

HOMEWORK 1

Assignment

Make a `Polynomial` class with *integer* coefficients. This will store polynomials p of the form

$$P(x) = a_0 + a_1x + a_2x^2 + \dots a_kx^k,$$

where $a = (a_0, a_1, \dots, a_k)$ is a **vector** of `ints`¹. For each function *declared* above `main`, define the function below `main`.

In `main`, your program should ask the user to enter in the coefficients of two polynomials by inputting one `int` value at a time until an input of `-1234` (yes I chose this value arbitrarily, but you must use it. A value like this is called a *sentinel value*). You may assume the user will **ONLY** input integers separated by spaces or return carriages, i.e., do NOT error check, simply assume the user will input values according to the specified manner. Your `main` should then add, subtract, and multiply these polynomials according to the sample output on page 3.

```
/** @class Polynomial

...(put in your own brief description)
*/
class Polynomial {

public:
    Polynomial();
    Polynomial(const vector<int>& coeffs);

    // Accessors
    int Degree() const;
    int Coefficient(int k) const;
    void print() const;

    // Mutators
    void constantMultiply(int x);
    void Transform();

private:
    vector<int> coefficient;

}
```

¹Please do NOT work with a `vector` of `double` for this assignment. I repeat, do NOT use `double` variables.

```
// poly1 + poly2
Polynomial Add(const Polynomial& poly1, const Polynomial& poly2);

// poly1 - poly2
Polynomial Subtract(const Polynomial& poly1, const Polynomial& poly2);

// poly1 * poly2
Polynomial Multiply(const Polynomial& poly1, const Polynomial& poly2);
```

Here is a brief description of what each function should do:

- `Polynomial()` is the default constructor, and should initialize the polynomial to 0 by default (this means initializing the vector private variable to have one element with value 0).
- `Polynomial(const vector<int>& coeffs)` is the constructor with parameters that sets the private variable coefficient to the input parameter coeffs.
- `Degree()` returns the degree of the polynomial (see definition below).
- `Coefficient(int k)` returns the coefficient of x^k .
- `print()` prints out the polynomial in the simplest form (doesn't have to be pretty, explanation below).
- `constantMultiply(int x)` multiplies the polynomial by an integer constant, x .
- `Transform()` Transforms the polynomial, the exact description is given below.

Notation and terms:

- The *degree* of a polynomial is the largest power of x^k with a non-zero coefficient. For example, the degree of $1 - x + x^2 + 3x^6$ is 6, and the degree of 34 is 0. For this assignment, 0 will have degree 0 (even though mathematically this is not an ideal interpretation).
- The k -th coefficient, or the coefficient of x^k is the numerical value multiplied by x^k . For example, the coefficient of x^2 in $1 - x + x^2 + 3x^6$ is 1, and the coefficient of x^5 is 0.
- The Transform function is defined as follows. For a polynomial

$$P(x) = a_0 + a_1x + a_2x^2 + \dots a_kx^k,$$

The Transform of $P(x)$ is given by the formula

$$a_1 + 2a_2x + 3a_3x^2 + \dots ka_kx^{k-1}.$$

In the case when $P(x) = a_0$, the Transform is 0. This includes the case when $a_0 = 0$.

Here are some examples of the Transform function:

- $P(x) = 1 + x + x^2 + x^3 + x^4$, then the transform is $1 + 2x + 3x^2 + 4x^3$.
- $P(x) = 1 - 2x + 34x^5 + x^{87}$, then the transform is $-2 + 170x^4 + 87x^{86}$
- $P(x) = 0$, then the transform is 0.
- $P(x) = x^4$, then the transform is $4x^3$.
- $P(x) = 5$, then the transform is 0.
- $P(x) = -x^2 - x^3 + 5x^4$, then the transform is $-2x - 3x^2 + 20x^3$.

