CIS 150 – Lab 06

**Submission of Your Work**

You need to prepare and submit ONE SINGLE MS Word document to Canvas (in your lab section) as LastName\_FirstName\_Lab07.doc. It must contain:

* Your NAME
* For each question:
  + Specify the question number.
  + After reading the question requirements, but before beginning any coding, create the test case table, below, through column Expected Output. Write your program then complete the **test table** with actual output results and include in your report.
  + Copy/Paste your completed source code. You must include standard “header” in every program even if code is provided.
  + Paste in a snippet of output showing results for **every listed test case**, labeled with test case # ONLY 4 TEST CASES NEEDED FOR EACH QUESTION

Test Table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Expected Output | Actual Output | Test Pass / Fail |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |

* Add / delete rows from Test Table as necessary
* Modify column widths in Test Table as necessary
* Test both valid and invalid input
* Test for every output expected
* If failure is an expected output and it happens then that test Passes
* Any test that fails means the program must be fixed so that it passes the test

**Question 1**

Write a program that calls a function **calculateSum** to calculate the sum from 0 to N integers. The function calculateSum has one parameter N of type integer and returns an integer which represents the sum from 0 to N, inclusive.

Write another function **calculateAverage** that calculates an average. This function will have two parameters: the sum and the number of items. It returns the average (of type float).

The main function should be responsible for all inputs and outputs. Your functions will only calculate and return the values and NOT print anything. N is provided by the user; user input must be asked for repeatedly until a negative integer is provided.

EX: INPUT IS 3, OUTPUT IS 3+2+1+0=6

ANY NEGATIVE # SHOULD BE SIGNAL TO EXIT PROGRAM

**Question 2**

Write a program that prompts users to enter:

* + The character to use (any character besides \*)
  + The width of the display
  + The height of the display

to produce a C shape. This is an example for width of 10 and height of 9

\*\*\*\*\*\*\*\*\*\*

\*

\*

\*

\*

\*

\*

\*

\*\*\*\*\*\*\*\*\*\*

You must use **separate functions to display the vertical and the horizontal lines**.

\*CALL TO HORZ, THEN VERT, THEN HORZ FUNCTION AGAIN.

\*BE SURE TO DO HEIGHT MINUS TWO SINCE TWO HORZ FUNCTIONS ACCOUNT FOR 2 OF THE HIEGHT UNITS

**Question 3 –PASS BY REFERENCE**

Consider the following functions:

void doIt(int &a, int b)

{

a = a + 5;

b = b + 5

}

void doIt2(int a, int& b)

{

a += 5;

b += 5

}

void doIt3(int & a, int& b)

{

a = a + 5;

b += 5

}

Fix any errors in the above functions and write a main program that prompts the user for two integer numbers (num1 and num2). Then, the main program calls the functions doIt, doIt2, and doIt3 with num1 and num2 as parameters. The main program will print the content of num1 and num2 to the screen after each function is called. Add an addendum to the test plan to explain for each function the difference between a and b.

\*addendum = add some paragraphs to the test cases (an additional section) to explain what happens with the a and b for each function call. CAN PICK ONE TEST CASE AND ADD ADENDUM FOR THAT ONE TEST CASE EXPLAINING WHAT IS HAPPENING FOR EACH FUNCTION AND DIFFERENCE BETWEEN A AND B