CMIS 102 Hands-On Lab

Week 4

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 (using pseudocode), and implementation with C code. The example provided uses sequential and selection statements.

Program Description

This program will calculate the sum of 5 integers. The program will ask the user to 5 integers. If the sum of the numbers is greater than 100, a message is printed stating the sum is over 100. The design step will include both pseudocode.

Analysis

I will use sequential, and selection programming statements.
I will define six integer numbers: value1, value2, value3, value4, value5 and sum. The value1, value2, value3, value4 and value5 variables will store the integers input by the user. The sum will store the sum of the 5 values.
The sum will be calculated by this formula:
sum = value1 + value2 + value3 + value4 + value5
For example, if the first values entered were value 1=1, value 2=1, value 3=2,value 4=2 and value 5=3 respectively:
sum = 1 + 1 + 2 + 2 + 3 = 9

The additional selection statement will be of this form:

```
If sum > 100 then
    print "Sum is over 100"
End If
```

Test Plan

To verify this program is working properly the input values could be used for testing:

Test Case	Input	Expected Output

1	value1=1
	value2=1
	value3=1
	value4=0
	Value5=2
		Sum = 5

2	value=100
	value=100
	value=100
	value=100
	value=200
		Sum = 600
		Sum is over 100.

3	value= -100
	value= -100
	value= -200
	value = 0
	value= 200
		Sum = -200

Pseudocode

```
// This program will calculate the sum of 5 integers.

// Declare variables
Declare value1, value2, value3, value4, value5, sum as Integer

//Initialize Sum to 0
Set sum = 0

// Enter values
   Print “Enter an Integer for value1”
   Input value1
   Print “Enter an Integer for value2”
   Input value2
   Print “Enter an Integer for value3”
   Input value3
   Print “Enter an Integer for value4”
   Input value4
   Print “Enter an Integer for value5”
   Input value5

// Calculate sum
sum = value1 + value2 + value3 + value4 + value5

   // Print results and messages
Print “Sum is “ + sum
If (sum >  100)
       Printf  “Sum is over 100”
End if
```

C Code

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

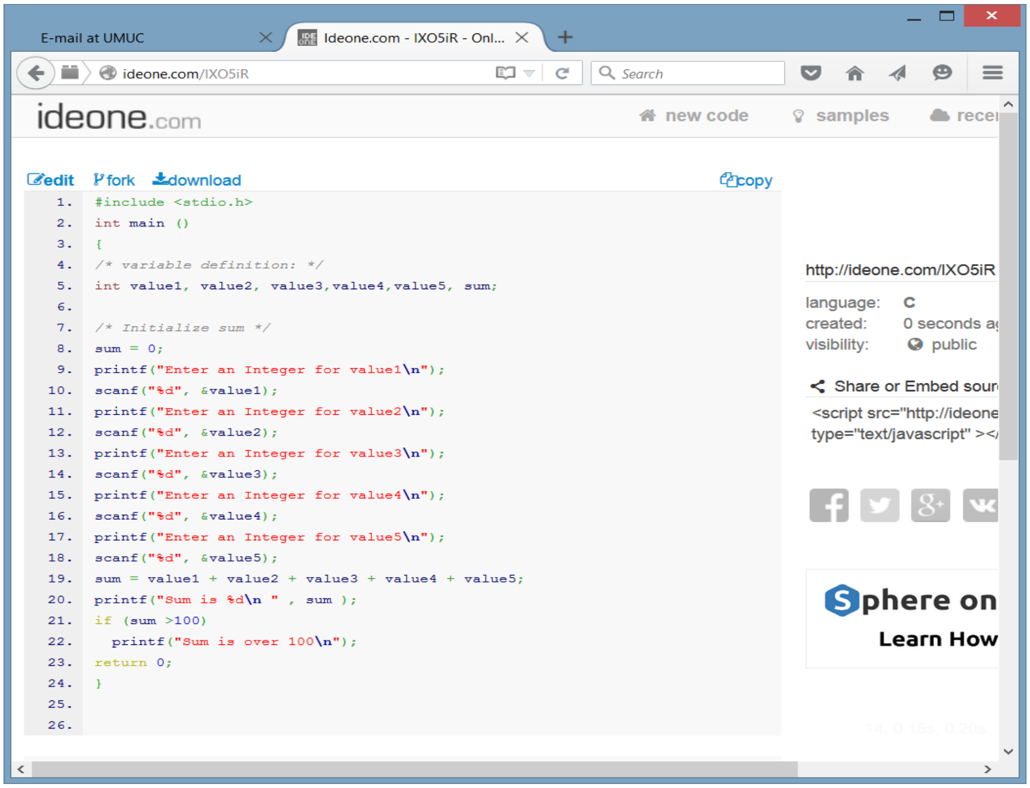
```
// C code

// This program will calculate the sum of 5 integers.

// Developer: Faculty CMIS102

// Date: Jan 31, XXXX

#include <stdio.h>

int main ()
```

```
{

/* variable definition: */

int value1,value2,value3,value4,value5,sum;


/* Initialize sum */

sum = 0;

printf("Enter an Integer for value1\n");
scanf("%d", &value1);
printf("Enter an Integer for value2\n");
scanf("%d", &value2);
printf("Enter an Integer for value3\n");
scanf("%d", &value3);
printf("Enter an Integer for value4\n");
scanf("%d", &value4);
printf("Enter an Integer for value5\n");
scanf("%d", &value5);

sum = value1 + value2 + value3 + value4 + value5;

printf("Sum is %d\n " , sum );

if (sum >100)

  printf("Sum is over 100\n");

return 0;

}
```