- Once constantMultiply and Add are defined, the Subtract function is very easy.
- The print function should suppress all 0 coefficients, but do not worry about testing for positive or negative. Also the coefficient of x^1 should be printed as just x . Examples:
 - $P(x) = 1 + x$ should print as $1+x$
 - $P(x) = 1 - x^2$ should print as $1+-x^2$
 - $P(x) = 3x - 17x^2 + 5x^5$ should print as $3x+-17x^2+5x^5$.
 - $P(x) = -5 - x - 12x^2 + x^{100}$ should print as $-5+-x+-12x^2+x^{100}$
- You may add as many helper functions as you like, but you must have all of the functions listed on this assignment to receive full credit.
- In general, you may NOT assume that the last entry in your vector of coefficients is nonzero unless you include code that checks for zero coefficients. For example, if p is the polynomial $1+2x$ and q is the polynomial $3-2x$, then the result of Add(p,q) would give 4, which might include an extra 0 at the end of your vector of coefficients. This is a hint to let you know that the Degree function should NOT just return the size of the vector, but should find the largest nonzero coefficient. (Many students in the past have successfully written a private member function called CleanUp that tests for any trailing zeros and pops them off of the vector.)

Place your code in a source file labeled *hw1.cpp*. ***If your file is not named this exactly, your homework will not be collected.*** As with all programs in this course, your code should contain useful comments. In particular, your name, the date, and a brief description of what the program does should appear at the top of your source file.

What to Turn in

Place in your Submit folder the source file *hw1.cpp* with exactly this name (all lowercase, no spaces). The files will be automatically collected on Monday 4/7/14 at 5:00pm.

Grading		
Correctness	No errors, input/output correct, output presented nicely	5 points
Arithmetic	Correctly performs polynomial arithmetic	5 points
Solution	Code is efficient but easy to follow	5 points
Style	Variable names, comments, indentation	5 points
	TOTAL	20 points

Note on grading: There is an automatic 5 point penalty for any homework that does not compile.

It is sufficient to use cin and cout for this assignment, there is no need to use getline().

A *sample* of output is below. The text in black is what should display verbatim on the console every time. The **blue text represents values input by the user**. The **green text is**

output that will depend on the particular input values. *These colors are only for easier explanation in this document. It is impossible to change the color of text on the console.*

NOTE! The values input are just an example, but *they must follow this exact procedure.* Your code will be run using values from a file (I have a program set up for this), so do not anticipate a human being reacting to your code, it must be set up to run automatically in the format below.

Welcome! Please input the coefficients of the first polynomial.
When you are finished, enter -1234.

1
-2
0
4
-1234

Your first polynomial is $1-2x+4x^3$.
Its transform is $-2+12x^2$.

Please input the coefficients of the second polynomial.

0
-1
5
0
0
0
-3
-1234

Your second polynomial is $-x+5x^2-3x^6$.
Its transform is $-1+10x-18x^5$.

The sum of these polynomials is

$1-3x+5x^2+4x^3-3x^6$

The first minus the second is

$1-x-5x^2+4x^3+3x^6$

The first multiplied by the second is

$-x+7x^2-10x^3-4x^4+20x^5-3x^6+6x^7-12x^9$

Thanks for using my program!

Press any key to continue...

Frequently asked questions:

(1) Q: Do we have to use `getline`?

A: It is sufficient to use `cin` and `cout` for this assignment, there is no need to use `getline`.

(2) Q: Does the greeting and spacing have to be verbatim as well?

A: The black text must print out verbatim, including spaces, return carriages, and all text.

(3) Q: If we have polynomial $1 + x - x^2$. What should `Coefficient(3)` return?

A: The coefficient should return 0, as it should if `Coefficient` is called with any number larger than 3 or less than 0.

(4) Q: What if the user inputs only -1234 and no other coefficients?

A: Then assume the polynomial is 0.

(5) Q: What if the user inputs all 0s?

A: Then the polynomial should be 0.

(6) Q: Can we make the print function nicer, i.e., can we do MORE work so that the instead of outputting `1+-x+-2x^2` it prints out `1-x-2x^2`?

A: Sure, but it is not required and you won't get any extra credit for doing so.