

CS 36
Lab 1 (5 points)
Lesson 1 to Lesson 14

(no loops, arrays or anything thing we have not covered in class,
you can declare strings as `char strname[value]`)

3 test runs for each question

Save you lab file as lastname_firstname_lab01.txt

1. Follow the Lab instructions and video on how to submit labs.
2. Follow the steps in the Lab instruction sample on Canvas to separate each question with a banner with question number, short description of the question.
3. Do not submit question 1
4. Submit questions 2 to 6 only. Must number your questions from 2 to 6. For each question you must provide output for 3 test runs (use the sample test run data already provided plus makeup 2 more yourself). If you do not have 3 test runs total, that question will result in a zero.
5. You must use the data given in the sample test runs that are given in the question. Provide your own data whenever there is no sample run data.
6. If the program does not run a zero will be given
7. If the program runs but does not fulfil all the specifications stated in the question, a zero score will be given.
8. You are not allowed to use loops, arrays, if statements, `#include<math.h>`, `#include<iostream>`, `#include<stdlib.h>`, `#include<string.h>`, and any other topics not covered in lesson 1 to lesson 14.
9. Using any other topics not covered in lesson 1 to lesson 14 will result in a zero for that question.

A Quick lesson on input/output in C

Input statements use `scanf()` – must use the `&` address operator next to variable store input value

```
int num1;  
float num2;  
char c;
```

//to input a single integer

```
scanf("%d", &num1);    // %d is type specifier for integer,
```

//to input a float integer

```
scanf("%f", &num2);    // use %f for floats
```

//to input a single character

```
scanf("%c", &c);        // use %c for characters
```

//to input more than one value in one `scanf()` statement

```
scanf("%d%f", &num1, &num2);    // user input two number on one line separated by a  
                                // space e.g 10 0.05
```

Output statements using `printf()` – no `&` next to variable to print.

```
num1 = 10;  
num2 = 23.5;  
c = 'H';
```

//To print a single integer num1 with a sentence(string)

```
printf("The integer is %d", num1);    // prints The integer is 10
```

// %d is the type specifier for integer and it is placed
//where you want to print the variable in the string

//To print a single float num2 with a sentence(string)

printf("The float number is %f", num2); //prints The float number is 23.5

//%f is the float type specifier and it is placed where
//you want to print the variable in the string

//To print a single character c with a sentence(string)

printf("The character is %c", c); //prints The character is H

//%c is the character type specifier and it is placed
//where you want to print the variable in the string

//to print more than one value in one printf() statement

printf("The integer %d and the float %f", num1, num2);

//prints The integer 10 and the float 23.5

Question 1 (do not submit)

Identify and correct the errors in each of the following statements. (Note: There may be more than one error per statement.)

a) scanf("d", value);

b) printf("The product of %d and %d is %d"\n, x, y);

c) firstNumber + secondNumber = sumOfNumbers

d) State the order of evaluation of the operators in each of the following C statements and show the value of x after each statement is performed.

1) x = 7 + 3 * 6 / 2 - 1;

2) x = 2 % 2 + 2 * 2 - 2 / 2;

3) x = (3 * 9 * (3 + (9 * 3 / (3))));

e) /* Program to determine the largest of three integers */

f) scanf("%d", anInteger);

g) printf("Remainder of %d divided by %d is\n", x, y, x % y);

h) print("The sum is %d\n," x + y);

i) printf("The value you entered is: %d\n", &value);

Question 2

Write a program that asks the user to enter two integers, obtains the two integers from the user and prints the sum, product, difference, quotient and remainder of the two numbers. Take the first input value minus the second, first input divide by second, and first input mod second. (hint : use correct datatype)

Here are two sample test runs (blue user input):

Test run 1

Enter two numbers: 20 5
The sum is 25
The product is 100
The difference is 15
The quotient is 4
The remainder is 0

Test run 2

Enter two numbers: 5 20
The sum is 25
The product is 100
The difference is -15
The quotient is 0
The remainder is 5

Test run 3 (provide your own data)

Question 3

Write a program that inputs one five-digit number, separate the number into its individual digits and prints the digits separated from one another by a blank space. [Hint: Use combinations of integer division and the remainder operation.] For example, if the user types in 42139 (no spaces), the program should print.

Here is a sample run (blue user input):

Test run 1

Enter a number : 42139
4 2 1 3 9

Test run 2 and 3 (provide your own data)

Question 4

Write a C program to find the diameter, area and circumference of a circle. Allow user to enter the radius of a circle. Display the output to 5 decimal places.

Here is a sample run (blue user input):

Test run 1

The radius of the circle is 5
The diameter of the circle is 10.00000
The area of the circle is 78.53975
The circumference is 31.41590

*You may use $\text{PI}=3.14159$ or if you want to experiment use $22/7$

Test run 2 and 3 (provide your own data)

Question 5

Write a C program to add two fractions. User must input exactly as in sample run, aka 5/6.

Here is a sample run (blue user input):

Test run 1

Enter First fraction : 5/6

Enter Second fraction : 3/4

The sum of the two fractions is 38/24

Test run 2 and 3 (provide your own data)

Question 6

Write a C Program that asks the user to enter a US dollar amount and then show how to pay that amount using the smallest number of \$20, \$10, \$5, \$1 bills. (hint: use Arithmetic operators)

Here is a sample run (blue user input):

Test run 1

Enter a dollar amount: 93

\$20 bills: 4

\$10 bills: 1

\$ 5 bills: 0

\$ 1 bills: 3

Test run 2 and 3 (provide your own data)