WEEK-01 Name: FARHAN

Question 1 WEEK-01-01 Correct

Reg No: 240701136

Marked out of 3.00 Flag question **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.

- Hello, World! **Answer:** (penalty regime: 0 %)
- #include<stdio.h> 2 v int main(){ printf("Hello, World!"); 3
- 4 return 0; } 5
- **Expected** Got Hello, World! Hello, World! Passed all tests! < 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**.
- Take a character, *ch* as input. **Output Format**

Answer: (penalty regime: 0 %)

#include<stdio.h>

Input Expected

C

data types.

output their sum:

1.

2.

3.

line.

Constraints

Output Format

Sample Output

14 6

6.0 2.0

6

7

8

9 10 Got

C

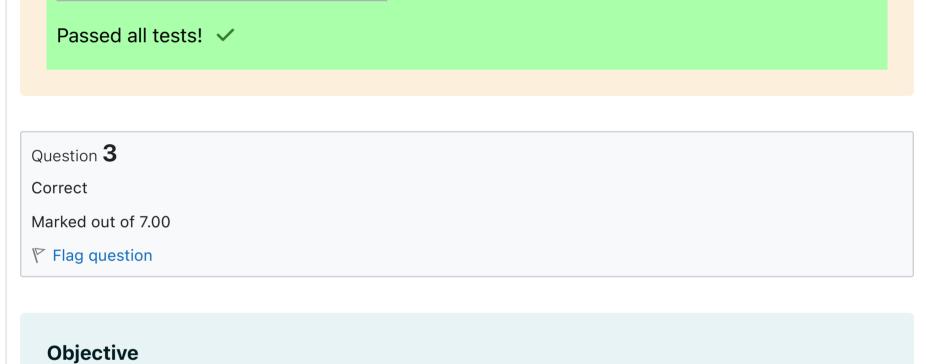
argument, then in the format string we have to

Print the character, *ch*.

7

Input Format

- 2 v int main(){ 3 char c; scanf("%c",&c); 4 printf("%c",c); 5 return 0;



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

The printf() function prints the given statement to the console. The syntax is printf("format

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

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

Read 2 lines of input from stdin (according to the sequence given in the 'Input Format'

Print the sum and difference of two float variable rounded to one decimal place on a new

string", argument_list);. In the function, if we are using an integer, character, string or float as

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

write %d (integer), %c (character), %s (string), %f (float) respectively.

from the console and stores the given value in variable *number*.

Input Format

Declare 4 variables: two of type int and two of type float.

Use the + and - operator to perform the following operations:

Print the sum and difference of two int variable on a new line.

section below) and initialize your 4 variables.

The first line contains two integers.

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

The second line contains two floating point numbers.

- **Sample Input** 10 4 4.0 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.

scanf("%f %f",&c,&d);

return 0;

printf("%d %d\n",a+b,a-b);

printf("%.1f %.1f",c+d,c-d);

- **Answer:** (penalty regime: 0 %) #include<stdio.h> 1 2 v int main(){ int a,b; 3 4 float c,d; scanf("%d %d",&a,&b); 5
 - Input **Expected** Got 10 4 14 6 14 6 4.0 2.0 6.0 2.0 6.0 2.0 20 8 28 12 28 12 8.0 4.0 12.0 4.0 12.0 4.0

Finish review

Passed all tests! <

WEEK-01-02 Question 1 Correct Marked out of 3.00 Flag question as mentioned in the output format section. All the test marks are in integers and hence calculate the average in integer as well. That is, Input format: Line 1: Name(Single character) Line 2: Marks scored in the 3 tests separated by single space.

Write a program to input a name (as a single character) and marks of three tests as m1, m2, and m3 of a student 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

you need to print the integer part of the average only and neglect the decimal part.

Second line of the output prints the average mark.

Output format: First line of output prints the name of the student.

Constraints

Marks for each student lie in the range 0 to 100 (both inclusive) Sample Input 1:

Α

346

Sample Output 1:

Α 4

Sample Input 2: Т

738

Sample Output 2: Τ

6

Answer: (penalty regime: 0 %)

1

3

4

5

6 7

8

Question 2 Correct Marked out of 5.00

Flag question Some C data types, their format specifiers, and their most common bit widths are as follows:

Reading char ch; double d;

Printing

double d = 234.432;

places.

Sample Input

Sample Output

12345678912345

char ch = 'd';

а 334.230 14049.304930000 **Explanation**

3

Answer: (penalty regime: 0 %)

2 🔻

3

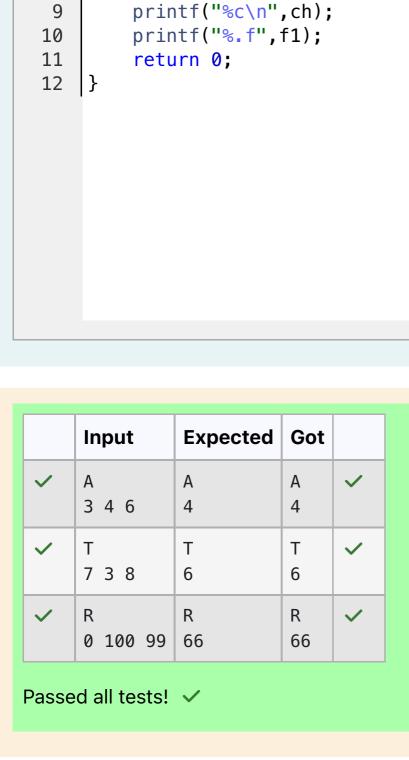
4 5

6 7

8

Input Ε

Quiz navigation Show one page at a time Finish review



#include<stdio.h>

char ch;

float f1;

f1=m4/3;

m4=m1+m2+m3;

int m1, m2, m3, m4;

scanf("%c %d %d %d",&ch,&m1,&m2,&m3);

2 v int main(){

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 To read a data type, use the following syntax: scanf("`format_specifier`", &val) For example, to read a *character* followed by a *double*:

To print a data type, use the following syntax:

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

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

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

Expected

334.230

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

12345678912345

Got

334.230

14049.304930000 | 14049.304930000

12345678912345

Finish review

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

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

printf("`format_specifier`", val)

Print int 3, followed by long 12345678912345, followed by char a, followed by float 334.23,

followed by double 14049.30493.

#include<stdio.h>

int main(){

int a; long l;

char ch; float f;

double d;

3 12345678912345 a 334.23 14049.30493

scanf("%d %ld %c %f %lf",&a,&l,&ch,&f,&d); printf("%d\n%ld\n%c\n%.3f\n%.9lf",a,l,ch,f,d); 9 return 0; 10 11 }

3 12345678912345 a 334.23 14049.30493

Passed all tests! < Question 3 Correct Marked out of 7.00 Flag question

Input

Output 69 DF **Answer:** (penalty regime: 0 %) #include<stdio.h> 2 🔻 int main(){ char ch,befch,aftch; 3 scanf("%c",&ch); 4 5 befch=ch-1; 6 aftch=ch+1;

7

8

9

10

}

character.

Input | Expected | Got Ε Passed all tests! <

69

DF

printf("%d\n",ch);

return 0;

printf("%c %c",befch,aftch);

69

DF