

Lập trình hướng đối tượng

Toán tử gán

Nguyễn Khắc Huy



Nội dung



Thành phần tĩnh



Ba vấn đề con trỏ



Toán tử gán



Bài tập



Thành phần tĩnh

- Thành phần của lớp (class members):
 - ✓ Thành phần đối tượng (instance members).
 - Thuộc tính và phương thức thông thường.
 - Mỗi đối tượng có bản sao riêng.
 - ✓ Thành phần tĩnh (static members).
 - Thuộc tính và phương thức tĩnh.
 - Các đối tượng dùng chung.

**Thành phần dùng chung cho
MỌI đối tượng của lớp!!**

PhanSo

số

u số

trị lớn nhất

trị nhỏ nhất

p1: PhanSo

- Tử số
- Mẫu số

p2: PhanSo

- Tử số
- Mẫu số

Thành phần tĩnh

□ Khai báo và sử dụng:

- ✓ Dùng từ khóa static.
- ✓ Truy xuất bằng toán tử ::

```
class PhanSo
{
private:
    static int m_iGiaTriLN;
public:
    static int layGiaTriLN();
private:
    int    m_iTu;
    int    m_iMau;
};
```

```
PhanSo::m_iGiaTriLN = 10000;

void main()
{
    PhanSo  p1(1, 2);
    PhanSo  p2(2, 3);

    int x1 = PhanSo::layGiaTriLN();
    int x2 = p1.layGiaTriLN();
}
```

Nội dung



Thành phần tĩnh



Ba vấn đề con trỏ



Toán tử gán



Bài tập



Ba vấn đề con trỏ

□ Ví dụ 1:

```
class Array
```

```
{
```

```
private:
```

```
    int    m_iSize;
```

```
    int    *m_pData;
```

```
public:
```

```
    Array(int iSize);
```

```
};
```

```
Array::Array(int iSize)
```

```
{
```

```
    m_iSize = iSize;
```

```
    m_pData = new int[m_iSize];
```

```
}
```

```
void main()
```

```
{
```

```
    Array  a1(5);
```

```
    // ...
```

