

Assignment 9
Assigned: April 25th 2016
Due: May 6th 2016

Prerequisite:

- Object Oriented Programming in C++
- Basic understanding of Polynomials

In this assignment, you will implement a Polynomial class called Poly. Each polynomial will be represented as a Linked List.

For example, the polynomial $x^3 + 2x^2 + 3x$ is represented by the following Linked List

1 → 2 → 3 → 0.

The polynomial $x^3 + 3x$ is represented by the following Linked List

1 → 0 → 3 → 0.

Notice that the head of the Linked List is the term in the polynomial with the highest degree. Also note that the polynomials do not need to be integers.

You are provided with four starter files: main.cpp, polynomial.hh and polynomial.cpp and functions.cpp

In the polynomial.hh file you will need to create the class structure for the Poly class (i.e. Constructors, destructors and so on...). Note that the file polynomial.hh already includes functions you need to implement. Feel free to add functions to your class.

In the polynomial.cpp file you will need to implement the functions in your Poly class.

The main.cpp file contains some test cases for you to use and test your program.

In the functions.cpp file you will implement three functions as follows:

1. Poly addPolynomials(Poly p1, Poly p2)
This function takes two Poly objects as input and returns the sum of the polynomials as a Poly object as well.
2. Poly subtractPolynomials (Poly p1, Poly p2)
This function takes two Poly objects as input and returns the difference (p1 - p2) of the polynomials as a Poly object as well.
After subtracting, if the first node value is zero you need to delete it.
3. Poly multiplyPolynomials(Poly p1, Poly p2)
This function takes two Poly objects as input and returns the product (p1*p2) of the polynomials as a Poly object as well.

Some of the files provided already include code to get you started.

The `polynomials.hh` contains the structure for the `Node` class and for the `Poly` class. Feel free to add functions to these classes.

Do not change any code given to you.

Hint:

- Think about what happens when you do `Poly result = addPolynomials (p1, p2)`