

# C4\_Toán tử tải bội >> ; <<

📅 Ngày học	@October 25, 2022
📎 Property	<a href="#">Bai giang_LTHDT - Chuong 4 (phan 2).pdf</a>
📎 Property 1	

Biểu thức được trả lại **trong câu lệnh return** phải là tên của một **biến xác định từ bên ngoài hàm**, bởi vì khi đó mới có thể sử dụng được giá trị của hàm.

Khi **giá trị trả về của hàm là tham chiếu**, ta có thể gặp câu lệnh gán hơi khác thường, trong đó **vế trái là một lời gọi hàm chứ không phải là tên của một biến**. Nói cách khác, **vế trái của lệnh gán** có thể là **lời gọi đến một hàm có giá trị trả về là một tham chiếu**.

## 1. Phân biệt tiền tố, hậu tố:

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    cout << "Toan tu ++a: (Tang roi moi dung):\n";
    int a = 1;
    int b = ++a;
    cout << "a = " << a; //a = 2
    cout << "\nb = " << b; // b = 2

    //++ sau: dung r moi tang
    cout << "\n\nToan tu a++: (Dung r moi tang):\n";
    int c = 3;
    int d = c++;
    cout << "c = " << c; // c = 4
    cout << "\nd = " << d; // d = 3
}
```

## 2. Xây dựng phương thức của tiền tố, hậu tố ++

```

class Diem:

//Tien to: tang roi moi dung
Diem Diem::operator++(){
    ++x;
    ++y;
    return *this;
}

//Hau to: dung roi moi tang
Diem Diem::operator ++(int){
    Diem tam;
    tam = *this;
    ++*this;
    return tam;
}

```

### 3. Quá tải toán tử []

#### !!! Xem lại khái niệm Hàm trả về giá trị tham số

Biểu thức được trả lại **trong câu lệnh return** phải là tên của một **biến xác định từ bên ngoài hàm**, bởi vì khi đó mới có thể sử dụng được giá trị của hàm.

Khi **giá trị trả về của hàm là tham chiếu**, ta có thể gặp câu lệnh gán hơi khác thường, trong đó **vế trái là một lời gọi hàm chứ không phải là tên của một biến**. Nói cách khác, **vế trái của lệnh gán** có thể là **lời gọi đến một hàm có giá trị trả về là một tham chiếu**.

```

Kiểu &Tên hàm(...){
//thân hàm
return <biến phạm vi toàn cục>;
}

```