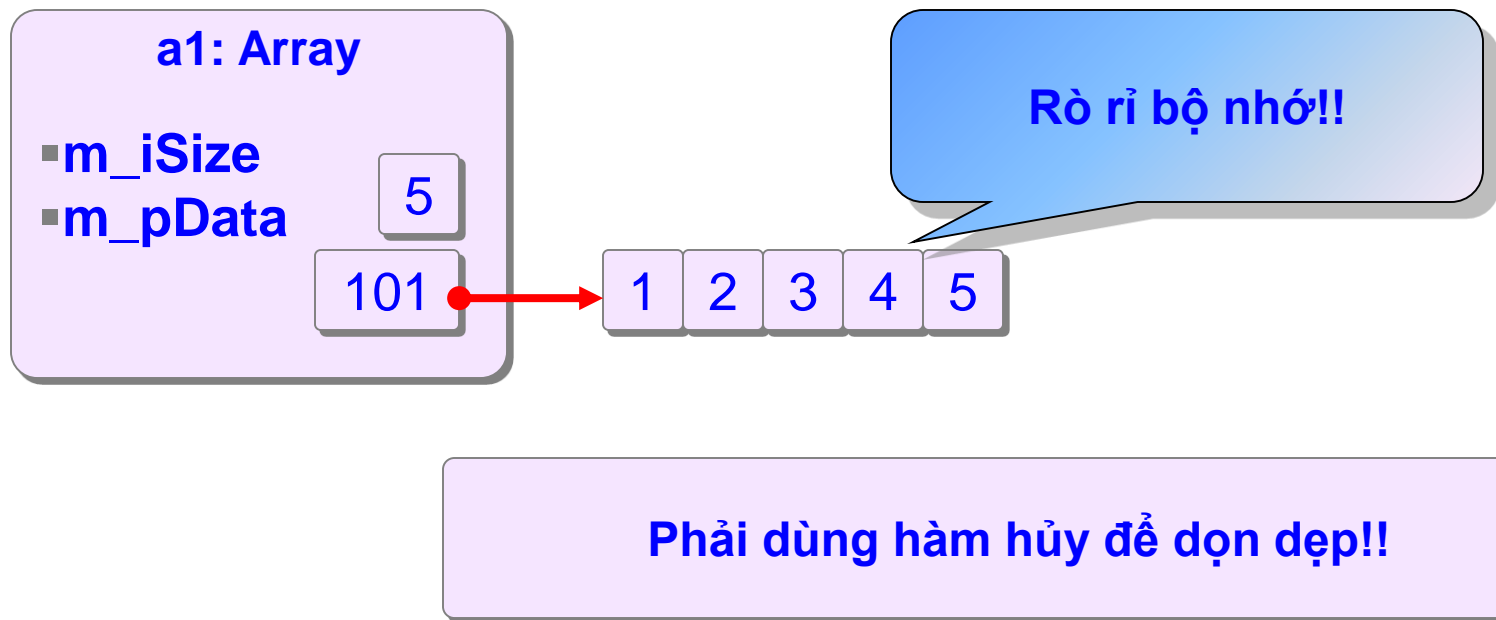
```
}
```



Ba vấn đề con trỏ

□ Vấn đề rò rỉ bộ nhớ:

- ✓ Lớp có thuộc tính con trỏ.
- ✓ Đối tượng của lớp bị hủy, bộ nhớ không thu hồi.



Ba vấn đề con trỏ

□ Ví dụ 1:

```
class Array
```

```
{
```

```
private:
```

```
    int    m_iSize;
```

```
    int    *m_pData;
```

```
public:
```

```
    Array(int iSize);
```

```
    ~Array();
```

```
};
```

```
Array::~~Array()
```

```
{
```

```
    delete m_pData;
```

```
}
```

```
void main()
```

```
{
```

```
    Array  a1(5);
```

```
    ...
```

```
}
```



Ba vấn đề về con trỏ

□ Ví dụ 2:

```
class Array
{
private:
    int    m_iSize;
    int    *m_pData;
public:
    Array(int iSize);
    ~Array();
};
```

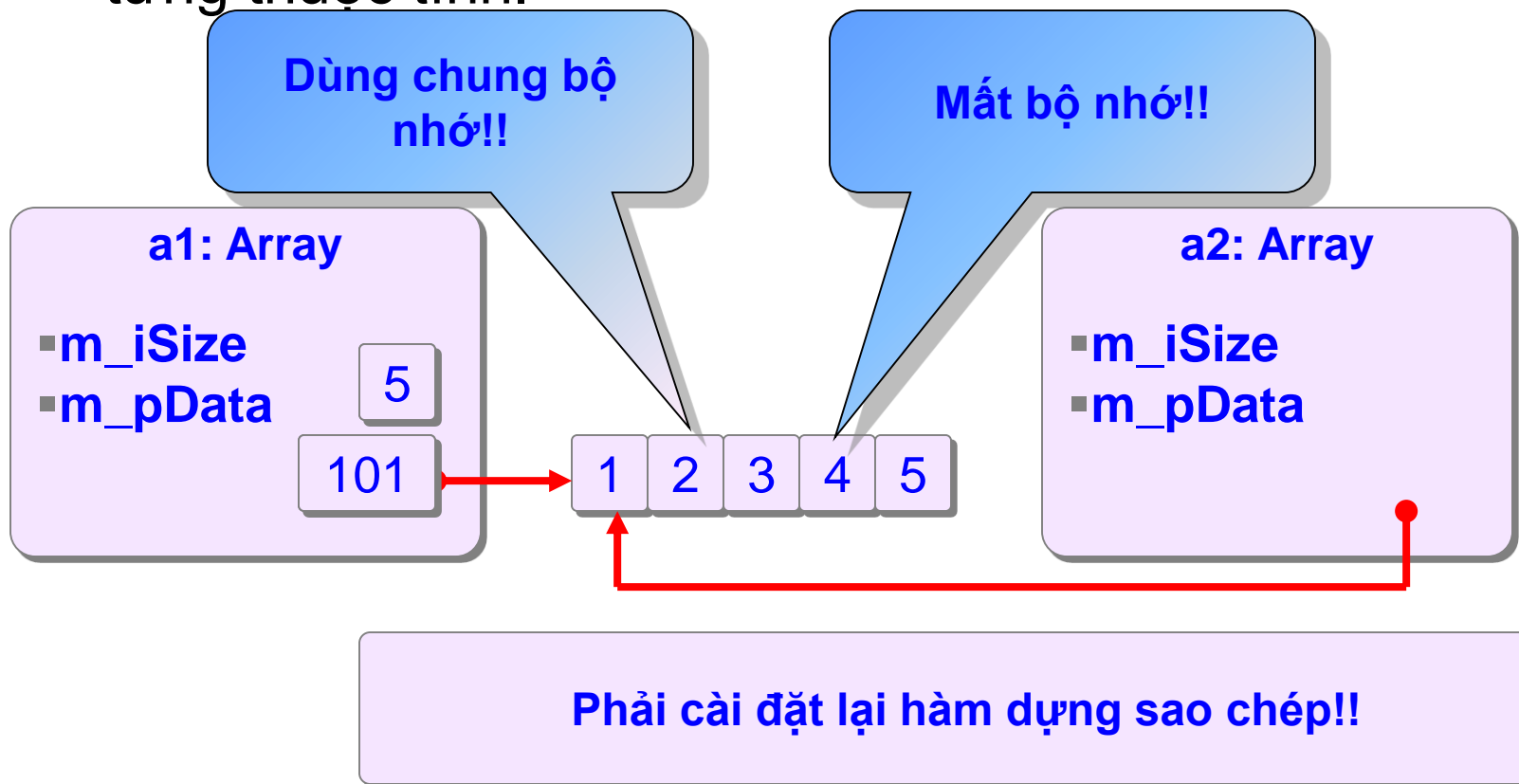
```
void main()
{
    Array  a1(5);
    Array  a2(a1);
    ...
}
```



