**ITCS 1212L**

**Pre-Lab 7**

**Objectives:**

* **To work with the while loop**
* **To introduce the do-while loop**
* **To work with the for loop**
* **To work with nested loops**
* **To briefly introduce the concept of files**
* **Work with the debugger**

**Work with Codeblocks Debugger:**

**Pre-Lab 7a:**

Debuggers provide some standard features which you need to learn about them in this prelab. In the Lab7a you need to work with these features in CodeBlocks.

* Breakpoint
* Next Line, Step into, Step out
* Add watch using the following steps:
  + Click the mouse to place it at the start of the line of code you want.
  + Click the Run to Cursor button on the Debugging toolbar.
  + Click the Debugging Windows button.
  + Choose the Watches command.

A sample program is given to you in lab 7a. You need to use the debugger to execute the program and check the values of the variables.

It said we’re doing this as part of the lab.

**Prelab** **TA Check: \_\_\_\_\_\_\_\_\_\_\_\_**

**Lab 7b:**

Create a project named Lab7b in CodeBlocks.

For lab 7b, you need to write a program that will calculate a tip on a meal, until the user decides to quit. The user needs to enter the price of the meal and the percent of the tip for each iteration using ‘while’ loop. For the prelab, you need to work on this problem and have the flowchart ready before the lab.

Process: First requirement for while loops, we need to ensure the first time the loop is encountered our program will meet its condition to run (since the program needs this loop to run). It will probably be this: while (x != -1)

Then we’ll ask the price, calculate the tip, display the tip, etc… The user will be informed that -1 is to exit, otherwise they’ll continue to be prompted to enter meal values.

**Prelab** **TA Check: \_\_\_\_\_\_\_\_\_\_\_\_**

**Prelab 7c:**

Study the following code. You need to run this code in the debug mode to check how the program validates the user’s input (price) that is positive. In the lab7c, you need to test the following program using both negative and positive input.

//This program asks a user for the price of an item. The value must be positive //or the user will have to try again.

int main( )

{

double price;

do

{

cout <<“Enter a positive price for an item: “;

cin >> price;

}while(price <= 0);

//when this is true, the block repeats-when the

// user enters bad data the loop exits when the

// user enters good data because the condition

// becomes false

cout << “Great, you entered a positive value for price!”;

return 0;

}

Add a block comment on top of this code and explain briefly what does this program do.

The program validates the user input for price with the ‘while’ line. It tests the condition (price <= 0). When it happens that the user didn’t follow instructions and enter a positive number, this condition catches the value and forces the user to re-enter it (because it doesn’t exit the do-while loop).

**Prelab TA Check: \_\_\_\_\_\_\_\_\_**

**Prelab 7d:**

In lab7d, you need to write and test a program to validate user input. The user must enter two double values, one greater than 5, one less than zero AND a menu choice of ‘A’, ‘B’ or ‘C’. This requires three do-while loops. The only thing this program does is input validation. Simply print a message after each do-while loop stating the data entered is valid. For prelab, you need to work on the algorithm and flowchart.

Process: This program should be fairly simple. I’ll explain what input is required at the beginning so the user understands what to do. Then I will implement a do-while loop that will ask the user for a value greater than 5, evaluate the value with the condition (value <= 5), and exit the loop or repeat depending on what the user entered.

Then I will make a second do-while loop that will ask the user for a value that is less than zero, evaluate the value with the condition (value >= 0), and exit the loop or repeat depending on what the user entered.

Finally I will place the last do-while loop that will ask the user for a selection from a provided menu (option a, b, and c). The do-while will evaluate the value with the condition (value != a && value != b && value != c), and exit the loop or repeat depending on what the user enetered.

**Prelab TA Check: \_\_\_\_\_\_\_\_\_\_\_**

**Prelab 7e:**

Study the following code that calculates average of 3 tests:

#include<iostream>

using namespace std;

int main( ) {

double testScore;

double total;

double average;

total = 0;

cout << “Enter the score for test 1: “;

cin >> testScore;

total = total + testScore;

cout << “Enter the score for test 2: “;

cin >> testScore;

total = total + testScore;

cout << “Enter the score for test 3: “;

cin >> testScore;

total = total + testScore;

average = total/3;

cout << “The average for the three tests is: “ << average << endl;

return 0;

}

In this lab, you need to change the above code so it will calculate an average and print the new average of several test scores which the number of them is determined by the user. Also, this can be done for any number of students. You need to look at the code example from the textbook that was discussed during the lecture. For the prelab, you need to work on the algorithm and flowchart.

Process: From what I understand, I’ll leave the first part of code functioning so we have a pre-existing average. To calculate more future averages, I’ll need to implement a loop to allow the user to continue entering scores (too add to the total and the interation +3 will divide this value for the average). When the do-while loop encounters a -1, it will let the user exit the loop because they are done entering values.

To allow this for more students, I’ll simply place the entire program within a loop. After -1 is entered for the first student (and any following students), the loop will ask if the user wants to enter scores for another student. If yes, the loop repeats, while counting the iterations and making that the student #.

**Prelab TA Check: \_\_\_\_\_\_\_\_\_\_**

**Prelab 7f:**

In this lab, you need to using a for-loop to write a program to print 10 random integers in the range of 1 to 30. Be sure to look at the output to be sure all the numbers printed are in this range. For the prelab, you need to work on the algorithm and the flowchart to figure out the logic first. Answer the following questions:

* How many times will the for loop run? 10 times (we’ll ensure this with the condition, so that we get precisely 10 random integers).
* What is range of random numbers you will generate ? 0-29 (but we’ll be adding 1).

**Prelab TA Check: \_\_\_\_\_\_\_\_**

**Process: We’ll randomize the time seed.**

**Then we need a for loop**

**For (x = 0; x < 10; x++) //Will iterate 10 times**

**{**

**Number = rand() % 30 + 1;**

**Cout << number << endl;**

**}**

**Return 0;**