# Fixing Broken Output

**The goal for this exercise** is to make sure that you can get a basic unit test to pass. This unit test will use I/O redirection to capture output, and/or set the input to your code.

In the starter project, you should find a class of tests named Basic\_IO\_Redirection. (Notice that while these tests are in the same project/file as the others, this exercise expects you to find them on your own)( Edit 🡪Find and Replace 🡪 Find In Files may be useful here, if you forget where the tests are). Your task for this exercise is to make all the tests in this class pass. There are (at least) two tests in this class.

Starting with this exercise, you should NOT modify the test. If you think that there’s an error in the test, you should post something to the Google Group, and/or notify the instructor.

In the project named something like 03\_PCE\_StudentCode, in the file named Student\_Answers.cs, you should find a class named Calculator\_Class.

In that class, the outline of the ‘PrintNumbers’ method is provided, but you’ll have to fill in most of it. The PrintNumbers method must print out all the integers between 1 and 10, inclusive. Each number should be printed on it’s own line, like so:

1

2

3

Etc.

In that class, the outline of the ‘AskUser’ method is provided, but you’ll have to fill in most of it. The AskUser method must ask the user for an integer, then ask the user for another integer, and then print out a message if the first number is larger, smaller, or equal to the second number. Here’s some example output (user input is indicated in **bold underline**)

Type in a number:

**1**

Type in a number:

**2**

The first number is smaller than the second.

Another transcript:

Type in a number:

**3**

Type in a number:

**2**

The first number is larger than the second.

Another transcript:

Type in a number:

**3**

Type in a number:

**3**

The first number is equal to the second.

Because this output is being auto-graded, you need to match the above output *exactly*. While the ‘fuzzy’ string comparison will be used, you still need to make sure that the contents of each line are the same.

You’re free to change the methods in Calculator\_Class however you like; you should NOT change the test.

**What you need to do for this exercise:**

1. Implement the PrintNumbers method in the Calculator\_Class, as described above.
2. Implement the AskUser method in the Calculator\_Class, as described above.
3. Make sure that all the tests in the Basic\_IO\_Redirection class pass.