Homework Assignments #1

Due by 9:20AM, Sept 30, Friday

1. Goals

This assignment aims to further understanding of some basic C++ concepts we've discussed in class, which include C++ class, constructor, destructor, overloading operators, etc.

2. Problem (20 points)

A polynomial of degree *n* has the form $a_0 + a_1x + a_2x^2 + ... + a_nx^n$, where $a_0, a_1,..., a_n$ are integer constants called the **coefficients** of the polynomial and $a_n \neq 0$. For example, $1 + 3x - 7x^3 + 5x^4$ is a polynomial of degree 4 with integer coefficients 1, 3, 0, -7, and 5. One common implementation of a polynomial stores the degree of the polynomial and the list of coefficients. The declaration of the polynomial class **Poly** is declared below. You are required to use a static array to implement the class. You must write a **client program** *test.cpp* to test your class using the functions/operations you have implemented.

The header file *poly.h* is as follows:

```
//poly.h
#ifndef POLY H
#define POLY H
#include <iostream>
using namespace std;
const int CAPACITY = 100;
class Poly {
private:
     int degree; // degree
     int coefs[CAPACITY]; //coefficient array
public:
     Poly(); //constructor
     ~Poly(); //destructor
     void display(ostream& out) const;
     void input(istream& in);
     int evaluate(int x);
     Poly operator+(const Poly& p);
     Poly operator*(const Poly& p);
};
ostream& operator<<(ostream& out, const Poly& p);</pre>
istream& operator>>(istream& in, Poly& p);
#endif
```

3. <u>Requirements</u>

- You must create four files: *poly.h, poly.cpp, test.cpp* and *Makefile.*
- The C++ file *poly.cpp* should implement all the functions specified below:
 - Overload input **operator** >>, which calls member function **input()**, allowing users to input degree and coefficients.
 - Overload output **operator** <<, which calls the member function **display()**, displaying the polynomial.
 - Implement **evaluate()** that takes an argument **x** and calculates the value of the polynomial for this **x**.
 - Overload operator +.
 - Overload operator *.
- The client program *test.cpp* must test all the functions you have implemented. You are strongly encouraged to test your program comprehensively in order to receive full credits.

4. Submission (Due by 9:20AM, Sept 30, Friday)

Before submission, you should ensure your program has been compiled and tested, hopefully extensively. Your assignment receives zero if your code cannot be compiled and executed. Collaboration is prohibited!

You are encouraged to include a readme file in your submission, which should state your program's purposes, assumptions, and issues (if any).

You can submit your program multiple times before the deadline. The last submission will be used for grading.

To submit your assignment, you should follow the steps below:

- a. Wrap all your files into a package, named *hw1.tar*
 - tar -cvf hw1.tar poly.h poly.cpp test.cpp Makefile readme
- b. Submit your newly generated package *hw1.tar* /home/fac/testzhuy/CPSC250/SubmitHW hw1 hw1.tar