Problem Solving Challenge 11

Develop abstract data type to represent polynomials, including both data elements, member functions, friend functions, overloaded operators, and dynamic arrays. This class must be able to represent polynomials of any degree (use a dynamic array). It must include friend operators to read and print, member function to evaluate polynomials and operator overloaded member functions for +, -, and =.

Discussion:

A polynomial has the form:

$P(x) = a_n x^n + a_{n-1} x^{n-1} + ... + a_1 x^1 + a_0$

The a_i's are fixed floating point numbers. x is a variable that is the input when evaluating the polynomial. We say that the degree of a polynomial is the highest power of x present. The degree of the above polynomial is n.

Examples

$P_1(x) = 1x^2 + -5x + 6$

$P_1(0) = 6$

$P_1(2) = 0$

$P_1(10) = 56$

$P_2(x) = 3x^4 + -4x^3 + 0x^2 + 1x + -5$

$P_2(0) = -5$

$P_2(2) = 13$

$P_3(x) = 1x^9 + -2x^8 + 3x^7 + -4x^6 + 5x^5 + -6x^4 + 7x^3 + -8x^2 + 9x -10$

$P_3(1) = -5$

$P_3(0) = -10$

$P_3(-1) = -55$

Polynomial Data Members

A polynomial class needs private data members to represent the polynomial. The class must store both the degree (an integer) and the coefficients (a floating point array).

Polynomial Operations

There are many operations that a computer might do on polynomials. These include, evaluating, adding, subtracting, multiplying, finding roots, calculating derivatives, calculating integrals, graphing and many more. We will consider four basic operations, reading in a polynomial, printing out the polynomial, evaluating the polynomial at a given value of x, and adding polynomials. Don’t forget you will also need constructor, destructor, and copy constructor functions and an assignment operator.

Polynomial Input

A class function to read in the polynomial should read in the degree and then read in degree+1 coefficients. A prompt should be printed for each number read in. The prompt should include the power associated with the current term. In other words, if the function is reading in the coefficient for the x^3 term the prompt should clearly state that it is the x^3 term. Hint: Store the polynomial coefficients into the array such that the array index indicates the power of x (i.e. constant should go into index 0).

Printing Polynomials

Unfortunately, C++ makes it difficult to print superscripts to our computer monitors. Hence, the class function to print polynomials needs another way to represent the powers. We choose to use ^ to represent the exponentiation operator. Therefore,

$3x^4 + -4x^3 + x + -5$ becomes 3x^4 + -4x^3 + 0x^2 + 1x + -5. If you desire, this could be printed as 3x^4 + -4x^3 + 0x^2 + 1x^1 + -5x^0 instead, but this is slightly less readable.

Evaluating Polynomials

Evaluating polynomials can be done in several different ways. A naive algorithm to evaluate polynomials would just calculate each power of x, multiply in the coefficients and add up the terms.

Polynomial
- degree:int - coefficients:double*

+Polynomial():
+Polynomial(size: int):
+~Polynomial():
+Evaluate(xValue:double):double
+Read():void
+Print():void