1. Re-read the document called “Software Testing” in the course handbook. Be sure that you understand it thoroughly — your testing on the project will be marked to this high standard.

2. Download class AnotherTwoEndedList from the web site. (It’s another implementation of interface LinkedTwoEndedList, but it has an additional method called append(), as well as a toString().)

3. Devise a systematic and thorough testing strategy for method append. Describe your strategy in a table below. Be sure to number the test cases, and to explain the significance of each test case.

4. Write a new class called TELTester whose sole purpose is to test method append. (You will probably use the other methods in the class to help set up your test cases. You may assume that these other methods work correctly.) Write the code so that it produces meaningful output that is easy to read.

5. Run TELTester. If you find some test cases crash, you should comment them out so that you can execute the later ones. Put a check-mark or ex beside each case in your table to indicate whether it worked properly.

6. Print the output from TELTester and annotate it by hand to
   • cross-reference your output to the test cases in your table, and
   • point out important things to notice.

Bring to tutorial: this handout with your testing strategy, a printout of your TELTester class, and your annotated test output.