Ba vấn đề về con trỏ

□ Vấn đề sao chép bộ nhớ (hdsc mặc định):

- ✓ Hàm dựng sao chép mặc định sao chép giá trị từng thuộc tính.



Ba vấn đề con trỏ

□ Ví dụ 2:

```
class Array
```

```
{
```

```
private:
```

```
    int    m_iSize;
```

```
    int    *m_pData;
```

```
public:
```

```
    Array(int iSize);
```

```
    Array(const Array &a);
```

```
    ~Array();
```

```
};
```

```
Array::Array(const Array &a)
```

```
{
```

```
    m_iSize = a.m_iSize;
```

```
    m_pData = new int[m_iSize];
```

```
    for (int i = 0; i < m_iSize; i++)
```

```
        m_pData[ i ] = a.m_pData[ i ];
```

```
}
```

```
void main()
```

```
{
```

```
    Array  a1(5);
```

```
    Array  a2(a1);
```

```
    ...
```

```
}
```



Ba vấn đề con trỏ

□ Ví dụ 3:

```
class Array
```

```
{
```

```
private:
```

```
    int    m_iSize;
```

```
    int    *m_pData;
```

```
public:
```

```
    Array(int iSize);
```

```
    Array(const Array &a);
```

```
    ~Array();
```

```
};
```

```
void main()
```

```
{
```

```
    Array a1(5);
```

```
    Array a2(6);
```

```
    ...
```

```
    a2 = a1;
```

