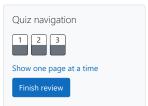
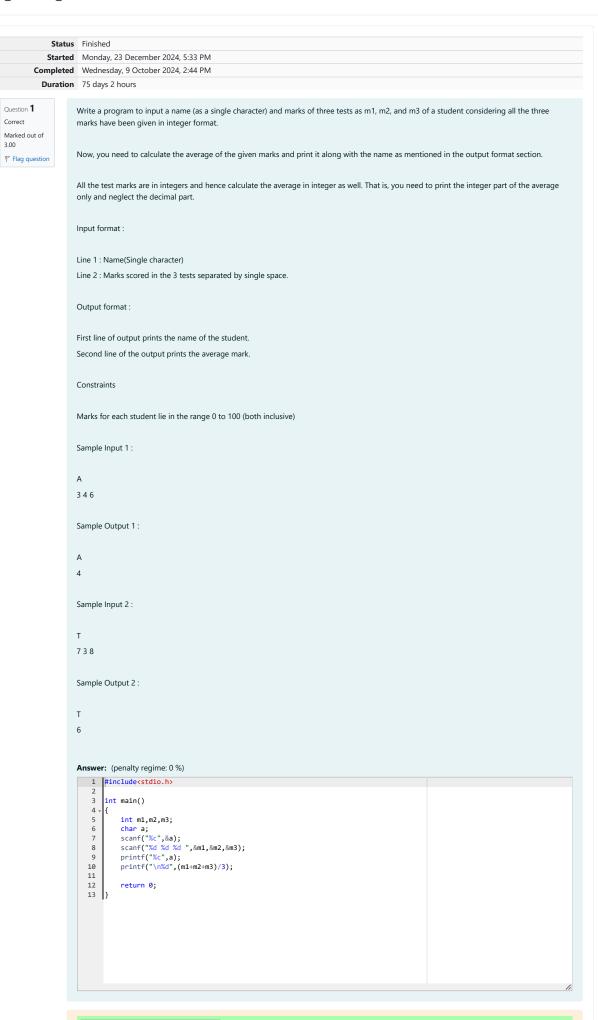
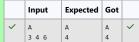
GE23131-Programming Using C-2024

Correct







~	Т	Т	Т	~
	7 3 8	6	6	
~	R	R	R	~
	0 100 99	66	66	

Passed all tests! ✓

Question ${\bf 2}$ 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:

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 million numbers as input and printing a million lines, it is faster to use scanf and printf.

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

Print each element on a new line in the same order it was received as input. Note that the floating point value should be 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

а

334.230

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 %)

```
1 #include <stdio.h>
            int main()
     4
                   long b;
char c;
    6
                   float d;
double e;
    8
   10
                    scanf("%d",&a);
   11
                   scant("%0",&a);
scanf("\n%ld",&b);
scanf("\n%c",&c);
scanf("\n%f",&d);
scanf("\n%lf",&e);
   12
   13
   14
15
   16
17
                   printf("%d",a);
printf("\n%1d",b);
printf("\n%c",c);
printf("\n%3.3f",d);
printf("\n%6.91f",e);
   18
19
   20
21
   22
23
                    return 0;
   24
```

`	~
15	15

334.230 334.230 14049.304930000 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 character.
```

Input

Ε

Output

69

D F

Answer: (penalty regime: 0 %)

```
char a;
scanf("%c",&a);
printf("%d",a);
         printf("\n%c %c",a-1,a+1);
```

	Input	Expected	Got	
~	E	69	69	~
		D F	DF	

Passed all tests! ✓