

# **Chương 6**

## **TOÁN TỬ GÁN (operator=)**

## 0. MỤC TIÊU

- Hiểu được phương thức toán tử gán là gì?
- Hiểu được vai trò của toán tử gán trong lập trình hướng đối tượng

# 1. KHÁI NIỆM

- Toán tử gán trong lập trình C được sử dụng để gán giá trị của biến này cho biến khác.
- Mở rộng cho C++ và các ngôn ngữ lập trình hướng đối tượng khác ta có thể nói như sau: **Toán tử gán được sử dụng để gán thành phần dữ liệu của đối tượng này cho đối tượng khác.**

## 2. ĐẶT VẤN ĐỀ

- Hãy khai báo và định nghĩa các phương thức cần thiết để các câu lệnh sau có thể thực hiện
- Đoạn chương trình

```
CHocSinh a, b;
```

```
a.Nhap();
```

```
b=a;
```

```
b.Xuat();
```

### 3. GIẢI QUYẾT VẤN ĐỀ

- Đoạn chương trình

```
CHocSinh a,b;
```

```
a.Nhap();
```

```
b=a;
```

```
b.Xuat();
```

- Để giải quyết vấn đề trên ta phải khai báo và định nghĩa **phương thức toán tử gán** cho lớp đối tượng CHocSinh

### 3. GIẢI QUYẾT VẤN ĐỀ

- Để giải quyết vấn đề trên ta phải khai báo và định nghĩa phương thức toán tử gán cho lớp đối tượng CHocSinh
- Khai báo lớp

```
class CHocSinh
{
    private:
        char hoten[31];
        int toan;
        int van;
        float dtb;
    public:
        operator=
};
```

### 3. GIẢI QUYẾT VẤN ĐỀ

- Để giải quyết vấn đề trên ta phải khai báo và định nghĩa phương thức toán tử gán cho lớp đối tượng CHocSinh
- Khai báo lớp

```
class CHocSinh
{
    private:
        char hoten[31];
        int toan;
        int van;
        float dtb;
    public:
        operator=
            (CHocSinh ) ;
};
```

### 3. GIẢI QUYẾT VẤN ĐỀ

- Để giải quyết vấn đề trên ta phải khai báo và định nghĩa phương thức toán tử gán cho lớp đối tượng CHocSinh
- Khai báo lớp

```
class CHocSinh
{
    private:
        char hoten[31];
        int toan;
        int van;
        float dtb;
    public:
        operator=
            (CHocSinh &) ;
};
```



### 3. GIẢI QUYẾT VẤN ĐỀ

- Để giải quyết vấn đề trên ta phải khai báo và định nghĩa phương thức toán tử gán cho lớp đối tượng CHocSinh
- Khai báo lớp

```
class CHocSinh
{
    private:
        char hoten[31];
        int toan;
        int van;
        float dtb;
    public:
        CHocSinh operator=
            (CHocSinh &) ;
};
```

### 3. GIẢI QUYẾT VẤN ĐỀ

- Định nghĩa phương thức toán tử gán

```
CHocSinh CHocSinh::operator=  
                                     (CHocSinh&x)  
{  
    strcpy(hoten, x.hoten);  
    toan = x.toan;  
    van = x.van;  
    dtb = x.dtb;  
    return *this;  
}
```

## 4. CÁC GHI CHÚ QUAN TRỌNG

- Toán tử gán được cài đặt bên trong lớp CHocSinh như là một phương thức của lớp.
- Miền giá trị của một biến con trỏ là địa chỉ ô nhớ.
- Miền giá trị của một con trỏ đối tượng là địa chỉ ô nhớ.
- Bên trong thân của một phương thức, `this` là một con trỏ đối tượng thuộc về lớp mà phương thức đó thuộc về.

## 4. CÁC GHI CHÚ QUAN TRỌNG

- Bên trong thân của một phương thức, **this** là một con trỏ đối tượng thuộc về lớp mà phương thức đó thuộc về.
  - + Ví dụ 01: Bên trong thân phương thức **Nhap** của lớp **CPhanSo**, **this** là con trỏ đối tượng thuộc về lớp CPhanSo.
  - + Ví dụ 02: Bên trong thân phương thức **Xuat** của lớp **CDiem**, **this** là con trỏ đối tượng thuộc về lớp CDiem.
  - + Ví dụ 03: Bên trong thân phương thức **operator =** của lớp **CHocSinh**, **this** là con trỏ đối tượng thuộc về lớp CHocSinh.