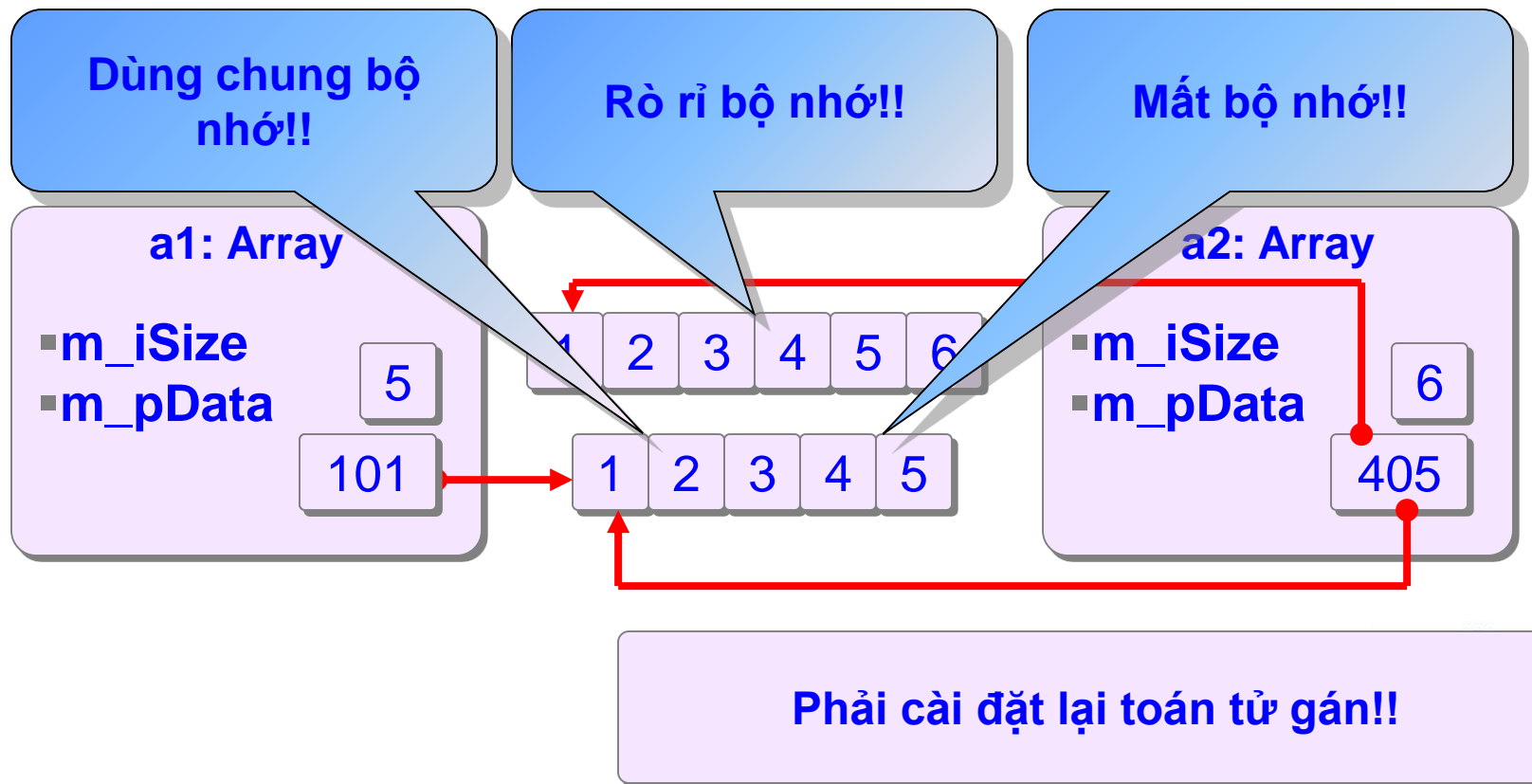
```
    ...
```

```
}
```



Ba vấn đề con trỏ

- ❑ Vấn đề sao chép bộ nhớ (ttg mặc định):
 - ✓ Toán tử gán mặc định sao chép giá trị từng thuộc tính.



Ba vấn đề con trỏ

□ Ví dụ 3:

```
class Array
{
private:
    int      m_iSize;
    int      *m_pData;
public:
    Array(int iSize);
    Array(const Array &a);
    ~Array();
    Array & operator =(const Array &a);
};
```

```
Array & Array::operator =(const Array &a)
{
    delete m_pData;
    m_iSize = a.m_iSize;
    m_pData = new int[m_iSize];
    for (int i = 0; i < m_iSize; i++)
        m_pData[ i ] = a.m_pData[ i ];
    return *this;
}
```

```
void main()
{
    Array  a1(5);
    Array  a2(6);
    ...
    a2 = a1;
    ...
}
```



Ba vấn đề con trỏ

□ Dr. Guru khuyên: **luật “ba ông lớn”**

✓ Lớp có thuộc tính con trỏ, phải luôn kèm theo:

- Hàm hủy: thu hồi bộ nhớ.
- Hàm dựng sao chép: sao chép bộ nhớ.
- Toán tử gán: sao chép bộ nhớ.

```
class HocSinh
{
private:
    char    *m_sHoTen;
public:
    HocSinh(const HocSinh &hs);
    ~HocSinh();
    HocSinh & operator =(const HocSinh &hs);
};
```



Nội dung



Thành phần tĩnh



Ba vấn đề con trỏ



Toán tử gán



Bài tập



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.



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

```
void main()
{
    CHocSinh a,b;
    a.Nhap();
    b=a;
    b.Xuat();
}
```

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



Giải quyết vấn đề

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

public:
    CHocSinh operator = (CHocSinh &);
}
```



Định nghĩa toán tử

```
CHocSinh CHocSinh::operator= (CHocSinh &x)
```

```
{
```

```
    if (hoten != NULL)
```

```
        delete[] hoten;
```

```
    hoten = new char[strlen(x.hoten)];
```

```
    strcpy(hoten,x.hoten);
```

```
    toan = x.toan;
```

```
    van = x.van;
```

```
    dtb = x.dtb;
```

```
    return *this;
```

```
}
```

20



Một số lưu ý

- ❑ 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.



Khai báo sử dụng

□ Khai báo

```
CHocSinh CHocSinh::operator= (CHocSinh &x)
```

□ Ví dụ 1:

