Homework 3

Problem 2:

a) We have chosen the array representation of a polynomial: RatNum[] coeffs, where coeffs[i] stores the coefficient of the term of exponent i. An alternative data representation is the list-of-terms representation: List terms, where each Term object stores the term's RatNum coefficient and integer exponent. The beauty of the ADT methodology is that we can switch from one representation to the other without affecting the clients of our RatPoly. Briefly list the advantages and disadvantages of the array representation versus the list-of-terms representation.

The advantages of the array representation would be allowing us to quickly access data using indexing. Arrays can be indexed while the List<Term> strategy would have to use iterators to comb down the list until they found the variable they desire. For the task at hand, all larger methods would slow down significantly since we have to loop through rather than using a find function or method.

The advantages of using Lists would be the separation of exponents and coefficients. If you run into a non-zero exponent followed by a coefficient of zero, it can be really easy to lump the two numbers into one when using arrays. This can cause several issues in computing the correct answer. Separation of numbers may be an issue when coding this way if you had used arrays.

b) Where did you include calls to checkRep() in RatPoly (at the beginning of methods, the end of methods, the beginning of constructors, the end of constructors, some combination)? Why?

In RatPoly, I only included calls to checkRep() at the end of the Constructors. This is being checkRep() prevents RatPoly from creating bad objects and throws any bad construction calls. The class RatPoly creates an immutable object, so checkRep() only needs to see if the object is made correctly since any other method cannot cause RatPoly to be a bad object.