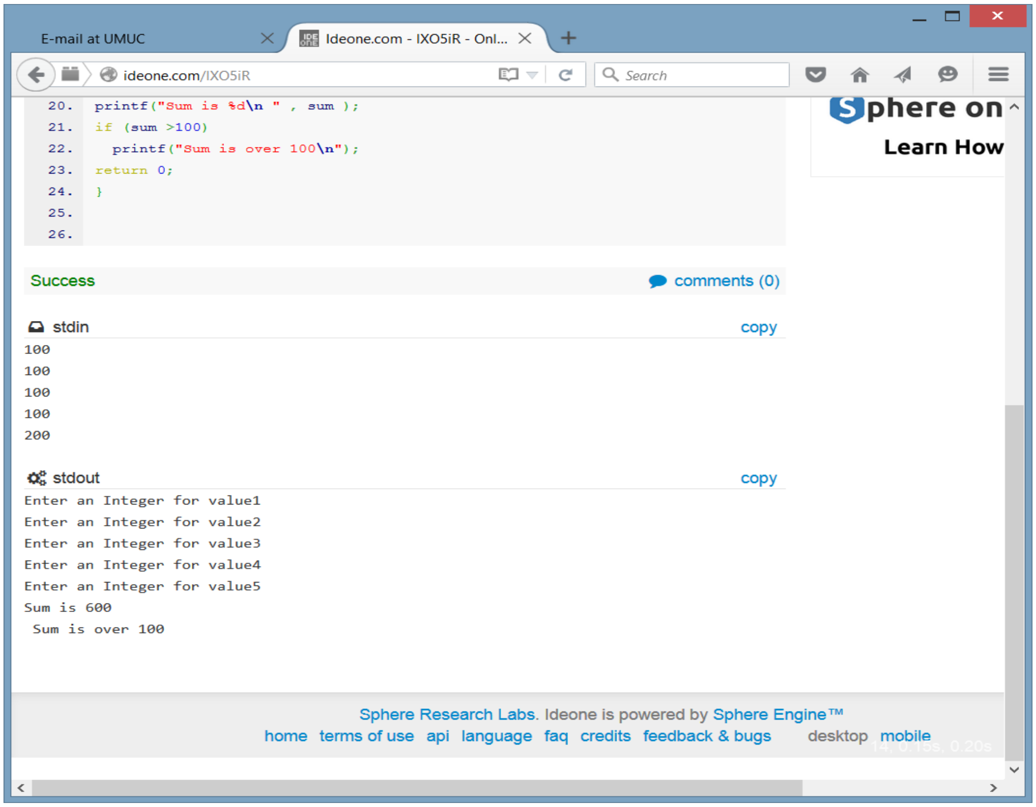
Setting up the code and the input parameters in ideone.com:

Note the input integer values are 100, 100, 100, 200 and 100, for this test case. You can change these values to any valid integer values to match your test cases.

