A cursory examination of the source code showed a major missing component for any refactoring: unit testing. Testing provides a means through which we can confirm that any refactoring done to the project does not affect the overall functionality and usability of the project. By providing a test suite to run against the various methods in the project, we can quickly and efficiently narrow down any errors in the code.

Given the expansive, naïve nature of the source code, we created tests with visions of how we would like the code to be organized, rather than tests that would work with the current source code. As a result, we created several new objects and methods that we would like to see in the refactored version of the code. As a result, however, this test suite cannot be compiled until these new objects and methods are written.

To begin, we created methods for InitializeBoards(), SetupBoards(), Fire(), and CheckWin().

Next, we refactored the code to use a multi-dimensional 10x10 array of Booleans, rather than the array of integers currently being used. This refactoring improves the performance and execution time of the code and better models the real-world example of a Battleship game.

A Message variable, which would store output to the console, was then created.

With these three major changes, the test suite can now compile. Running the tests should produce the failed outputs shown below. We purposefully fail these tests in an effort to model the “red, green, refactor” method of refactoring and test writing.

Add methods for InitializeBoards, SetupBoards, Fire, and CheckWin

Create 2-D 10x10 array of bool’s for Board1 and Board2

Add a Message variable that prints to console

References

<https://www.cprogramming.com/cgi-bin/source/source.cgi?action=Category&CID=2>

<https://msdn.microsoft.com/en-us/library/jj620919.aspx>

<https://msdn.microsoft.com/en-us/library/hh598953.aspx>

<http://www.geeksforgeeks.org/multidimensional-arrays-c-cpp/>

<https://answers.yahoo.com/question/index?qid=20110315131034AA08bPe>