```
CHocSinh a,b;
```

```
a.nhap();
```

```
b = a;
```

//Đối tượng b gọi thực hiện phương thức toán tử
gán với tham số là đối tượng a.



Khai báo sử dụng

□ Khai báo

```
CHocSinh CHocSinh::operator= (CHocSinh &x)
```

□ Ví dụ 1:

```
CHocSinh a,b,c,d,e;
```

```
a.nhap();
```

```
a = b = c = d = e;
```

//Đối tượng d gọi thực hiện phương thức toán tử
gán với tham số là đối tượng..



Nội dung



Thành phần tĩnh



Ba vấn đề con trỏ



Toán tử gán



Bài tập



Bài tập

- Hãy khai báo và định nghĩa phương thức toán tử gán cho các lớp đối tượng sau:
 - Lớp ngày (CNgay)
 - Lớp thời gian (CThoiGian)
 - Lớp đơn thức (CDonThuc)
 - Lớp điểm không gian (CDiemKhongGian)
 - Lớp đường thẳng (CDuongThang)
 - Lớp hỗn số (CHonSo)
 - Lớp hình cầu (CHinhCau)



BÀI TẬP VỀ NHÀ



Bài tập

□ Bài tập 3.1:

Bổ sung vào lớp **phân số** những phương thức sau:

(Nhóm tạo hủy)

- ✓ Khởi tạo mặc định phân số = 0.
- ✓ Khởi tạo với tử và mẫu cho trước.
- ✓ Khởi tạo với giá trị phân số cho trước.
- ✓ Khởi tạo từ một phân số khác.

(Nhóm toán tử)

- ✓ Toán tử số học: =
- ✓ Các phương thức cần thiết giải quyết 3 vấn đề con trỏ



Bài tập

□ Bài tập 3.2:

Bổ sung vào lớp **số phức** những phương thức sau:

(Nhóm tạo hủy)

- ✓ Khởi tạo mặc định số phức = 0.
- ✓ Khởi tạo với phần thực và phần ảo cho trước.
- ✓ Khởi tạo với giá trị thực cho trước.
- ✓ Khởi tạo từ một số phức khác.

(Nhóm toán tử)

- ✓ Toán tử số học: =
- ✓ Các phương thức cần thiết giải quyết 3 vấn đề con trỏ



Bài tập

□ Bài tập 3.3:

Bổ sung vào lớp **đơn thức** những phương thức sau:

(Nhóm tạo hủy)

- ✓ Khởi tạo mặc định đơn thức = 0.
- ✓ Khởi tạo với hệ số và số mũ cho trước.
- ✓ Khởi tạo với hệ số cho trước, số mũ = 0.
- ✓ Khởi tạo từ một đơn thức khác.

(Nhóm toán tử)

- ✓ Toán tử số học: =
- ✓ Các phương thức cần thiết giải quyết 3 vấn đề con trỏ



Bài tập

□ Bài tập 3.4:

Bổ sung vào lớp **học sinh** những phương thức sau:

(Nhóm tạo hủy)

- ✓ Khởi tạo với họ tên và điểm văn, toán cho trước.
- ✓ Khởi tạo với họ tên cho trước, điểm văn, toán = 0.
- ✓ Khởi tạo từ một học sinh khác.

(Nhóm toán tử)

- ✓ Toán tử số học: =
- ✓ Các phương thức cần thiết giải quyết 3 vấn đề con trỏ



Bài tập

□ Bài tập 3.5:

Bổ sung vào lớp **mảng** những phương thức sau:

(Nhóm tạo hủy)

- ✓ Khởi tạo mặc định mảng kích thước = 0.
- ✓ Khởi tạo với kích thước cho trước, các phần tử = 0.
- ✓ Khởi tạo từ một mảng `int []` với kích thước cho trước.
- ✓ Khởi tạo từ một đối tượng `IntArray` khác.
- ✓ Hủy đối tượng `IntArray`, thu hồi bộ nhớ.

(Nhóm toán tử)

- ✓ Toán tử số học: =
- ✓ Các phương thức cần thiết giải quyết 3 vấn đề con trỏ



Tham khảo

- Bài giảng phương pháp lập trình HDT:
 - Thầy Đinh Bá Tiến : dbtien@fit.hcmus.edu.vn
 - Thầy Nguyễn Minh Huy: nmhuy@fit.hcmus.edu.vn
 - Thầy Lê Xuân Định lxdingh@fit.hcmus.edu.vn

