HW3 실습

polynomial

연산자 오버로딩 (operator overloading)

- 기본 존재하던 연산자들을 직접 사용자가 정의하여 연산자를 Class에 맞게 직접 설계 후 사용하는것

```cpp
#include <iostream>
using namespace std;

class Point {
private :
    int x, y;

public :
    Point(int x_, int y_) {
        x = x_;
        y = y_;
    }

    void print() {
        cout << "x : " << x << ", y : " << y << "\n";
    }
};


int main(void) {
    Point p1 = { 1, 1 };
    Point p2(2, 2);

    Point p3 = p1 + p2;

    p3.print();

    return 0;
}
```



```cpp
#include <iostream>
using namespace std;

class Point {
private :
    int x, y;

public :
    Point(int x_, int y_) {
        x = x_;
        y = y_;
    }

    void print() {
        cout << "x : " << x << ", y : " << y << "\n";
    }

    Point operator + (Point& p) {
        x = x + p.x;
        y = y + p.y;
        return Point(x, y);
    }
};


int main(void) {
    Point p1 = { 1, 1 };
    Point p2(2, 2);

    Point p3 = p1 + p2;

    p3.print();

    return 0;
}
```

Sort algorithm

- C++의 algorithm 헤더에 포함
- 기본적으로 오름차순 정렬 수행

- Compare함수를 만들어 해당 함수의 반환 값에 맞게 정렬이 동작하도록 설정 가능

```
#include<iostream>
#include<algorithm>
using namespace std;
void Print(int *arr){
    cout << "arr[i] : " ;
    for(int i=0; i<10; i++){
        cout << arr[i] << " ";
    }
    cout << endl;
}

bool compare(int a, int b){
        return a > b;
}


int main(void){
    int arr[10] = {3, 7, 2, 4, 1, 0, 9, 8, 5, 6};
    int arr2[10] = {3, 7, 2, 4, 1, 0, 9, 8, 5, 6};
    Print(arr); //정렬 전 출력


    sort(arr, arr+10); // default(오름차순)로 정렬
    Print(arr); //정렬 후 출력

    sort(arr2, arr2+10, compare); // compare(내림차순)로 정렬
    Print(arr2); //정렬 후 출력
    return 0;
}
```

실습 설명

- 기술서에 적혀있는 대로 polya.cpp 나 polyb.cpp의 비어있는 부분을 작성할 것
- 몇몇 함수의 경우 교재에 있으니 참고해서 작성할 것
- 컴파일 방법: make hw3_ (a나 b)

실습 설명

```
ostream& operator<< (ostream& os, Polynomial& p) {
   .....
   return os;
}
void Polynomial::NewTerm(const float theCoeff, const int theExp)
{
   .....
}
Polynomial Polynomial::operator+(Polynomial& b)
{
   .....
}
Polynomial Polynomial::operator*(Polynomial& b)
{
   Polynomial c;.
   .....
   return c;
}
```

```
[B611107@localhost hw2]$ hw2a < poly.in
x^8 -7x^5 -x^3 -3
x^5 +2x^3 -4
x^8 -6x^5 +x^3 -7
```

질문

- pemta81718@gmail.com
- 간단한 구글링으로 알 수 있는 내용은 답변하지 않습니다.