CS 149: Programming Fundamentals James Madison University Coding Review Exam #2 Spring 2018 *This work complies with the JMU Honor Code. I have neither given nor received unauthorized assistance, and I will not discuss the exam contents with anyone who has not taken it for credit.*

Name:_____ Signature:_____

Grade Calculator

When calculating grades, teachers often want to know what the class average is and often give awards or assess penalties for early and late assignments. You will be performing both of those functions in your code today.

For the purpose of the exam, you must write one classes, GradeCalculator and complete the test for a second template provided class GradeCalculatorTest. The first class will implement two methods and the second class should use JUnit to test the code in the first class thoroughly. At the end of (and during) the exam, you will submit these two files via https://autolab.cs.jmu.edu . You may submit as many times as you like; only the last submission will be graded.

The two methods, **average** and **newGrades**, must be implemented in the class GradeCalculator. The **average** method will take a single double array parameter called **grades** and will return a double

value that is the average of all of the numbers in grades. A null **grades** should return -1.

The **newGrades** method will take two array parameters, a double array of **grades** and a character array indicating **time** and will return a double array of adjusted grades based on whether the student indicated by the array index turned the assignment in early, on-time, or late. *Your method should return null, for null array input.* The student is indicated by the index in the array. Index 0 indicates student zero in both the grades and time arrays, index 1 indicates student 1, etc. Scores should be adjusted and returned in a new array as follows:

Students who turned the assignment in early ('e' in time array) should get +5 pts.
Students who turned the assignment in on-time ('o' in time array) should not have grade changed.
Students who turned the assignment in late ('l' in time array) should get -10 pts.

You must write complete documentation comments for both classes (including your name and today's date) and all methods (including @param and @return tags). As with previous assignments, the code you submit must pass Checkstyle without any errors. It's strongly recommended that you complete the exam in this order:

1. Write documentation and stubs for all classes and methods. [10 min] 2. Implement your test cases; they should all fail at this point. [10 min] 3. Submit to Autolab to make sure that it compiles and runs. [5 min] 4. Implement/test **average**; submit to Autolab. [10 min]

5. Implement/test **newGrades**; submit to Autolab. [15 min]
You may use the examples on the back of this sheet for reference. Good luck!

Below is an example test file template with ***key pieces (and javadoc) missing.***

```
import org.junit.Assert;
import org.junit.Test;

public class GradeCalculatorTest {

@Test public static void testGradeCalculator() {

}

@Test public void testAverage() {

Assert.assertEquals("TODO: test failure message ", expected, actual);

}

@Test public void testNewGrades() {

}

}
```

The command in UNIX to zip files to submit to Autolab:
zip Exam2.zip GradeCalculator.java GradeCalculatorTest.java

Autolab url: https://autolabcs.jmu.edu