Object Oriented Programming
Spring 2023
Assignment#4

Submission date: Thursday April 6, 2023

Redo the assignment 3 using the concept of composition and operator overloading. Make a class term that has two members: coefficient and exponent. The class polynomial must contain a dynamic array of type term and an integer to store the number of terms.

Define the following functions as members of class polynomial.

Constructors: Define default, copy and overloaded constructors for the class polynomial.

Overload stream extraction operator (>>): first ask the number of terms in the polynomial and then dynamically allocate the array of terms to store exponent and coefficient of each term using the operator new. Next, for each term input the exponent and coefficient of each term. Note to store the polynomial in decreasing order of exponent. Also exponents must be unique. For example `3x^3 + 4x + 5 + 3x must be stored as 3x^3 + 7x + 5.` Overload the operator that allows cascaded function calls

Overload stream insertion operator (<<): This function must output the polynomial in the standard format. Overload the operator that allows cascaded function calls

Overload Add operator (+): This function must take a polynomial as parameter and compute and return the sum of polynomials (caller polynomial and the one passed as parameter).

Overload multiply operator (*): This function must take a polynomial as parameter and compute and return the product of polynomials (caller polynomial and the one passed as parameter).

Overload evaluate operator (()): This function must take an integer ***val*** as parameter and evaluate the polynomial where the variable has value ***val***. This function must return the value of the polynomial. For example, for a polynomial p = `3x^3 + 7x + 5, p(2) must return 43.`

Destructor: Define the destructor for the class polynomial.

Assignment operator: Overload the assignment operator for the class polynomial that uses deep copy.

Constraints:

- The exponents are non-negative whole numbers.
- The coefficients can be positive/negative real numbers.

The size of dynamically allocated arrays must be exactly equal to the number of terms in the polynomial. There must not be any logical errors (memory leak and dangling pointers)

Important Note:
Submit your code files at google classroom as one single file. If you have more than one files compress them and submit them as single file. The name of file must your complete roll number. Your code must be properly commented.
This is an individual assignment and the code submitted must be your own contribution. Any sort of plagiarism will be dealt seriously and may lead to severe consequences including negative marking.