

    scanf("%s", a);
    scanf("%s", b);

    for (i = 0; a[i] != '
        c[i] = a[i]; // C
    }

    for (j = 0; b[j] != '
        c[i++] = b[j]; //
    }

    c[i] = '\0'; // Compl

    printf("%s\n", c);
    return 0;
}
```

	Input	Expected	Got
✓	Narendra Modi	NarendraModi	Nare

Passed all tests! ✓

Question 4

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to check whether the given two strings are equal or not.

Read two strings from the standard input device and write a loop to check each character of the first string with second string till the end of the first string is reached.

If any character is not equal then break the loop and say "**Two strings are not equal**".

If all the characters are equal and the length of two strings is also equal then display "**Two strings are equal**".

For example:

Input	Result
Godavari Godavari	Two strings are equal
Narmada narmada	Two strings are not equa

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main() {
    char a[20], b[20];
    int i = 0, flag = 1;

    scanf("%s", a);
    scanf("%s", b);

    while (a[i] != '\0')
        if (a[i] != b[i])
            flag = 0;
            break;
        }
        i++;
    }

    if (flag == 1) {
        printf("Two strin
    } else {
        printf("Two strin
    }

    return 0;
}
```

	Input	Expected
✓	Godavari Godavari	Two strings are e
✓	Narmada narmada	Two strings are n

Passed all tests! ✓

Question **5**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to search the occurrence of a given character in a given string.

Read a string and a character from the standard input device and write a loop to check each character of the string with a given character.

If the given character is equal to a character in the string then increment the count with in the loop.

Finally, display the count variable which has the total number of occurrences of the given character.

For example:

Input	Result
CurrencyDemonitisation n	Occurence

Answer: (penalty regime: 0 %)

Reset answer

```c
#include <stdio.h>
#include <string.h>

int main() {
    char str[20], ch;
    int count = 0, i;

    scanf("%s", str);
    scanf(" %c", &ch);

    for (i = 0; str[i] !=
        if (str[i] == ch)
            count++;
        }
    }

    if (count == 0) {
        printf("The chara
    } else {
        printf("Occurence
    }

    return 0;
}
```

	Input	Exp
✓	CurrencyDemonitisation n	Occ

Passed all tests! ✓

Question **6**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to count total number of uppercase and lowercase characters from the accepted string.

Read a string from the standard input device and write a loop to check each character, whether it is uppercase or lowercase of the given string.

If the given character is uppercase then increment the upper_count with in the loop.

If the given character is lowercase then increment the lower_count with in the loop.

Finally display
the upper_count and lower_count.

For example:

Input	Result
KrishnaAndGodavariAreRivers	Numbe Numbe

Answer: (penalty regime: 0 %)

Reset answer

```c
#include <stdio.h>

int main() {
    int upper_count = 0,
    char ch[80];
    int i = 0;

    scanf("%s", ch); // C

    while (ch[i] != '\0')
        if (ch[i] >= 'A'
            upper_count++
        }

        if (ch[i] >= 'a'
            lower_count++
        }

        i++;
    }

    printf("Number of upp
    printf("Number of low

    return 0;
}
```

	Input
✓	KrishnaAndGodavariAreRivers

Passed all tests! ✓

Question **7**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to reverse the given string.

Hints
Step:1 Read a string from the standard input device.
Step:2 Write a loop to find the length of the string.
Step:3 Write another loop to interchange the characters from first to last of the string.
Step:4 Finally display the reverse of a string.

For example:

Input	Result
Software	The reverse of a given s

Answer: (penalty regime: 0 %)

Reset answer

```c
#include<stdio.h>

int main()
{
    char ch[80], temp[80]
    int i, j;

    scanf("%s", ch);

    i = j = 0;
    while (ch[j] != '\0')
        j++;
    j--;

    while (j >= 0)
    {
        temp[i] = ch[j];
        i++;
        j--;
    }
    temp[i] = '\0';

    printf("The reverse o
    return 0;
}
```

	Input	Expected
✓	Software	The reverse of a

Passed all tests! ✓

Question **8**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to check whether the given string is a palindrome or not.

Read a string from the standard input device and write a loop to check the characters of the given string with the reverse string.

If all the characters are equal then display **"The given string is a palindrome"**, otherwise display **"The given string is not a palindrome"**.

For example:

Input	Result
12321	The given string 12321
amaravathi	The given string amara

Answer: (penalty regime: 0 %)

Reset answer

```c
#include <stdio.h>

int main() {
    char ch[80];
    int i, j, length, fla
    scanf("%s", ch); // C

    length = 0;
    while (ch[length] !=
        length++;

    }

    j = length - 1;
    for (i = 0; i < lengt
        if (ch[i] != ch[j
            flag = 1;
            break;

        }
        j--;

    }

    if (flag == 0) { // W
        printf("The given
    } else {
        printf("The given

    }

    return 0;
}
```

	Input	Expected
✓	12321	The given strin
✓	amaravathi	The given strin

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

In **C** language, we have four types of string functions that are used for performing **string operations**. They are strlen(), strcpy(), strcat(), strcmp().

