

Department of Electrical and Computer Engineering University of Puerto Rico Mayagüez Campus

ICOM 4035 – Data Structures Fall 2001

Project #1: Algebra and Calculus on Polynomials Due Date: 12:00 PM-October 4

Objectives

- 1. Understand the design, implementation and use of the vector class container.
- 2. Gain experience implementing abstract data types using already developed data structures.
- 3. Gain experience with object-oriented programming concepts, especially copy constructors and operator overloading

Overview

You will implement and test a polynomial class, using the vector class container as the data structure to store the terms in a polynomial. The general form of a polynomial is as follows:

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

Here, each term has a coefficient, denoted as a_i , and a exponent *i*, which represent the power of the variable x. Your polynomial class must implement the following operations:

- 1. Addition given two polynomials P_1 and P_2 , compute the polynomial $P_3 = P_1 + P_2$.
- 2. Subtraction given two polynomials P_1 and P_2 , compute the polynomial $P_3 = P_1 P_2$.
- 3. Multiplication given two polynomials P_1 and P_2 , compute the polynomial $P_3 = P_1 * P_2$.
- 4. Derivative given a polynomial P, finds its derivative.
- 5. Indefinite integral given a polynomial P, finds its indefinite integral (anti-derivative).
- 6. Definite integral given a polynomial P, evaluate its definite integral over an interval [a,b].
- 7. Degree given a polynomial P, find its degree (the largest exponent in any term).
- 8. Evaluate given a polynomial P, evaluate it at value x, to compute y = P(x).

You must implement the vector class container, which is slightly more robust that the version discussed in class. The vector container will store the terms in the polynomial, in decreasing order of exponent. Thus, each element in the vector represents a term in the polynomial. For example, if we need to represent the following polynomial: $3x^2 + 2x + 1$, then the organization of the vector container associated with the polynomial class should look like this:



The dotted lines are meant to convey the fact that each element is a structure with two fields: the coefficient of the term and the exponent to which the variable x should be raised. In your implementation, you cannot store terms containing a coefficient equal to zero. The only exception is the case in which the polynomial correspond to P(x) = 0, meaning that the polynomial is just the number 0. When you implement your mathematical operations you must make sure you don't add terms to the polynomial that are zero. Again, the only exception is when the resulting polynomial is the value 0.

To clarify, this point consider the following expression: (2x + 1) - (2x - 2). In this case, the resulting polynomial will be 3, and the representation will be:



As you can see, the terms with variable x cancel out, and there is no need to represent 0x in the polynomial. Likewise, there is no need to represent the term corresponding to xraised to the first power in this polynomial: $2x^2 + 1$



In this case, the term corresponding to ax is not represented since the coefficient is zero.

You implementation will consist of adding C++ code to implement two modules: vector.cc and polynomial.cc. You will receive all the .h files necessary for this project. In addition, you will be provided with a main program that uses the polynomial class, and interacts with the user to ask his/her input on the operations and polynomials to be evaluated. Finally, you will be given a Makefile with all the commands needed to compile your project. In summary, you program will consist of the following files:

- 1. Object.h declaration of the term data type and the object data type used by the vector class.
- 2. vector.h interface for the vector class container.
- 3. vector.cc implementation of the vector class container. YOU MUST IMPLEMENT THE METHODS THAT APPEAR IN THIS FILE.
- 4. polynomial.h interface for the polynomial class.
- 5. polynomial.cc implementation of the polynomial class. YOU MUST IMPLEMENT THE METHODS THAT APPEAR IN THIS FILE.
- 6. polymain.cc menu-driven main program that interacts with user to perform operations on polynomials.

You can go to the class web page and download a tar fle containing all these files. Just access the link named Projects, and down load the sources files associated with the link: Project #1 - Polynomials.

PROJECT DUE DATE: 12:00 PM – October 4, 2001.