

C3_Con trỏ đối tượng, con trỏ this

📅 Ngày học	@September 20, 2022
📎 Property	Bai giang_LTHDT - Chuong 3.pptx (phan 1).pdf
📎 Property 1	

1. Con trỏ đối tượng

Con trỏ đối tượng dùng để chứa địa chỉ của biến đối tượng, được khai báo như sau :

Tên lớp * Tên con trỏ;

Chẳng hạn, với lớp Diem, ta có thể khai báo:

Diem *p1,*p2,*p3; // Khai báo 3 con trỏ p1,p2,p3

2. Con trỏ this

C++ sử dụng một con trỏ đặc biệt có tên là this, nó là **tham số đầu tiên của các phương thức**.

Khi một đối tượng gọi đến phương thức thì tham số truyền cho con trỏ this chính là **địa chỉ của đối**

tượng đó, vì vậy **con trỏ this sẽ trỏ tới đối tượng đó**.

Các thuộc tính viết trong phương thức được hiểu là thuộc một đối tượng do con trỏ this trỏ tới. Như vậy, với lời gọi trên thì các thuộc tính x, y trong hàm nhập chính là: this->x, this->y

3. Phép gán trên đối tượng

Phép gán = có thể được sử dụng để **gán một đối tượng cho một đối tượng khác** của cùng một kiểu.

Toán tử gán như thế bình thường được thực hiện bởi toán tử sao chép thành phần – mỗi thành phần của một đối tượng được sao chép riêng tới cùng thành phần ở đối

tượng khác

Việc sao chép thành phần có thể phát sinh lỗi khi sử dụng với một lớp mà thành phần dữ liệu chứa **vùng nhớ cấp phát động**. Trong trường hợp này, ta cần phải quá tải toán tử gán = (chương 4)

```
//Ham Cong, rutgon deu thuoc class PhanSo
PhanSo PhanSo::Cong(PhanSo p){
    PhanSo s ;
    s.tu = (tu * p.mau) + (mau * p.tu);
    s.mau = mau * p.mau;
    return s;
}

int main(){
    PhanSo p1, p2, tong;
    cout << "\tNhap phan so thu nhat: \n";
    p1.Nhap();
    cout << "\tNhap phan so thu hai: \n";
    p2.Nhap();
    tong = p1.Cong(p2);
    tong.rutgon();
    cout << "\t\tKET QUA: \n";
    p1.Xuat(); cout << "\t+" ; p2.Xuat(); cout << "\t=";
}
```