

Ngoc Nguyen NITK17K - 1700062

Mid-term Project

Question

Write a C++ program to multiply and then evaluate two polynomials of an arbitrary degree provided by the user. Up to five variables are possible in each polynomial, ex. $(10x^{10})(15.5y^3)(1.76z^7)(34p^2)(15f^4)+(4y^4)(4p)$.

You are allowed to use power and sqrt functions from C++ standard library along with all input/output libraries (iostream and like so). Other functions/methods are required to be implemented by you. You must follow the object-oriented pattern provided by C++ (using the concept of classes and information hiding at least).

Your program should have an adequate user interface (text or graphical), which helps the user to understand and conveniently provides the inputs and understand the output. Employing a graphical user interface is not required, but if implemented, it will get a bonus. The originality of the source code is also of high importance, and plagiarism detection software may apply. After submitting your project, we will have an online or in-person meeting to get more information on the process of developing your program and the level of your programming skills.

Github link

<https://github.com/NgocNguyenGit/cpp-programming>

Internal logic of the program

Polynomial skeleton

If we have a full, and simplified polynomial as the following:

$$-4x^2z + 3$$

Then, it is possibly to have this table:

Polynomial	Term	Element		
$-4x^2z + 3$	$-4x^2z$	-4	Coefficient	-4
			Variable	Not found
			Exponent	Not found
		x^2	Coefficient	1
			Variable	x
			Exponent	2
		z	Coefficient	1
			Variable	z
			Exponent	1
	+3	+3	Coefficient	+3
			Variable	Not found
			Exponent	Not found

Object-oriented pattern

This project follows the object-oriented pattern provided by C++ (using the concept of classes and information hiding at least).

Element

In *Element*, the constructors were created to support these cases:

Polynomial		Description
p1	-4	Polynomial contains only constant (or coefficient)
p2	x^2	Polynomial contains variable name and exponent
p3	5y	Polynomial contains coefficient and variable name
p4	$3z^2$	Polynomial contains coefficient, variable name, and exponent

Regarding Encapsulation in C++, coefficient, variable name, and exponent are supposed to be read or modify by class *Polynomial*; therefore, I provide public get methods.

```
//get function - values to be used in polynomial
int Element::getCoeff () const {
    return e_coeff;
}
char Element::getVar () const {
    return e_var;
}
int Element::getExp () const {
    return e_exp;
}
```

Term

Vector container was applied here because of its function, "Just like arrays, vectors use contiguous storage locations for their elements, which means that their elements can also be accessed using offsets on regular pointers to its elements, and just as efficiently as in arrays. But unlike arrays, their size can change dynamically, with their storage being handled automatically by the container."¹

In *Term*, these actions should be taken:

- Parsing terms from polynomial inputted by the user, then extracting elements from each term;
- Storing blocks of elements in each term (under vector container);
- Multiplying two terms by multiplying each element from each term to another;
- A get function was created to be accessed and used later in *Polynomial* class;
- Evaluating result from each term (multiplication equation).

Polynomial

Most of the core actions are taken in *Polynomial* class.

Constructors here support these ideas:

- Prompting for input polynomial from the user, then parsing it to *Term* in order to extract elements;
- Creating a vector to store polynomial;

¹ <http://www.cplusplus.com/reference/vector/vector/>

- Splitting polynomial into lower hierarchy elements by inserting a special character, e.g. ";";
- Storing pieces of elements to *Term* vector
- Multiplying two polynomial;
- Printing out the result;
- Evaluating the polynomial by the input values to variables (addition equation).

Main

Because of the requirement of classes and information hiding at least, in *Main* file simply includes the sentinel method for user to interact with the program as long as he/she would like to exit by entering the sentinel number.

There are three actions can be taken and shown in the interface:

- Format rule
- Polynomial Calculator
- Quit

The rule of this program should be strictly followed by the user; otherwise, the program will run and return the unexpected result. The format of a standard, and simplified polynomial is supposed to be inputted. It means that there is only one integer coefficient (constant) in each term and followed by variables. Variables are supposed to be alphabetical characters and be ordered ascendingly- the requirement is for aesthetic purpose only and does not affect the result of equation. The program limits the input's exponent within the range from 0 to 10 exclusively. It is unacceptable to have spaces between characters from the input.

Polynomial calculator is supposed to allow users to input two polynomials, then multiply them and prompt the user to input values for each variable to evaluate the multiplication's polynomial.

Text user interface

A screenshot of example program run is enclosed below.

```

1 - Format rule
2 - Polynomial Calculator
3 - Quit
Enter action: 1

Format | (coefficient)(variable name)^(exponent)
Rules  | Terms must be SIMPLIFIED
      | Each term includes coefficient — (integer, max. 1 value)
      | variable names — (alphabetical character, ascending order)
      | exponents according to each variable — (integer, 0 <= value < 10)
      | NO SPACE ACCEPTED
Example| ACCEPT -4x^2y^3z+2x NOT -4x^2y^3zx^6 NOR -4x^2 + 3
Note   | Values for evaluation phase can be in DOUBLE type

1 - Format rule
2 - Polynomial Calculator
3 - Quit
Enter action: 2

***** Data Input Phase *****
p1: -4x^2z+3
p2: y^3

***** Multiplication Phase *****
p1 * p2 = p3: -4x^2zy^3+3y^3

***** Evaluation Phase *****
x: 1
z: -2
y: 3.5
Result: 471.625

1 - Format rule
2 - Polynomial Calculator
3 - Quit
Enter action: |

```

Deficiency of the program

- This program cannot function properly with exponents greater than 10 (more than one digit).
- The result of the multiplication cannot be simplified e.g., $2xy+5xy$ cannot be $7xy$, nor reorder the variables in alphabetically e.g., $2yx$ cannot be $2xy$.
- If any spaces between characters are detected, the program will run and return unexpected result.
- The author cannot create a graphic user interface because of the shortage of expertise.

Disclaimer

The program is created in XCode with MacOS; therefore, it will have some unexpected performance if be opened in other operating system or IDE.