Review Lab #3

As the material in this lab is intended as a review, it might be useful to click ‘first year CS Topics’ link on the homepage of our course (**located in Canvas**).

**Polynomials**

Objects are one of the most difficult to understand topics, and their usefulness can take some time to sink in – we'd better start now. Object oriented programming is a big part of AP computer science! A (very) brief refresher:

***A class is a custom type*** *that you define – a bundle of variables and methods. In addition, a class is a blueprint for objects. Objects have all the fields and methods defined in the class, but with actual values.*

*An object is a specific* ***instance*** *of the class – one where all the declared variables have values.*

Polynomials are expressions that appear in *many* areas of math and science; a polynomial is a series of coefficients, variables, and non-negative exponents (e.g. x2 – 4x + 8). In this lab, we will only be considering single variable quadratic polynomials.

1. **Create a new project in BlueJ called ReviewLab3.** Make a new class called [Monomial](https://en.wikipedia.org/wiki/Monomial). The Monomial class represents a *single term* in a Polynomial (e.g. x2 could be a Monomial object). This class has the following:
   1. Private Instance variables:
      * private double coefficient – the red portion of this term: **4.97**x2
      * private int exponent – the red portion of this term: 4.97x**2**
   2. Methods
      * public Monomial(double coeff, int exp) – constructor, initializes instance variables to given values
      * public double getCoefficient()
      * public int getExponent()
2. Make another class called [Polynomial](https://en.wikipedia.org/wiki/Polynomial). The Polynomial class will represent a quadratic polynomial (e.g. x2 + 2x – 7); in other words, a Polynomial is comprised of three Monomials.
   1. Private Instance variables
      * Monomial firstTerm – first term in the Polynomial, e.g. x2
      * Monomial secondTerm – second term in the Polynomial, e.g. 3.17x
      * Monomial thirdTerm – third term in the Polynomial, e.g. 20.12
   2. Methods
      * public Polynomial() – constructor, initializes all the Monomials objects to null.
      * Appropriate accessor methods
      * public void setTerm(double coeff, int exp) – the parameters of this method supply the values for ***initializing*** the corresponding Monomial instance variable.

If exp == 2 the method will initialize firstTerm (the x2 term), if exp == 1 it will initialize secondTerm, exp == 0 will be thirdTerm.

/\* When a Polynomial object is created, the Monomial objects *firstTerm*, *secondTerm*, and *thirdTerm* are **declared only** (i.e. null). In this method, don't forget to **instantiate** them (set their values to a new Monomial object with *coeff* and *exp* passed in to the constructor) \*/

* + - public double evaluate(double x) – evaluates the polynomial for the given value of x (plug in the parameter value x for all occurrences of x in the polynomial).

/\* example: for a polynomial f(x) = x2 + 2x – 7, evaluating f(5) would result in 52 + 2\*5 – 7, or 28 \*/

* + - public String toString() – should return a String representation of this Polynomial in this format (don't worry about the signs, e.g. "+ -3" is fine):

3.5x^2 + 0.6x + 13.6

Using the Polynomial and Monomial classes you wrote, solve the following problems in a class called Runner with a main() method.

1. Copy the following into the Runner class to test your code. Fix any mistakes you may have.

Polynomial a = new Polynomial();

a.setTerm(1.0, 2); a.setTerm(2.0, 1); a.setTerm(3.0, 0);

System.out.println(a.evaluate(4.5)); //should print: 32.25

System.out.println(a.toString()); //should print: 1.0x^2 + 2.0x + 3.0

If you get NullPointerException when you run your code, this means that you are attempting to call a method on variable that stores a null reference. In other words, that variable hasn't been initialized to store a reference to an object. Isolate the offending variable and initialize it.

1. Evaluate the polynomial f(x) = 4.32x2 – 2.12x + 3.67 for f(2).
2. Suppose the concentration (in parts per million) of a new medicine in the bloodstream after t hours is given by the polynomial -0.05t2 + 2t + 2. Write the code to print a "table" that shows the concentrations each hour for 8 hours, to ensure proper dosing.
3. If a projectile is fired vertically into the air, its position at any time is given by the following ballistic trajectory formula (i.e. polynomial): y = y0 + v0\*t – 1/2\*G\*t2 where y0 is the initial launch height, v0 is the initial launch velocity, and G is the gravitational acceleration constant.

If a projectile is launched at 50 m/s from an initial height of 3m, find the time (calculated in increments of one second) when the projectile passes a height of 100m.

**When you're done, save your work to your H://AP\_CS folder.**