Name: _____   UIN: _____

This lab can be submitted during lab on Wednesday 11/26/14 or during lecture on Monday 12/1/14.

This lab uses links to web pages found at: https://www.cs.uic.edu/bin/view/CS211/LabexercisesF14

Testing is when we determine whether a program executes correctly. I.E. Does it perform as described in the program specification? (Note that debugging is trying to find the causes of failed tests and how to correct the issues. Debugging is different from testing.) In this lab, we will focus on testing.

Program Specification for a Simple Tax Calculation:

Create a web form that will allow the user to calculate the amount of taxes owed to the government. The user should enter the income amount in an input field, then press/click a button and the amount of taxes owed is to be displayed. If the user enters a non-numeric income amount or a negative income amount, display an error message telling the user that a positive numeric value must be entered. The amount of taxes owed is determined by:

- *If the income amount is $5,000 or less, the tax amount is 10% of the income amount.*
- *If the income amount is more than $5,000 and is $50,000 or less, the tax amount is 15% of the income amount.*
- *If the income amount is more that $50,000, the tax amount is 20% of the income amount.*

You will find 3 attempts at completing the above specification at:
- Attempt 1: http://tigger.uic.edu/~troy/tax1.html
- Attempt 2: http://tigger.uic.edu/~troy/tax2.html
- Attempt 3: http://tigger.uic.edu/~troy/tax3.html

Test cases describe the actions/inputs to be given and the expected results. Multiple test cases are needed for each action/input. These test cases should include inputs which result in all possible error messages generated by the program.

Test cases should include inputs from all possible ranges of inputs (including error inputs). Each of these possible ranges of input is often called an **Equivalence Class**. For the above specification, one Equivalence Class is the values greater than $5,000 up to and including $50,000. Another Equivalence Class is the non-numeric values. For each Equivalence Class, a test case using a value from the "middle" of the range of inputs is to be created.

Test cases should also include inputs that verify the proper functioning of the program for values at the boundary between two Equivalence Classes. These are often called **Boundary Cases**. Each boundary should generate multiple test cases: a test for the boundary value, and a test for each value on either side of the boundary value. For the above specification, one boundary case is the value of $5,000. So we should have a test case that uses the value of $5,000, another with the value of $4,999 and one more with the value of $5,001.

For this lab, create multiple test cases to fully test the three attempts at the above specification. Each test case should specify a specific input value, the expected result, whether the test case is testing an Equivalence Class or a Boundary Case, and the result (pass or fail) for each of the three attempts. A table is provided below for you to fill in. A minimum of 8 test cases must be created (at least 4 Equivalence Class tests and 4 Boundary Case tests – the first one has been started for you). These test cases should thoroughly test the entire specification, so you may need more than 8 test cases. (Hint: you need at least 8 test cases to properly do this and it can be done in 16 or less test cases.)

Test #	Exact Input Value for Test	Exact Expected Result	Equivalence or Boundary	Result from Attempt 1 (pass/fail)	Result from Attempt 2 (pass/fail)	Result from Attempt 3 (pass/fail)
1	$10000	$1500	Equivalence Class Test			
2						
3						
4						
5						
6						
7						
8						