

## CMIS 102 Hands-On Lab

### Week 2

#### Overview

This hands-on lab allows you to follow and experiment with the critical steps of developing a program including the program description, analysis, test plan, design, and implementation with C code.

#### Program Description

This program will sum two integer numbers to yield a third integer number. Once the calculations are made the results of all the numbers will be printed to the output screen.

#### Analysis

We will use sequential programming statements.

We will define 3 integer numbers: a, b, c.

c will store the sum of a and b.

#### Test Plan

To understand this program the following input numbers could be used for testing:

a = 10

b = 20

c = a + b = 10 + 20 = 30

In table format the following results are expected:

Run #	Input a	Input b	Expected Output
1	10	20	30
2	0	0	0
3	124	356	480
4	-30	-90	-120

#### Design using Pseudocode

```
// This program will sum two integer numbers to yield a third integer number.  
// It will also divide two float numbers to yield a third float number.  
  
// Declare variables  
Declare a,b,c as Integer  
  
// Set values of Integers  
Set a=10  
Set b=20  
Set c= a + b  
  
// Print a, b, c  
Print a,b,c
```

## C Code

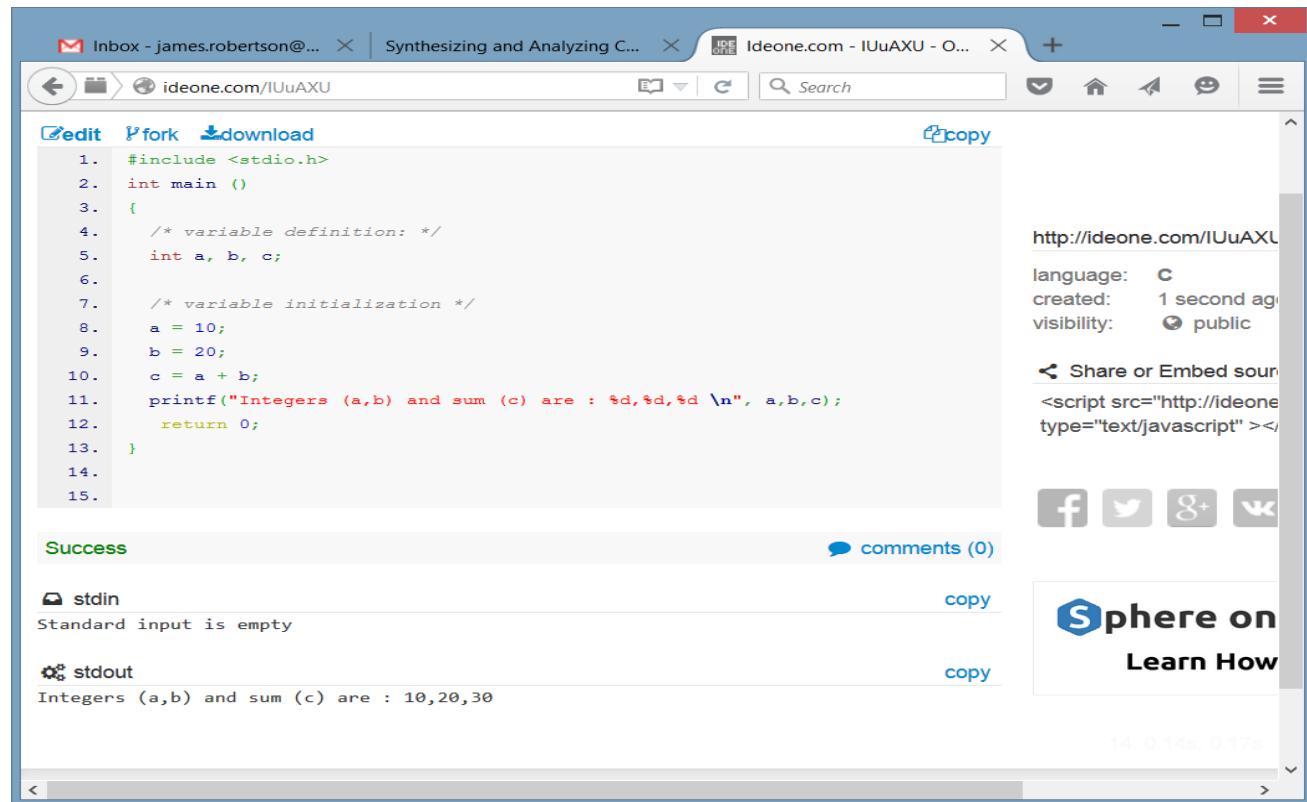
The following is the C Code that will compile and execute in the online compilers.

```
// C code
// This program will sum two integer numbers to yield a third integer number.
// Developer: Faculty CMIS102
// Date: Jan 31, XXXX

#include <stdio.h>
int main ()
{
    /* variable definition: */
    int a, b, c;

    /* variable initialization */
    a = 10;
    b = 20;
    c = a + b;
    printf("Integers (a,b) and sum (c) are : %d,%d,%d \n", a,b,c);
    return 0;
}
```

Results from running the programming at ideone.com:



The screenshot shows the ideone.com interface with the following details:

- Code:**

```
1. #include <stdio.h>
2. int main ()
3. {
4.     /* variable definition: */
5.     int a, b, c;
6.
7.     /* variable initialization */
8.     a = 10;
9.     b = 20;
10.    c = a + b;
11.    printf("Integers (a,b) and sum (c) are : %d,%d,%d \n", a,b,c);
12.    return 0;
13.
14.
15.
```
- Execution Results:**
  - stdin:** Standard input is empty
  - stdout:** Integers (a,b) and sum (c) are : 10,20,30
- Sharing:** The URL <http://ideone.com/IUuAXL> is displayed, along with options to share or embed the source code.
- Comments:** 0 comments
- Sphere on Learn How:** A sidebar with the Sphere logo and a "Learn How" button.

### **Learning Exercises for you to complete**

1. Change the C code to calculate the product of two integers as opposed to the sum of two integers. Support your experimentation with a screen capture of executing the new code.
2. Prepare a new test table with at least 3 distinct test cases listing input and expected output for the product of two integers.
3. Change the C code to calculate the quotient (e.g. a/b) of two floats (e.g. 2.3/1.5). Hint: Use float variable types as opposed to integers. What happens if the denominator is 0.0? Support your experimentation with screen captures of executing the new code
4. Prepare a new test table with at least 3 distinct test cases listing input and expected output for the quotient of two floats.

### **Submission**

Submit a neatly organized word (or PDF) document that demonstrates you successfully executed this lab on your machine using an online compiler. You should provide a screen capture of the resulting output.

Also, provide the answers, associated screen captures, C Code and descriptions of your successful completion of learning exercises 1, 2, 3 and 4.

The answers to the learning exercises, screen captures, C code and descriptions can be included in the same neatly organized document you prepared as you ran this lab. Note the code can be embedded in the word document. However; be sure all code compiles and runs perfectly before submitting the document.

Submit your document no later than the due date listed in the syllabus or calendar.

### Grading guidelines

Submission	Points
Successfully demonstrates execution of this lab with online compiler. Includes a screen capture.	3
Successfully modifies the code to calculate the product of two integers.	2
Provides a new test table with at least 3 distinct test cases listing input and expected output for the product of two integers	1
Modifies the code to calculate the quotient of two floats. Describes what happens if the denominator is 0.0? Support your experimentation with screen captures of executing the new code	2
Provides a new test table with at least 3 distinct test cases listing input and expected output for the quotient of two floats	1
Document is well-organized, and contains minimal spelling and grammatical errors.	1
<b>Total</b>	<b>10</b>