The function strlen() is used to find the **length** of the given string. This function returns only the **integer data** (or) **numeric data**.

The function strlen() counts the number of characters in a given string and returns the integer value.

It stops counting the character when **NULL** character is found. Because, **NULL** character indicates the end of the string in **C**.

The syntax of strlen() is integer_variable = strlen(string);.

Here string is a group of characters, strlen() function finds the **length** of the string and the **integer** value will be stored in the integer_variable.

The string.h header file supports all the string functions in **C** language.

Fill in the missing code in the below program to find the **length** of a string using **strlen()** function.

For example:

Input	Result
NarendraModi	The length of the st

Answer: (penalty regime: 0 %)

Reset answer

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

int main() {
    char ch[20];
    scanf("%s", ch);
    printf("The length of
    return 0;
}
```

	Input	Expected
✓	NarendraModi	The length of

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

The function strcpy() is used to **copy** one string into another string including the NULL character (terminator char '\0').

The syntax of strcpy( ) is strcpy(string1, string2);.

Where string1, string2 are two strings and the string2 is copied into string1. In this case the copied string is available in string1 and both strings contains the same data.

If the length of string1 is less than the length of string2 then entire string2 value will not be copied into string1.

For example, consider the length of string1 is **20** and the length of string2 is **30**. Then, only the first **20** characters from string2 will be copied into string**1**, the remaining **10** characters will not be copied and will be **truncated**.

Understand and retype the below code which demonstrates the usage of **strcpy()** function.

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

int main()
{
  char str1[20], str2[20];
  scanf("%s", str2);
  strcpy(str1, str2);
  printf("The copied string = %s", str1);
  return 0;
}
```

For example:

Input	Result
Rose	The copied string = Rose

Answer: (penalty regime: 0 %)

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

int main() {
    char a[20], b[20];

    scanf("%s", b);
    strcpy(a, b);

    printf("The copied st

    return 0;
}
```

	Input	Expected
✓	Rose	The copied string = R

Passed all tests! ✓

Question **3**

Correct

Marked out of 1.00

Flag question

The function strcat() is used to concatenate two strings into a single string.

The syntax of strcat() is strcat(string1, string2);.

where string1, string2 are two different strings. Here string2 is concatenated with string1, and the **concatenated string** is stored in string1.

In strcat() operation, **NULL character ('\0')** of string1 is **overwritten** by first character of string2 and **NULL character ('\0')** is appended (added) at the end of **new** string1 which is created after strcat() operation.

Fill the missing code in the below program to display the **concatenated** string using **strcat()** function.

For example:

Input	Result
REC Chennai	RECChennai

Answer: (penalty regime: 0 %)

Reset answer

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

int main() {
    char str1[20], str2[2

    scanf("%s", str1);
    scanf("%s", str2);

    strcat(str1, str2);

    printf("%s\n", str1);

    return 0;
}
```

	Input	Expected	Got
✓	REC Chennai	RECChennai	RECCher

Passed all tests! ✓

Question **4**

Correct

Marked out of 1.00

Flag question

The function strcmp() is used for comparison of two strings and it always returns the numeric data. This function compares strings character by character using their ASCII values.

The syntax of strcmp() is variable_name = strcmp (string1, string2);.

Where string1, string2 are two strings and the variable is of **integer** datatype.

The comparison of two strings is dependent on the **alphabets (characters)** and not on the size (length) of the strings.

If the function strcmp() returns zero, both strings are **equal**.
If the function strcmp() returns a value which is less than zero, **string2** is higher than **string1** (because the ASCII value of first unmatched character of **string1** is less than the ASCII value of the corresponding character in **string2**)
If the function strcmp() returns a value which is greater than zero, **string1** is higher than **string2** (because the ASCII value of first unmatched character of **string1** is greater than the ASCII value of the corresponding character in **string2**)

Fill the missing code in the below program to compare two strings using **strcmp()** function.

For example:

Input	Result
NarendraModi narendramodi	The string narendram
Krishna Godavari	The string Krishna i
REC REC	The given two string

Answer: (penalty regime: 0 %)

Reset answer

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

int main() {
    char a[20], b[20];

    scanf("%s", a);
    scanf("%s", b);

    if (strcmp(a, b) == 0
        printf("The given
    } else if (strcmp(a,
        printf("The strin
    } else {
        printf("The strin

    }

    return 0;
}
```

	Input	Expected
✓	NarendraModi narendramodi	The string na
✓	Krishna Godavari	The string Kr
✓	REC REC	The given two

ll tests! ✓

Review

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{int a,b,c;
scanf("%d%d%d",&a,&b,&c);
if(a+b+c>=180)
{printf("The candidate is
}
else
{printf("The candidate is
}
}
```

	Input	Expected
✓	70 60 80	The candidate
✓	50 80 80	The candidate

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{int x,y,z,smallest;
scanf("%d\n%d\n%d\n",&x,&
smallest=x;
if(y<smallest)
{ printf ("%d",y);
}
else if(z<smallest)
{printf("%d",z);
}
else
{ printf("%d",x);
}
}
```

	Input	Expected	Got	
✓	40 30 50	30	30	✓
✓	15 15 15	15	15	✓