## 4. CÁC GHI CHÚ QUAN TRỌNG

- Bên trong thân phương thức của một lớp, **this là một con trỏ đối tượng giữ địa chỉ của đối tượng đang gọi thực hiện phương thức.**
- Bên trong thân phương thức của một lớp, this là một con trỏ đối tượng giữ địa chỉ của đối tượng đang gọi thực hiện phương thức. **Hơn nữa, \*this chính là đối tượng đang gọi thực hiện phương thức.**

## 5. HƯỚNG DẪN SỬ DỤNG

```
CHocSinh CHocSinh::operator=  
    (CHocSinh&x)  
{  
    strcpy(hoten, x.hoten);  
    toan = x.toan;  
    van = x.van;  
    dtb = x.dtb;  
    return *this;  
}
```

– Hướng dẫn sử dụng 01:

1. CHocSinh a,b;
2. a.Nhap();
3. b = a;

– Trong câu lệnh thứ 3 của đoạn chương trình trên ta nói: đối tượng b gọi thực hiện phương thức toán tử gán với tham số là đối tượng a.

## 5. HƯỚNG DẪN SỬ DỤNG

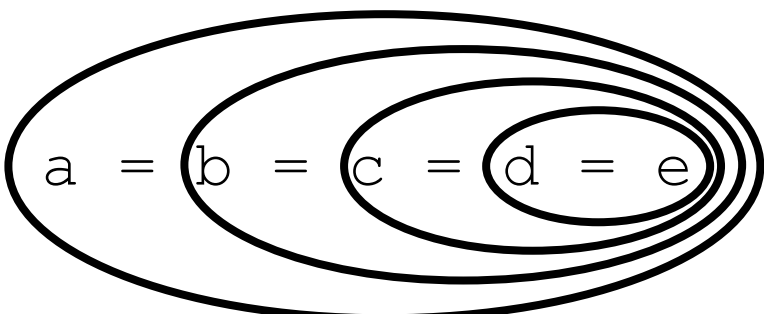
- Định nghĩa phương thức toán tử gán

```
CHocSinh CHocSinh::operator=  
    (CHocSinh&x)  
{  
    strcpy(hoten, x.hoten);  
    toan = x.toan;  
    van = x.van;  
    dtb = x.dtb;  
    return *this;  
}
```

## 5. HƯỚNG DẪN SỬ DỤNG

```
CHocSinh CHocSinh::operator=  
    (CHocSinh&x)  
{  
    strcpy(hoten,x.hoten);  
    toan = x.toan;  
    van = x.van;  
    dtb = x.dtb;  
    return *this;  
}
```

- Hướng dẫn sử dụng 02:
  1. CHocSinh a,b,c,d,e;
  2. e.Nhap();

3.  a = b = c = d = e;

Trong câu lệnh thứ 3 của đoạn chương trình trên ta nói: nói không nổi.



## 5. HƯỚNG DẪN SỬ DỤNG

- Định nghĩa phương thức toán tử gán

```
CHocSinh CHocSinh::operator=  
                                     (CHocSinh&x)  
{  
    strcpy(hoten, x.hoten);  
    toan = x.toan;  
    van = x.van;  
    dtb = x.dtb;  
    return *this;  
}
```

## 6. ỨNG DỤNG

- Hãy khai báo và định nghĩa phương thức toán tử gán cho lớp đối tượng CHonSo
- Khai báo lớp

```
class CHonSo
{
    private:
        int nguyen;
        int tu;
        int mau;
    public:
        CHonSo operator=(CHonSo &);
};
```

## 6. ỨNG DỤNG

- Định nghĩa phương thức toán tử gán

```
CHonSo CHonSo::operator =  
    (CHonSo &x)  
{  
    |   nguyen = x.nguyen;  
    |   tu = x.tu;  
    |   mau = x.mau;  
    |   return *this;  
}
```

## 7. BÀI TẬP VỀ NHÀ

- Hãy khai báo và định nghĩa phương thức toán tử gán cho các lớp đối tượng sau:
  1. Lớp phân số (CPhanSo)
  2. Lớp điểm (CDiem)
  3. Lớp ngày (CNgay)
  4. Lớp thời gian (CThoiGian)
  5. Lớp đơn thức (CDonThuc)
  6. Lớp điểm không gian (CDiemKhongGian)
  7. Lớp đường thẳng (CDuongThang)
  8. Lớp hỗn số (CHonSo)
  9. Lớp số phức (CSoPhuc)
  10. Lớp đường tròn (CDuongTron)
  11. Lớp tam giác (CTamGiac)
  12. Lớp hình cầu (CHinhCau)