# ISIT 324 Homework 2

**30 Points Possible**

1. Given your solution to the Module 1 Homework, question 3:
   1. You may discover a variety of ambiguities in the assignment. In other words, ample opportunities for defects exist. Describe as many possible defects as you can. *Submit either a text file or a Word doc.* **(8 points)**
   2. Create a set of test cases that you think would have a reasonable chance of revealing the faults you identified above and document the rationale in your test set as inline comments. (Note: you don’t have to use a test framework for this. Just put your test code in Main() and call the Union method for each test. Print “test passed” or “test failed” for each test. *Submit a .cs file including the Main() method with tests and the Union method.* **(12 points)**
   3. (recommended) Run your tests with your implementation to ensure you get the expected results.
2. Given the following code (note that the program is written in Java and doesn’t follow C# conventions), answer the questions below. You might note that it’s very similar to question 2 in the Module 1 assignment. **(10 points)**

|  |
| --- |
| /\*\* |
| \* Count positive elements |
| \* |
| \* @param x array to search |
| \* @return count of positive elements in x |
| \* @throws NullPointerException if x is null |
| \*/ |
| public int countPositive (int[] x) |
| { |
| int count = 0; |
| for (int i=0; i < x.length; i++) |
| { |
| if (x[i] >= 0) |
| { |
| count++; |
| } |
| } |
| return count; |
| } |
| // test: x = [-4, 2, 0, 2]; Expected = 2 |

* 1. Explain what is wrong with this code. Describe the fault precisely by proposing a modification to the code.
  2. Give a test case that does not execute the fault and explain why it doesn’t.
  3. Give a test case that executes the fault but does **not** result in a failure.
  4. Implement your fix to the code in C# and call it from Main() with your test cases to demonstrate that the fault is corrected. Then submit your .cs file.