

Status	Finished
Started	Friday, 26 September 2025, 12:22 PM
Completed	Friday, 26 September 2025, 12:58 PM
Duration	36 mins 48 secs
Marks	3.00/3.00
Grade	10.00 out of 10.00 (100%)

Question 1

Correct

Mark 1.00 out of 1.00

objective

this is a simple challenge to help you practice printing to stdout.|

we're starting out by printing the most famous computing phrase of all time! in the editor below, use either printf or cout to print the string **hello, world!** to stdout.

Input Format

You do not need to read any input in this challenge.

Output Format

Print **Hello, World!** to stdout.

Sample Output

Hello, World!

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("Hello, World!");
5     return 0;
6 }
```

	Expected	Got	
>Hello, World!	Hello, World!	Hello, World!	

Passed all tests!

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

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 %)

```
1 #include<stdio.h>  
2 int main()  
3 {  
4     char ch;  
5     scanf("%c",&ch);  
6     printf("%c",ch);  
7     return 0;  
8 }
```

	Input	Expected	Got	
$\text{m} \tilde{\text{l}}$	c	c	c	$\text{m} \tilde{\text{l}}$

Passed all tests!

Correct

Marks for this submission: 1.00/1.00.

Question 3

Correct

Mark 1.00 out of 1.00

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:
 - * print the sum and difference of two int variable on a new line.
 - * 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**

· $1 \leq \text{float variables} \leq 10^4$

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
```

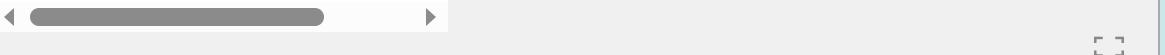
explanation

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 **6** 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;
6     scanf("%d%d",&a,&b);
7     scanf("%f%f",&c,&d);
8     printf("%d %d",a+b,a-b);
9     printf("\n%.1f %.1f",c+d,
10    return 0;
11 }
```



	Input	Expected	Got	
$\pi\Gamma$	10 4 4.0 2.0	14 6 6.0 2.0	14 6 6.0 2.0	$\pi\Gamma$
$\pi\Gamma$	20 8 8.0 4.0	28 12 12.0 4.0	28 12 12.0 4.0	$\pi\Gamma$

Passed all tests! $\pi\Gamma$

Correct

Marks for this submission: 1.00/1.00.