

Nạp chồng hàm

Trong 1 class, ta có thể ghi các hàm cùng tên với nhau. Tuy nhiên, các tham số trong các hàm này phải khác nhau.

Nạp chồng hàm

```
class PrintData {  
    public:  
        void print(int i) {  
            cout << "Printing int: " << i << endl;  
        }  
        void print(double f) {  
            cout << "Printing float: " << f << endl;  
        }  
        void print(char* c) {  
            cout << "Printing character: " << c << endl;  
        }  
};
```

Nạp chồng hàm

```
PrintData p1, p2;
```

```
p.print(1);
```

```
p.print(1.0);
```

```
p.print('c');
```

Nạp chồng hàm

```
class PrintData {  
    public:  
        int print(int i) {  
            cout << "Printing int: " << i << endl;  
  
            return 1;  
        }  
        void print(int f) {  
            cout << "Printing float: " << f << endl;  
        }  
};  
  
//Lỗi
```

Nạp chồng hàm

```
class PrintData {  
    public:  
        void print(int i, int j) {  
            cout << i << ", " << j << endl;  
        }  
        void print(int f) {  
            cout << "Printing float: " << f << endl;  
        }  
};
```

Nạp chồng hàm

```
PrintData p1, p2;
```

```
p.print(1);
```

```
p.print(1, 2);
```

Nạp chồng toán tử

Trong C++, cho phép chúng ta có thể thay đổi định nghĩa các toán tử như `+` `-` `*` `/` `<<` `>>` `++` `--` ... để dùng với các đối tượng.

```
SoPhuc a, b, c;
```

```
c = a + b;
```

```
C = a - b;
```

Nạp chồng toán tử

```
class PhanSo {  
    public:  
        float tuso, mauso;  
        PhanSo operator+(int a);  
};  
PhanSo p;  
PhanSo p1 = p + 5;
```


Nạp chồng toán tử

```
class PhanSo {  
    public:  
        float tuso, mauso;  
        PhanSo operator+(PhanSo a);  
        PhanSo operator+(int a);  
};  
  
PhanSo p1, p2;  
PhanSo p3 = p1 + 5;  
PhanSo p4 = p1 + p2;
```

Nạp chồng toán tử

```
class PhanSo {  
    public:  
        float tuso, mauso;  
        int operator%(int a);  
};  
PhanSo p1;  
int p = p1 % 5;
```