Results from running within ideone



```c
#include <stdio.h>
int main ()
{
/* variable definition: */
int value1, value2, value3,value4,value5, sum;

/* Initialize sum */
sum = 0;
printf("Enter an Integer for value1\n");
scanf("%d", &value1);
printf("Enter an Integer for value2\n");
scanf("%d", &value2);
printf("Enter an Integer for value3\n");
scanf("%d", &value3);
printf("Enter an Integer for value4\n");
scanf("%d", &value4);
printf("Enter an Integer for value5\n");
scanf("%d", &value5);
sum = value1 + value2 + value3 + value4 + value5;
printf("Sum is %d\n " , sum );
if (sum >100)
  printf("Sum is over 100\n");
return 0;
}


```


E-mail at UMUC
Ideone.com - IXO5iR - Onl...
ideone.com/IXO5iR
Search
Sphere on
Learn How

```c
printf("Sum is %d\n " , sum );
if (sum >100)
  printf("Sum is over 100\n");
return 0;
}


```

Success

comments (0)

stdin

copy

```
100
100
100
100
200
```

stdout

copy

```
Enter an Integer for value1
Enter an Integer for value2
Enter an Integer for value3
Enter an Integer for value4
Enter an Integer for value5
Sum is 600
 Sum is over 100
```

Sphere Research Labs. Ideone is powered by Sphere Engine™
home terms of use api language faq credits feedback & bugs desktop mobile

Learning Exercises for you to complete

1. Demonstrate you successfully followed the steps in this lab by preparing screen captures of you running the lab as specified in the Instructions above.

2. Change the C code to sum 10 integers as opposed to 5? (Hint: Please don't use arrays or Loops for this. We will be using those features in another week.) Support your experimentation with a screen capture of executing the new code

3. Using the code you create in step 1, modify the code to print an additional statement if the sum of the value is negative (Hint: Consider modifying the existing selection statement) Support your experimentation with a screen capture 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 code you created after step 2.

5. Create your own C code implementation of **one** of the following mathematical formulas:

    a. y = mx + b; (slope of a line) Assume the user will be prompted to input m, x and b and the program will calculate y. If the value of y is greater than 10, inform the user the value is greater than 10.
    b. a = PI * r*r; (area of circle). Assume the user will be prompted to input the radius r. You can define PI as 3.1416. . If the value of a is greater than 10, inform the user the value is greater than 10.
    c. v = 4/3 PI r*r*r; (volume of sphere) Assume the user will be prompted to input the radius r. You can define PI at 3.1416. If the value of v is greater than 10, inform the user the value is greater than 10.

    Be sure you provide not only the C code but a test table with at least 3 distinct test cases listing input and expected output your mathematical formula.

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. Submit all C code you created in files.

Also, provide the answers and any associated screen captures of your successful completion of exercises 1, 2, 3 and 4.

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

Grading guidelines

Submission	**Points**
Demonstrates the successful execution of this Lab within an online compiler. Provides supporting screen captures.	2
Modifies the C code to sum 10 integers as opposed to 5. Supports your experimentation with screen captures of executing the code.	2
Using the code created in step 1, modifies the code to print an additional statement if the sum of the value is negative Supports your experimentation with a screen capture 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 code you created after step 2.	1
Creates your own unique C code implementation of **one** of the provided mathematical formulas. Provides a new test table with at least 3 distinct test cases listing input and expected output your mathematical formula. Supports your experimentation with a screen capture of executing the new code.	2
Document is well-organized, and contains minimal spelling and grammatical errors.	1
Total	**10**