# CPSC 1100 – Lab 9

If/Else Statements

This lab will deal with using if statements in java to implement different tasks. The tasks will be assigned from the textbook. The entire lab will be due next week. **PLEASE COMMENT YOUR CODE.** You will have points taken off if you do not comment your code. Keep your code neat.

**Some useful links:**

BlueJ tutorial [www.bluej.org/tutorial/tutorial-201.pdf](http://www.bluej.org/tutorial/tutorial-201.pdf)

Java tutorial home page: <http://docs.oracle.com/javase/tutorial/>

Start here: <http://docs.oracle.com/javase/tutorial/java/index.html>

variables <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>

data types <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>

relational operators <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/op2.html>

if-then <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html>

java math library <http://docs.oracle.com/javase/7/docs/api/java/lang/Math.html>

Simple Video on BlueJ Debugger <http://www.youtube.com/watch?v=LUUPTbWV0g8>

**Some helpful tips:**

1. Compile often – do it.
2. Perform the tasks by hand to verify your work.

### **Tasks: Complete the following tasks.**

Create a new project, and then add a class named “Lab09Tester”. You will need a single method in this class, your main method.

Complete the following tasks. The problems from the book are copied at the end of this document for your reference. I have made slight changes to the problems listed in the book. ***USE MY DESCRIPTION FOR EACH TASK.***

***Task01***: Problem E5.1 from Book. Write a class called ValueAnalyzer. You should have a single int as an instance variable for this class. Create a constructor that accepts an initial value for the instance variable. Create a public mutator method called setValue() that returns void and accepts a new value for the instance variable. Create an accessor method called getValue(). Place a public method in this class called analyzeValue(). The analyzeValue() method should return 1 if the instance variable is positive, 0 if the instance variable is zero, and -1 if the instance variable is negative. Your main method should then print whether the value is negative, zero, or positive (use if statements based on the return value from the analyzeValue() method). Add tests to your main method in the Lab09Tester class that will test ALL branches of your if statements. Include expected values in your main method. (Note that you only need to create one object here, and you can use the setValue() method to change the instance variable for your tests).

***Task02***: Problem E5.2 from Book. Write a class called DoubleAnalyzer that has a single instance variable for the class (a double). Create a 1 argument constructor to set this instance variable to the parameter of the constructor. Create a mutator (setValue()) to change the instance variable. Create an accessor to return the instance variable (getValue()). Finally, create a method analyzeDouble() that accepts no parameters and returns void. Your method should print “zero” if the instance variable is 0, otherwise print “positive” if the instance variable is positive, and print “negative” if the instance variable is negative. Additionally, if the absolute value of the instance variable is less than 1, also print “small”. Finally, if the absolute value of the instance variable is > 1,000,000 you should also print “large”. You need to add sufficient tests to your main method to reach all branches of your if statements. (Again, you can create a single DoubleAnalyzer object, and use the mutator method to change the value before you make calls to analyzeDouble()).

***Task03***: Problem E5.3 from the Book. Write a class called DigitPrinter that has a single instance variable stored as an int. Create a constructor to accept an initial value for the instance variable. Create an accessor (getValue()) and a mutator (setValue()) for this instance variable. Create a method printDigits() that accepts no parameters and returns void. Your method should print how many digits the instance variable has by checking if the variable is >=10, >=100, and so on. You may assume that your instance variable will be less than 10 million. (You do not need to test values > 10 million or < -10 million). You will need to find a solution to deal with negative instance variables. (for example, -101 is three digits, but -101 is NOT >=100). You need to add sufficient tests to your main method to reach all branches of your if statements.

***Task04***: Problem P5.16 from the Book. Write a class called DigitalThermometer that has a single instance variable stored as a double that represents ***Vm***. Create a one argument constructor to accept an initial value for this instance variable. Create an accessor (getVoltage()) and mutator (setVoltage()) for this variable. Create a method calculateTemp(). Your method should print the temperature based on the problem description given in the text book. Note that if ***Vm*** is out of range then you should print an error message. You need to add sufficient tests to your main method to reach all branches of your if statements. (See the photo at the end of this document for more information about this problem).

***Task05***: Get a capture of your final output, which should show output for Task01, Task02, Task03, and Task04 (Make sure you have both expected and actual values printed as appropriate).

## To Turn In via Google Drive

You should turn in your java files and a document containing your output.



