# CS 310 Programming Assignment #1

# **Representing and Manipulating Polynomials**

# Due: Wednesday, February 16 12:00 PM (noon)

A polynomial is the sum of a finite number of terms of the form  $c * x^n$  where c is called the coefficient and n is the exponent. For this assignment, you have to represent and manipulate **polynomials** represented as **linked lists**.

More precisely, for this assignment, you have to write code for

- 1. A *queue* class implemented using a linked list (see Section 4.4.1 of the textbook)
- 2. An *extended queue* class that is derived from the *queue* class (see Section 4.4.2 of the textbook)
- 3. A *polynomial* class that is implemented in terms of an *extended queue* (see Section 4.5 of the textbook)

You will be provided with the interface and some of the member functions for the *queue, extended queue*, and *polynomial* classes. Your task is to write the missing methods for these classes. Specifically, you have to

- Write the following methods for the *queue* class:
   (a) *empty* (b) *retrieve* (c) the destructor (d) the copy constructor (e) the overloaded assignment operator
- Write the following methods for the *extended queue* class:
  (a) *full* (b) *clear* (c) *serve and retrieve*
- 3. Write the following methods for the *polynomial* class:
  (a) read (b) print (c) equals\_sum (d) equals\_product (e) mult\_term (f) evaluate

For the *read* and *print* methods of the *polynomial* class, you can use the *read* and *print* methods provided in the textbook as a starting point. However, these methods should be modified to read input and print output according to the formats described below. The *evaluate* method of the *polynomial* class computes the value of the polynomial for a given value of *x*.

# Input

Your program should read its input data from a file name **poly.dat**. The data in this file corresponds to a number of polynomials and a number of *x* values. The format of this file is as follows:

Line 1: *n r* 

where n = the number of polynomials and r = the number of x values

#### Lines 2 to *n*+1: the polynomials.

The values for each polynomial will be entered on a single line as coefficient, exponent pairs. The list will be terminated by a 0. For example, the polynomial  $3x^4 + 4x - 16$  is represented in the input as

 $3\ 4\ 4\ 1\ -16\ 0\ 0$ 

Thus, there is one pair of integers for each term in the polynomial. Further, the terms of the polynomial are listed in order of decreasing exponents.

## Line *n* + 2: *x* values

This line will contain r integers in the range (1 - 10) each representing a value for x. Notice that there are no restrictions on n and r except that they are positive.

#### Example input file

The first line in the input specifies that there are 4 polynomials and 3 x values that you have to read. The polynomials in the above input are  $-9x^3 - 4x^2$ , x + 2,  $5x^3 + 5x^2 + 5x$ , and 2. There are 3 values of x for which you have to evaluate each polynomial: 3, 4, and 2.

## Output

The output for this assignment will be the sum of the input polynomials (as a polynomial), the product of the input polynomials (as a polynomial), and then the values of each input polynomial for the given values of x.

The output for the input given above would be:

```
Sum: -4x**3 + x**2 + 6x + 4

Product: -90x**7 - 310x**6 - 390x**5 - 300x**4 - 80x**3

For x = 3

-279 5 195 2

For x = 4

-640 6 420 2

For x = 2

-88 4 70 2
```

Notice that \*\* is used to represent exponentiation in your output. Be sure to consider the special cases for the polynomial output (i.e. exponent = 1 or 0, coefficient = 1, 0 or a negative number).

## Error Checking

You should detect:

- 1. Negative exponents in the input.
- 2. If the exponents of the terms in an input line are not in decreasing order.
- 3. Out of range x values on the last line of input (i.e., not between 1 and 10 inclusive).

Print an error message and skip over input lines that have the first two errors above. In the case of the third type of input error, do not evaluate the polynomial for that particular input.

## Submitting your Assignment

The source code for this assignment should be submitted electronically as Assignment #1. Remember to turn in <u>all</u> <u>the files</u> (including header files) that are needed to compile and run your program. The external documentation should be turned into your instructor (or the TA) on or before the due date. Remember that we do not accept late submissions. See online notes about electronic submission and documentation of your code. On the class web page, you can also see the grading criteria that will be used for all the assignments in this class.

## Assignment Web Page

The source code for the interfaces and member functions of the *queue*, *extended queue*, and *polynomial* classes is available at the URL <u>http://mason.gmu.edu/~cs310</u>