Ngôn ngữ lập trình C++

Hàm, con trỏ, mảng, xâu ký tự, trong C++

Nguyễn Văn Tiến

Nội dung

1. Hàm
2. Con trỏ
3. Mảng
4. Xâu ký tự

Hàm trong C++

Khái niệm

- Một hàm là một nhóm các câu lệnh cùng nhau thực hiện một nhiệm vụ. Mỗi chương trình C++ có ít nhất một hàm, là hàm main()

Lợi thế của các hàm trong C++

- Có khả năng sử dụng lại
- Tối ưu hóa chương trình

Hàm trong C++

Khai báo và sử dụng hàm

```
<Kiểu trả về> <Tên hàm>(Danh sách tham số) {
                    <Câu lệnh 1>;
                    <Câu lệnh 2>;
}
```

Ví dụ

```
int kiem_tra_nt(long long n) {
        if (n < 2) return 0;
        for (int i = 0; i <= sqrt(n); i++) {
                if (n % i == 0) return 0;
        }
        return 1;
}
```

Hàm trong C++

Chồng hàm

- Chồng hàm là kỹ thuật sử dụng 2 hàm có cùng tên nhưng khác tham số hoặc khác kiểu trả về

Ví dụ

```
int tong(int a, int b) {
        return a + b;
}
float tong(float a, float b) {
        return a + b;
}

tong(5, 6);
tong(5.5f, 4.5f);
```

Tham trị, tham chiếu

Tham trị

- Giá trị được truyền vào tham số của hàm, sau khi kết thúc hàm, giá trị của biến được truyền vào không thay đổi.

Tham chiếu

- Nội dung được truyền vào tham số của hàm, sau khi kết thúc hàm, giá trị của biến được truyền vào bị thay đổi.
- Để truyền tham chiếu vào hàm, sử dụng dấu **&** trước tên biến.
- **Lưu ý:** Mảng truyền vào hàm sẽ thay đổi giá trị sau khi gọi hàm nếu trong hàm thực hiện thay đổi giá trị các phần tử.

Nguyên tắc xây dựng hàm

Chức năng

- Cố gắng mỗi hàm thực hiện giải quyết một vấn đề duy nhất

Xác định kiểu trả về

- Nếu hàm trả giá trị thì xác định kiểu dữ liệu tương ứng, return phù hợp
- Nếu hàm chỉ thực hiện thủ tục, không mang giá trị thì kiểu trả về là void
- Mỗi hàm chỉ trả về được 1 giá trị thông qua kiểu trả về, muốn trả về nhiều giá trị thì có thể sử dụng tham chiếu

Đặt tên

- Đặt tên hàm có ý nghĩa

Con trỏ

Khái niệm

- Con trỏ là một biến đặc biệt, nó chứa địa chỉ của một biến khác. Con trỏ có kiểu là kiểu của biến mà nó trỏ tới

<Kiểu dữ liệu> *<tên con trỏ>;

Lấy địa chỉ con trỏ

<Tên con trỏ> = &<tên biến>

Lấy giá trị con trỏ

*<Tên con trỏ>

Phép gán con trỏ

<Tên con trỏ1> = <Tên con trỏ2>

Con trỏ

Đặc trưng

- Hiệu quả nhưng khó điều khiển
- Quan hệ chặt chẽ với mảng và xâu
- Có thể khai báo kiểu con trỏ cho bất kỳ kiểu dữ liệu nào
- Con trỏ khởi tạo bằng 0, NULL (Không trỏ vào gì cả), hoặc một địa chỉ.

Con trỏ

Lỗi thường gặp

```
int value, *p;

p = value; // sai, p cần địa chỉ, không phải giá trị

*p = &value; // sai, *p là giá trị nên phải được gán giá trị, không phải địa chỉ

p = &value; //đúng

*p = value; // đúng
```

Mảng

Khái niệm

- Mảng là một loại cấu trúc dữ liệu trong ngôn ngữ lập trình C/C++, nó lưu trữ một tập hợp tuần tự các phần tử cùng kiểu với độ dài cố định.
- Mảng thường được sử dụng để lưu trữ tập hợp dữ liệu, nhưng nó cũng hữu dụng khi dùng để lưu trữ một tập hợp biến có cùng kiểu.

