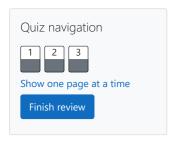
# GE23131-Programming Using C-2024



Status Finished

Started Monday, 23 December 2024, 5:33 PM

Completed Thursday, 17 October 2024, 10:49 AM

Duration 67 days 6 hours

Question **1**Correct
Marked out of 3.00

Flag

question

Write a program to input a name (as a single character) and marks of three tests as m1, m2, a considering all the three marks have been given in integer format.

Now, you need to calculate the average of the given marks and print it along with the name a format section.

All the test marks are in integers and hence calculate the average in integer as well. That is, ye part of the average only and neglect the decimal part.

Input format:

Line 1 : Name(Single character)

Line 2: Marks scored in the 3 tests separated by single space.

Output format:

First line of output prints the name of the student.

Second line of the output prints the average mark.

Constraints

Marks for each student lie in the range 0 to 100 (both inclusive)

Sample Input 1:

Α

3 4 6

Sample Output 1:

Α

4

Sample Input 2:

Τ

738

Sample Output 2:

Т

6

Answer: (penalty regime: 0 %)

Input	Expected	Got
A	A	A
3 4 6	4	4
T	T	T
7 3 8	6	6
R	R	R
0 100 99	66	66

Passed all tests!

Question **2**Correct

Marked out of 5.00

□ Flag question

Some  ${\it C}$  data types, their format specifiers, and their most common bit widths are as follows:

- · Int ("%d"): 32 Bit integer
- · Long ("%ld"): 64 bit integer
- · Char ("%c"): Character type
- Float ("%f"): 32 bit real value
- · Double ("%lf"): 64 bit real value

## Reading

To read a data type, use the following syntax:

scanf("`format\_specifier`", &val)

For example, to read a *character* followed by a *double*:

char ch;

double d;

scanf("%c %lf", &ch, &d);

For the moment, we can ignore the spacing between format specifiers.

## Printing

To print a data type, use the following syntax:

printf("`format\_specifier`", val)

For example, to print a *character* followed by a *double*:

char ch = 'd';

double d = 234.432;

printf("%c %lf", ch, d);

**Note:** You can also use *cin* and *cout* instead of *scanf* and *printf*, however, if you are taking a n printing a million lines, it is faster to use *scanf* and *printf*.

### **Input Format**

Input consists of the following space-separated values: int, long, char, float, and double, respe

#### **Output Format**

Print each element on a new line in the same order it was received as input. Note that the flo correct up to 3 decimal places and the double to 9 decimal places.

## Sample Input

3 12345678912345 a 334.23 14049.30493

#### **Sample Output**

3

12345678912345

14049.304930000

## **Explanation**

Print int 3,

followed by long 12345678912345,

followed by char **a**,

followed by float 334.23,

followed by double 14049.30493.

Answer: (penalty regime: 0 %)

Input	Expected	Got
3 12345678912345 a 334.23 14049.30493	3 12345678912345 a 334.230 14049.304930000	3 12345678912345 a 334.230 14049.304930000

Passed all tests!

Question **3**Correct
Marked out of 7.00

Flag question

Write a program to print the ASCII value and the two adjacent characters of the given charac

Input

Ε

Output

69

DF

Answer: (penalty regime: 0 %)