Extra Credit

- **Problem 3.10, page 46**

Write a class that supports course grading; it contains an array of 30 students such that each student is identified by an integer and has a course grade. There is also a method that can change the student grade and a method that computes the grade point average for the entire class of 30 students.

Answer:

```
function CourseGrader(studentArray) {
        if (!studentArray || studentArray.length !== 30) {
                throw "Only 30 students in class allowed."
        }
        this.students = studentArray;
}
CourseGrader.prototype.changeGrade = function(student, grade) {
         this.students[student] = grade;
}
CourseGrader.prototype.gpaAverage = function() {
        return this.students.reduce(function(a, b) {
                return a + b;
         }) / this.students.length;
}
```

- **Problem 10.6, page 166**

For the following program design a minimal test suite:

```
         public static void main(String args[]) {
            int i,j;
            int k = 0;
            i = Integer.parseInt(args[0]);
            j = Integer.parseInt(args[1]);
            if( i > 30 ) {
               if( i <= 60 && j <= 150 )
                  k = 1;
               else if( i <= 90 && j <= 150 )
                  k = 2;
               else
                  k = 3;
```

```
            }
         if( i == j && i <= 30 )
            k=4;
         System.out.println( k );
      }
```

Answer:
Here is the test method that I created!

```
public void testMain() {
        assert( main(["50", "100") == "1" );
        assert( main(["80", "100"]) == "2" );
        assert( main(["100", "100"]) == "3" );
        assert( main(["20", "20"]) == "4" );
}
```

- **Problem 10.8, page 167**

In test driven development, a test is written and then the code to pass it. Given the following test, write a method that passes the test.

```
      public class TestGrade {
         Grade testGrade;
         public void testGetFinalGrade() {
         assert( testGrade.getFinalGrade(70) == "Pass" );
         assert( testGrade.getFinalGrade(69) == "Fail" );
      };
```

Answer:
Here is my getFinalGrade function that I created:

```
public String getFinalGrade(int grade) {
        if (grade >= 70) {
                return "Pass";
         } else {
                return "Fail";
        }
}
```

- **Problem 10.9, page 167**

Expand the test in exercise 10.8 to include grades A to F.

Letter Grade	Minimum Percentage
A	90
B	80
C	70
D	60
F	Less than 60

Answer:

```
public class TestGrade {
        Grade testGrade;

        public String getFinalGrade(int grade) {
                if (grade >= 60) {
                        return "D"
                } else if (grade >= 70) {
                        return "C"
                } else if (grade >= 80) {
                        return "B"
                } else if (grade >= 90) {
                        return "A"
                } else {
                        return "F"
                }
        }

        public void testGetFinalGrade() {
                assert( testGrade.getFinalGrade(90) == "A" );
                assert( testGrade.getFinalGrade(80) == "B" );
                assert( testGrade.getFinalGrade(70) == "C" );
                assert( testGrade.getFinalGrade(60) == "D" );
                assert( testGrade.getFinalGrade(59) == "F" );
        }
};
```