Homework 1

The goal of this homework assignment is to write a Python class to represent rational polynomials. Everything must be in the file `polynomial.py`. By rational polynomials, I mean the ratio of two polynomials, each of which have integer coefficients. In general, the polynomials could have rational coefficients, but this is equivalent to integer coefficients. First you will write a `Polynomial` class; the `RationalPolynomial` class will then contain a numerator and denominator, which are both instances of `Polynomial`.

Here's a list of features you need to implement for the `Polynomial` class:

- Attributes of `Polynomial`: the order of the polynomial, and the coefficients. You should store the coefficients using a `numpy` array of integers, e.g.,

  ```
  coefficients=np.zeros(order+1,dtype=int)
  coefficients[order] = 1
  ```

  makes the polynomial x^n where n is the order. Strictly speaking, you don't need the order to be an attribute, because it is related to the length of the coefficients array, but it might be convenient to store it as its own varible.

- A `staticmethod` called `from_string` which allows the user to define a polynomial from a string. This should be able to make sense of strings of the form `"-1 - 7*x^4 + 3*x^2"`. *Hint:* First split the equation up into monomial terms, and then extract the coefficients from each monomial.

- A `__repr__` function which gives a useful visual representation of the polynomial.

- Methods for addition, subtraction, multiplication, and equality.

- Dividing two polynomials (`__truediv__`) should return a `RationalPolynomial`.

The `hw1_tests1.py` script contains the `pytest` tests I will use to grade the `Polynomial` class.

Once you can represent polynomials, you should write a `RationalPolynomial` class. This class should look a lot like the `Fraction` class I discussed in lecture. Features for the `RationalPolynomial` class:

- It should store a numerator and denominator, each of which are `Polynomials`; in `Fraction` they were both integers.

- It should have a `staticmethod` called `from_string`. The string used to specify a `RationalPolynomial` will have parentheses around the number and denominator, and these will be separated by a `/` character. E.g., `"(2 + x)/(-1 + x + 2*x^3)"`.

- When you initialize a `RationalPolynomial`, you should check if there are any common factors between the numerator and denominator. *Hint:* It is probably easier to find a package that will do this for you, rather than coding up an algorithm to do this yourself.

- Methods for addition, subtraction, multiplication, division, and equality.

I also encourage you to write a `__repr__` function, which will be useful for debugging.