Sử dụng mảng

<Kiểu dữ liệu> <tên biến mảng>[<Số lượng phần tử tối đa>];

```
int array[4] = { 5, 8, 2, 7 };
```

Mảng hai chiều trong C++

Sử dụng mảng 2 chiều

```
int arr[3][5];

int arr[3][5] = {{5, 12, 17, 9, 3}, {13, 4, 8, 14, 1}, {9, 6, 3, 7, 21}};
```

Con trỏ và mảng

Mảng và con trỏ có quan hệ mật thiết

- Tên mảng coi như con trỏ hằng
- Con trỏ có thể thao tác trên chỉ số của mảng

Truy cập các phần tử mảng bằng con trỏ

- Phần tử b[n] có thể được truy cập bằng *(bPtr+n)
- Địa chỉ, ví dụ: &b[3]tương đương bPtr + 3
- Tên mảng có thể coi như con trỏ, ví dụ:

  b[3] tươngđương *(b+3)

- Có thể đánh chỉ số cho con trỏ, ví dụ:

  bPtr[3] tương đương b[3]

Con trỏ và mảng

Khai báo mảng bằng con trỏ

```
int *a;

a = new int[SIZE];

int **a;

a = new int*[SIZE];
```

Thu hồi bộ nhớ cấp phát

```
delete[] a;
```

Xâu ký tự trong C++

Mảng các ký tự

- Một xâu có thể được coi là một mảng một chiều, trong đó mỗi phần tử là một ký tự. Ký tự kết thúc là '\0'

```
char s1[200];
```

- fgets(s1,200,stdin): nhập xâu
- strcpy(s1, s2): copy xâu s2 vào s1
- strcat(s1, s2): nối xâu s2 vào sau xâu s1
- strlen(s) : tính độ dài xâu s
- strcmp(s1, s2) : so sánh hai xâu s1 và s2 theo thứ tự từ điển.

Xâu ký tự trong C++

Kiểu dữ liệu string

- Trong C++ có kiểu dữ liệu dành riêng cho xử lý chuỗi là string

```
string str;
```

s1 = s2 sao chép xâu s2 vào s1

s1 = s1 + s2 Nối s2 vào sau s1

s1 == s2 So sánh xâu s1 có trùng xâu s2 hay không

s1 > s2 So sánh xâu s1 có đứng trước s2 hay không theo thứ tự từ điển

s1 < s2 So sánh xâu s1 có đứng sau s2 hay không theo thứ tự từ điển

s1 <= s2 So xánh xâu s1 có đứng trước hoặc trùng s2 hay không theo thứ tự từ điển

s1 >= s2 So xánh xâu s1 có đứng sau hoặc trùng s2 hay không theo thứ tự từ điển

s1 != s2 So sánh xâu khác nhau hay không

Xâu ký tự trong C++

Một số phương thức trong xử lý xâu

STT	Tên hàm (Phương thức)	Ý nghĩa
1	s.size()	Trả lại độ dài string s
2	s.length()	Trả lại độ dài string s
3	getline(cin, s)	Nhập một dòng từ bàn phím cho string s
4	s.erase(n, k)	Xóa k ký tự trong s kể từ vị trí thứ n
5	s.insert(n, s1)	Chèn s1 vào s kể từ vị trí thứ n.
6	s.insert(n, s1, k, m)	Chèn m ký tự kể từ ký tự thứ k trong s1 vào s kể từ vị trí thứ n.
7	s.replace(n, k, s1)	Thay thế k ký tự trong s kể từ vị trí thứ n bằng xâu s1.
8	s.find(s1)	Trả lại vị trí xuất hiện đầu tiên của s1 trong s.
9	s.rfind(s1)	Trả lại vị trí xuất hiện cuối cùng của s1 trong s.
10	s.at(int i)	Truy nhập đến phần tử thứ i trong string