

Static Members and Operators

Nguyễn Văn Khiết

Nội dung

- Static Member
- Operator độc lập
- friend

Static Member

- Lớp đối tượng có thể có thuộc tính hoặc phương thức khai báo dưới dạng static
- Lúc này các thuộc tính hoặc phương thức static là thuộc tính chung của lớp chứ không phải thuộc tính, phương thức của đối tượng
- ➔ Thuộc tính hoặc phương thức toàn cục ở cấp độ lớp.
- Truy xuất không cần thông qua đối tượng.

Static Member

- Khai báo hàm static : từ khóa static

```
class Math {  
    public:  
        static double Max(double, double);  
};
```

```
double Math::Max(double x, double y){  
    if (x>y)  
        return x;  
    return y;  
}
```

...

```
double d = Math::Max(2.34, 4.55);
```

Static Member

- Khai báo thuộc tính static:
 - Từ khóa static
 - Ta phải khởi tạo giá trị ban đầu của thuộc tính static bên ngoài class
 - Sau khi khởi tạo, có thể sử dụng thuộc tính:
 - Là dữ liệu dùng chung cho tất cả các thể hiện của lớp, không phụ thuộc số đối tượng được tạo ra của lớp đó

Static Member

- Khai báo thuộc tính static:

```
class GPS {  
    ...  
    public:  
        static int ObjectCount;  
    ...  
};
```

```
int GPS::ObjectCount = 2;
```

```
void main()  
{  
    ...  
    cout<<GPS::ObjectCount;  
}
```

Static Member

- Ví dụ: Viết lớp thực hiện các thao tác toán học trên số: Max, Min, Abs, Round, Ceiling, Floor, Sqrt, Sin, Cos, ...

Operator độc lập

- Toán tử ở cấp toàn cục và không thuộc lớp nào
- Định nghĩa mở rộng cho các toán tử có sẵn khi có thêm kiểu dữ liệu (lớp đối tượng) mới cho các toán hạng.
 - VD: cộng số nguyên với phân số

PhanSo operator+(int a, PhanSo& ps)

```
{  
    PhanSo kq;  
    ...  
    return kq;  
}
```


Operator độc lập

- Đôi khi, có thể định nghĩa cho 2 toán hạng thuộc cùng kiểu dữ liệu (lớp đối tượng)
- Một số ví dụ
 - Mở rộng toán tử $<<$, $>>$
 - Phép cộng 2 phân số.

Khai báo friend

- Hàm friend của một lớp là hàm nằm ngoài lớp nhưng được truy cập đến tất cả thành viên (public, private, protected) của lớp đó.
- Khai báo hàm friend bên trong header của lớp với từ khóa friend.

Khai báo hàm friend

- VD:

```
class PhanSo {  
    ...  
    friend ostream& operator<<(ostream &out, const PhanSo& ps)  
}  
  
ostream& operator<<(ostream &out, const PhanSo& ps)  
{  
    out<<ps.mTuSo<<"/"<<ps.mMauSo<<endl;  
    return out;  
}  
  
...  
PhanSo ps1(1,3);  
cout<<ps1;
```

Khai báo friend

- Lớp friend của một lớp là lớp có thể truy cập đến tất cả thành viên (public, private, protected) của lớp đó.
- Khai báo lớp friend bên trong header của một lớp với từ khóa friend.

```
class GPS
{
    private:
        double mLongitude;
        double mLatitude;
        double mSpeed;
        double mDirection;
    public:
        void Print();
        friend class MobileCab;
};
```