Week-01-Overview of C, Constants, Variables and Data Types

Week 01-02 Practice Session Coding

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

Source Code

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int m1,m2,m3;
    char n;
    scanf("%c\n",&n);
    scanf("%d %d %d",&m1,&m2,&m3);
    if(m1<=100 && m2<=100 && m3<=100)
    {
        printf("%c\n",n);
        printf("%d",(m1+m2+m3)/3);
    }
    else
    {
        printf("Inavalid Marks");
    }
    return 0;
}
```

Result

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

Passed all tests! ✓

Some 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)

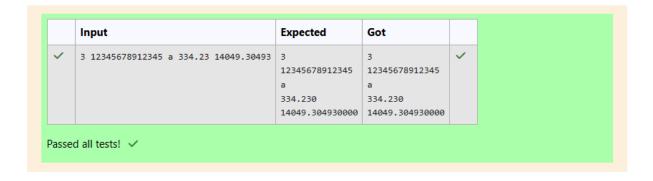For example, to read a *character* followed by a *double*:
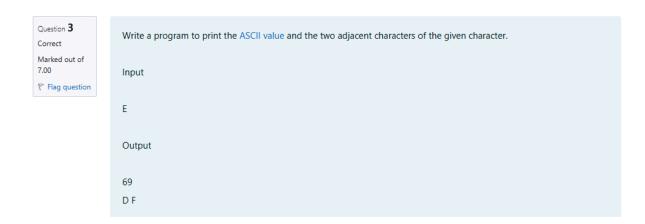
char ch;

double d;

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

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

Source Code

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int i;
    char c;
    double lf;
    long ld;
    float f;
    scanf("%d %ld %c %f %lf",&i,&ld,&c,&f,&lf);
    printf("%d\n%ld\n%c\n%.3f\n%.9lf\n",i,ld,c,f,lf);
    return 0;
}
```

Result

	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! ✓

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

Input

E

Output

69

D F

Source Code

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>

int main()
{
    char x;
    scanf("%c",&x);
    printf("%d\n",x);
    printf("%c %c",x-1,x+1);
    return 0;
}
```

Result

	Input	Expected	Got	
✓	E	69 D F	69 D F	✓

Passed all tests! ✓