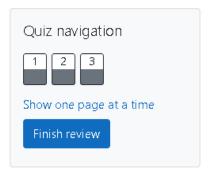
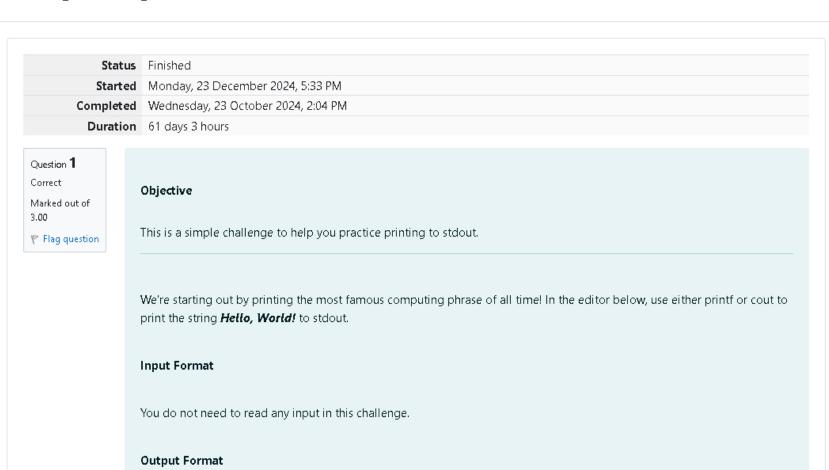
GE23131-Programming Using C-2024





Print Hello, World! to stdout.

Answer: (penalty regime: 0 %)

Sample Output

Hello, World!

```
#include<stdio.h>
int main()
3 * {
    printf("Hello, World!");
    return 0;
}
```

o, World! 🗸

Question **2**Correct Marked out of 5.00

Flag question

Objective

This challenge will help you to learn how to take a character, a string and a sentence as input in C.

To take a single character **ch** as input, you can use scanf("%c", &ch); and printf("%c", ch) writes a character specified by the argument char to stdout:

char ch; scanf("%c", &ch); printf("%c", ch);

This piece of code prints the character ch.

Task

You have to print the character, ch.

Input Format

Take a character, ch as input.

Output Format

Print the character, ch.

Answer: (penalty regime: 0 %)



Passed all tests! <

Question **3** Correct Marked out of 7.00

P Flag question

Objective

The fundamental data types in c are int, float and char. Today, we're discussing int and float data types.

The printf() function prints the given statement to the console. The syntax is printf("format string",argument_list);. In the function, if we are using an integer, character, string or float as argument, then in the format string we have to write %d (integer), %c (character), %s (string), %f (float) respectively.

The scanf() function reads the input data from the console. The syntax is scanf("format string", argument_list);. For ex:

The scanf("%d",&number) statement reads integer number from the console and stores the given value in variable *number*.

To input two integers separated by a space on a single line, the command is scanf("%d %d", &n, &m), where n and m are the two integers.

Task

Your task is to take two numbers of int data type, two numbers of float data type as input and output their sum:

- 1. Declare 4 variables: two of type int and two of type float.
- 2. Read **2** lines of input from stdin (according to the sequence given in the 'Input Format' section below) and initialize your **4** variables.
- 3. Use the + and operator to perform the following operations:
- o Print the sum and difference of two intivariable on a new line.
- o Print the sum and difference of two float variable rounded to one decimal place on a new line.

Input Format

The first line contains two integers.

The second line contains two floating point numbers.

Constraints

- 1 ≤ integer variables ≤ 10⁴
- 1 ≤ float variables ≤ 10⁴

Output Format

Print the sum and difference of both integers separated by a space on the first line, and the sum and difference of both float (scaled to 1 decimal place) separated by a space on the second line.

Sample Input

10.4

4.0 2.0

Sample Output

14 6

6.0 2.0

Explanation

When we sum the integers 10 and 4, we get the integer 14. When we subtract the second number 4 from the first

number 10, we get b as their difference.

When we sum the floating-point numbers **4.0** and **2.0**, we get **6.0**. When we subtract the second number **2.0** from the first number **4.0**, we get **2.0** as their difference.

Answer: (penalty regime: 0 %)

```
1 #include(stdio.h>
2 int main()
3 - {
4
      int a,b;
5
      float c,d;
    scanf("%d%d",&a,&b);
      scanf("%f%f",&c,&d);
7
      printf("%d %d n",a+b,a-b);
8
       printf("%.1f %.1f \n",c+d,c-d);
9
       return 0;
10
11 }
```

	Input	Expected	Got	
~	10 4 4.0 2.0	14 6 6.0 2.0	14 6 6.0 2.0	~
~	20 8 8.0 4.0	28 12 12.0 4.0	28 12 12.0 4.0	~

Passed all tests! 🗸