IT 1050 – Programming Logic

**Matthew Mosinski**

Midterm

*30 points (extra credit problem +2)*

1. (4 points) Identify and briefly describe 3 types of control structures that we can use to order the statements in our programs?

Sequential control structure- In this control structure, statements are executed line by line. The console executes the program identical to reading, top to bottom, left to right. The system defaults to this control structure.

Selection control structure- Statements typically used in this control structure are *if, if/else, and switch.* The expressions in these statements can evaluate to either true or false. Different actions are taken for true and false conditions and the system has the ability differentiate between the two.

Iteration/repetition control structure- This control structure is typically used for the *for, while, and do/while* loops. The conditions in the loops must be true in order for the console to execute the statement. A *for* loop is used when we have a known number of repetitions in the loop. A *while* loop is used when the number of repetitions is unknown.

1. (4 points) Create an infinite while loop. Use a Boolean variable called keepLooping that set to true in the loop’s termination condition. Hint: Use CTRL+C or Debug -> Terminate All to end the program.

1. (4 points) Write a while loop to prints 2 through 128 in brackets, each on a new line. You should initialize your loop control variable to 2. Output the value of the loop control variable each time through the loop. Use a condition that ends the loop after 128 is printed.

[2]

[4]

[8]

[16]

[32]

[64]

[128]

1. (4 points) Write a for loop that prints 49 through 1 separated by a comma. Note, you will need to use a condition inside of the loops so it does not print the comma the last time through (no newlines).

49, 48, 47, …, 3, 2, 1

1. (4 points) Write a while loop that prints all odd numbers 1 through 21 separated by spaces (no newlines).

1 3 5 7 9 11 13 15 17 19 21

1. (5 points) What is the output for the following code?

int n = 8; int i = 10; // initialize

do {

Console.Write("\*"); i++; // update!

} while (i < n); // test condition

The output of the given code is one asterisk(\*).

What would the output be with the exact same code but using a while statement as opposed to a do-while?

The output is nothing, because *i* is not less than *n*.

1. (5 points) Explain how do we combine multiple Boolean values? Write an if statement that outputs “Let’s go outside!” when both Boolean values are false.

boolean icyRain; boolean tornadoWarning;

1. Extra Credit: Use nested loops to print the following to the console:

123454321

1234321

12321

121

1