```
#include <vector>
#include <algorithm>
#include <iostream>
#include <stdlib.h>

using namespace std;
class Polynomial;
class Term
{
          friend Polynomial;
private:
          float coef;
          int exp;
};
class Polynomial {
public:
          Polynomial();
          //construct the polynomial p(x) = 0;
          Polynomial Add(Polynomial b);
          //return the sum of the polynomials *this and b
          //Polynomial Mult(Polynomial b);
          //return the product of the polynomials *this and b
          //float Eval(float f);
          //evaluate the polynomial *this at f and return the result
          void NewTerm(const float theCoeff, const int theExp);
          int Display();
          int GetData();
private:
          static Term* termArray;
          static int capacity;
          static int free;
          int start, finish;
          int terms;
};
Polynomial::Polynomial()
{
          start = -1;
          finish = -1;
          terms = 0;
}
int Polynomial::Display() {
          int aPos = start;
          for (; aPos <= finish; aPos++) {
                    cout << termArray[aPos].coef << "x^" << termArray[aPos].exp;
                    if ((aPos - finish) != 0)
                              cout << " + ";
          }
          cout << "Wn";
          return 0;
}
void Polynomial::NewTerm(const float theCoeff, const int theExp)
{
          if (terms == capacity)
          {
                    capacity *= 2;
                    Term* temp = new Term[capacity];
                    copy(termArray, termArray + free, temp);
                    delete[] termArray;
                    termArray = temp;
          }
          termArray[free].coef = theCoeff;
          termArray[free++].exp = theExp;
}
int Polynomial::GetData() {
          int i, degree;
```

```
        float coef;
        int expo;
        cout << "Enter Degree Of Polynomial:" ;
        cin >> degree;
        start = free;
        for (i = degree; i > 0; i--) {
                cout << "Enter coefficient of x^" << i << ":";
                cin >> coef;
                cout << "Enter exponent of x^" << i << ":";
                cin >> expo;
                NewTerm(coef, expo);
        }
        finish = free - 1;
        terms = finish - start + 1;
        return 0;
}
Polynomial Polynomial::Add(Polynomial b)
{
        Polynomial c;
        int aPos = start, bPos = b.start;
        c.start = free;
        while ((aPos <= finish) && (bPos <= b.finish))
                if ((termArray[aPos].exp == b.termArray[bPos].exp))
                {
                        float t = termArray[aPos].coef + b.termArray[bPos].coef;
                        if (t) c.NewTerm(t, termArray[aPos].exp);
                        aPos++; bPos++;
                }
                else if ((termArray[aPos].exp < b.termArray[bPos].exp))
                {
                        c.NewTerm(b.termArray[bPos].coef, b.termArray[bPos].exp);
                        bPos++;
                }
                else
                {
                        c.NewTerm(termArray[aPos].coef, termArray[aPos].exp);
                        aPos++;
                }
        for (; aPos < finish; aPos++)
                c.NewTerm(termArray[aPos].coef, termArray[aPos].exp);
        for (; bPos < b.finish; bPos++)
                c.NewTerm(b.termArray[bPos].coef, b.termArray[bPos].exp);
        c.finish = free - 1;
        return c;
}
int Polynomial::capacity = 100;
Term* Polynomial::termArray = new Term[100];
int Polynomial::free = 0;
int main(void) {
        int choice;
        Polynomial P1, P2, P3, P4;
        cout << "Instruction:- ₩nExample:-₩nP(x)=5x^3+3x^1₩nEnter the Polynomial
likeWnP(x)=5x^3+0x^2+3x^1+0x^0₩n";
        cout << "Enter Polynomial1:-" << endl;
        //P1.GetData();
        cin >> P1;
        cout << "Enter Polynomial2:-" << endl;
        //P2.GetData();
        cin >> P2;
        cout << "Enter Polynomial3:-" << endl;
        cin >> P3;
        while (1) {
                cout << "₩n****** Menu Selection ******" << endl;
                cout << "1: Addition₩n2: Substraction₩n3: Multiplication₩n0: Exit" << endl;
                cout << "Enter your choice:";
```

```
                cin >> choice;
                switch (choice) {
                case 1:
                        cout << "₩n--------------- Addition ---------------₩n";
                        cout << "Polynomial1:";
                        //P1.Display();
                        cout << P1;
                        cout << "Polynomial2:";
                        //P2.Display();
                        cout << P2;
                        cout << p3;
                        //P3 = P1.Add(P2);
                        P4 = P1 + P2 + P3;
                        //P3.Display();
                        cout << P4;
                        cout << "----------------------------------------₩n";
                        break;
                case 2:
                        cout << "₩n------------- Substraction -------------₩n";
                        cout << "Polynomial1:";
                        //P1.Display();
                        cout << P1;
                        cout << "Polynomial2:";
                        //P2.Display();
                        cout << P2;
                        //P3.Substract(P1, P2); 학생 구현 실습 대상
                        P4 = P1 - P2;
                        cout << P4;
                        cout << "----------------------------------------₩n";
                        break;
                case 3:
                        cout << "₩n----------- Multiplication -------------₩n";
                        cout << "Polynomial1:";
                        //P1.Display();
                        cout << P1;
                        cout << "Polynomial2:";
                        //P2.Display();
                        cout << p2
                                //P3.Multiply(P1, P2);
                                P3 = P1 * P2;
                        cout << "----------------------------------------₩n";
                        break;
                        case 4: //P2.Eval(5); 학생 구현 실습 대상
                                cout << P4;
                                P2.Eval(3);
                                break;
                case 0:
                        cout << "Good Bye...!!!" << endl;
                        exit(0);
                }
        }
        system("pause");
        return 0;
}
```