

# LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

## BÀI TẬP TUẦN 3

2023 – 2024

Bài tập thực hành Operator Overloading and Static Member.

**Bài 1** Viết chương trình đếm số đối tượng tạo ra của một lớp, Ví dụ lớp Hình chữ nhật.

Biết rằng: Tập tin giao diện của lớp Hình chữ nhật định nghĩa như sau.

```
/* *****  
 * Các định nghĩa của lớp Rectangle  
***** */  
class Rectangle {  
private:  
    double length;  
    double height;  
    static int count;    //Static data member  
public:  
    Rectangle(double lenght, double heigth);  
    Rectangle();  
    ~Rectangle();  
    Rectangle(const Rectangle& rect);  
    static int getCount();    //Static member function  
    // other functions  
    void print();  
  
    // Override Extraction '>>' and Insertion '<<' Operator  
    // Definition of << operator  
    friend ostream& operator<<(ostream& out, const Rectangle& rec);  
    // Definition of >> operator  
    friend istream& operator>>(istream& in, Rectangle& rec);  
  
};  
// Initialization of static data member  
int Rectangle::count = 0;
```

**Bài 2** Viết chương trình xử lý số phức với các chức năng

- Nhập/Xuất số phức.
- Các phép toán trên số phức

Biết rằng tệp giao diện của lớp số phức định nghĩa như sau:

```
#ifndef COMPLEX_H
#define COMPLEX_H
#include <iostream>
using namespace std;

// Định nghĩa lớp Complex
class Complex {
private:
    int Re, Im;
public:
    Complex();
    Complex(int r, int i);

    // Overload operator '+', '-', '*', '/'
    friend Complex operator+ (Complex const &C1, Complex const &C2);
    friend Complex operator- (Complex const &C1, Complex const &C2);
    friend Complex operator* (Complex const &C1, Complex const &C2);
    friend Complex operator/ (Complex const &C1, Complex const &C2);

    friend ostream& operator<< (ostream& out, Complex& C);
    friend istream& operator>> (istream& in, Complex& C);
    // Other functions
};
#endif
```

## Bài 3 – Đơn thức, Đa thức

### 3.1. Cho đơn thức:

$M(x) = ax^n$ ; với a: hệ số (coefficient), x: biến, n: bậc (degree).

Thiết kế lớp **Đơn thức** (*class DonThuc*) với các chức năng:

- Nhập đơn thức
- Xuất đơn thức
- Tính giá trị đơn thức
- Cộng, trừ, nhân, chia đơn thức.

### 3.2. Cho Đa thức:

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

Với :  $a_n x^n$  đơn thức bậc n,

Thiết kế lớp **Đa thức** (*class DaThuc*)– sử dụng mảng động, danh sách liên kết, với các chức năng:

- + Nhập đa thức
- + Xuất đa thức
- + Tính giá trị đa thức

✚ Cộng, trừ, nhân, chia đa thức.

🔴 Tip: list-of-monomial – danh sách các đối tượng Đơn thức

Monomial\*\* list-of-monomial;

### Hướng dẫn Bài 3

```

/*****
* Định nghĩa lớp đơn thức (Monomial)
* file donthuc.h
*  $M(x) = ax^n$ , Với a: hệ số, n: số mũ
*****/
#ifndef DonThuc_H
#define DonThuc_H
#include <iostream>
using namespace std;

/* Lớp Đơn thức */
class DonThuc {
private:
    int coef;
    int exp;
public:
    //(1) Constructors: default constructor, parameter constructor, copy
    constructor.
    DonThuc();
    DonThuc(int, int);
    ~DonThuc();
    //(2) Operators: '+', '-', '*', '/', Gán bằng: '='.
    // Các toán tử so sánh '==', '!=', '>', '<'.
    /* Cong 2 đơn thức cùng bậc => DonThuc */
    DonThuc& CongDT(const DonThuc& src);
    // Định nghĩa phép toán '+' 2 đơn thức cùng bậc
    DonThuc& operator+=(const DonThuc&);
    // So sánh 2 phân số
    /* so sánh 2 DonThuc bằng, lớn hơn, nhỏ hơn */
    bool Equal(const DonThuc& src);
    // overloading operator '=='
    bool operator==(const DonThuc&);

    //(3) Input/Output
    void output();
    // Định nghĩa phép toán xuất '<<'
    friend ostream& operator<<(ostream &out, const DonThuc& src);
    void input();
    // istream operator overloading
    friend istream& operator>>(istream &in, DonThuc& src);
};
#endif
```

## Cài đặt lớp đa thức

Cho đa thức  $P(x)$  như sau:

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

Hay:

$$P(x) = \sum_{k=0}^n a_k x^k, \text{ với } k = 0, 1, \dots, n$$

**Yêu cầu :**

- **Cộng**, trừ, nhân, chia 2 đa thức

**Cộng 2 Đa Thức**

$P1 = 5X^3 + 10X^2 + 15X + 5$   
 $P2 = 2X^2 + 10$   
Tính tổng  $P = P1 + P2$

P1	5X <sup>3</sup>	10X <sup>2</sup>	15X	5
P2		2X <sup>2</sup>		10
P=P1+P2	5X <sup>3</sup>	12X <sup>2</sup>	15X	15

## Cộng hai đa thức

Solution 1: Sử dụng mảng động

Solution 2: Sử dụng danh sách liên kết

-----