# 204 Labs 1A & 1B

You will need to complete the following this week –

* Junit Lab (instructions provided in this document)
* Exception Lab (provided PDF file)
* Submit a compressed (Zip) file and include the following:
  + Small write-up (several paragraphs) highlighting your learning experience
    - What did you learn?
    - What issues did you encounter, if any?
    - What would you have done differently?
    - How can you apply this concept in the future?
    - Anything else you would like to share?
    - Screenshot(s) of your GitHub repo
    - Screenshot(s) of your Junit test runs
    - Screenshot(s) of your Exception test runs
  + Updated GradeBook.java (file)
  + Exception file
  + Junit Test file

# JUnit Lab

For this lab, you will need to –

* Implement (add) the following methods to the provided **Gradebook.java** –
  1. **getScoreSize()** method which returns scoresSize
  2. **toString()** method which returns a string with each score in **scores** separated by a space
  3. **minimum()** method – see Javadoc comment
* Create a new Junit Test Class, **GradebookTester**.java, by –

1. Select the setUp and tearDown method.
2. Select all of the methods of Gradebook, except for the constructor to create tests for.
3. In the setUp method of GradebookTester, create at least two objects of Gradebook of size 5. Call the addScore method for each of the Gradebook classes at least twice (but no more than 5 times).
4. In the teardown method of GradebookTester, set the two objects of Gradebook to null;
5. Create test for the methods of Gradebook:
   1. addScore
      1. Use the toString method to compare the contents of what is in the scores array vs. what is expected to be in the scores array

assertTrue( . . .)

* + 1. Compare the scoreSize to the expected number of scores entered.

assertEquals(. . .)

* 1. sum
     1. Compare what is returned by sum() to the expected sum of the scores entered.
  2. minimum
     1. Compare what is returned by minimum() to the expected minimum of the scores entered.
  3. finalScore
     1. Compare what is returned by finalScore() to the expected finalscore of the scores entered.

The finalScore is the sum of the scores, with the lowest score dropped if there are at least two scores, or 0 if there are no scores.

For example:

As a private member of GradeBookTest:

GradeBook g1;

In setup:

g1 = new GradeBook(5);

g1.addScore(50);

g1.addScore(75);

In teardown:

g1 = null;

in testSum():

assertEquals(125, g1.sum(), .0001);

in testMinimum():

assertEquals(50, g1.minimum(), .001);

in addScoreTest();

assertTrue(g1.toString().equals(“50.0 75.0 ”);