

Mảng trong C

Mảng là gì, mảng nhúng là gì, cách phân bố trong bộ nhớ của mảng, làm thế nào khai báo và làm việc với mảng, khi nào sử dụng mảng

Mảng là 1 cấu trúc dữ liệu lưu trữ 1 tập hợp các phần tử có cùng kiểu có định nghĩa, có cùng kiểu dữ liệu

Ví dụ: `int arr[5];`

Kích thước của mảng arr là $5 \times \text{kích thước integer} = 5 \times 4 \text{ bytes} = 20 \text{ bytes}$

Kích thước của mảng phải là số nguyên dương

Cách nhúng mảng

Method #1: <code>int arr[5] = {1, 2, 6, 9, 14};</code>	Method #2: <code>int arr[] = {1, 2, 6, 9, 14};</code>
Method #3: <code>int arr[5]; arr[0] = 1; arr[1] = 2; arr[2] = 6; arr[3] = 9; arr[4] = 14;</code>	Method #4: <code>int arr1[5] = {5, 6}; int arr2[5] = {0};</code> Method #5: <code>int arr[5] = {[4] = 12, [1] = 7};</code>

Mảng nhúng là mảng của các mảng, cú pháp tổng quát như mảng 1 chiều

Multidimensional arrays can be defined as an array of arrays

Syntax:

```
data_type name_of_array [size1][size2]...[sizeN];
```

For example:

- 2-D array: `char arr[2][3];`
- 3-D array: `int arr1[4][5][6];`

Số lượng chiều của mảng sẽ ảnh hưởng đến kích thước bộ nhớ và kiểu dữ liệu integer, ví dụ mảng 2 chiều

`Char arr[2][3];` đây là mảng gồm 2 hàng và 3 cột

Mảng 3 chiều `int arr1[4][5][6];` có thể coi là 4 mảng 2 chiều 5 hàng và 6 cột

Cách kh i t o t ng t m ng l chi u

Ưu điểm của mảng

-) Có th d dàng truy c p các ph n t trong b nh c a m ng
-) Vì c tìm ki m ph n t trong m ng d dàng
-) M ng nhi u chi u h tr các bài toán x lý bài toán nh ma tr n
-) Vì c phân b b nh liên ti p trong m ng giúp chúng ta có th s d ng con tr truy c p t t c các ph n t trong m ng

Nhược điểm

-) Do kích th c c a m ng là c nh cho nên trong quá trình s d ng ta ko th t ng hay gi m kích th c c a m ng
-) Ko th l u tr c các ph n t khác ki u d li u
-) Vì c l u tr các ph n t khác ki u d li u có th cân nh c s d ng struct tu vào bài toán
-) Vì c thêm ho c xoá ph n t trong m ng không d dàng

Làm thế nào để làm việc với mảng

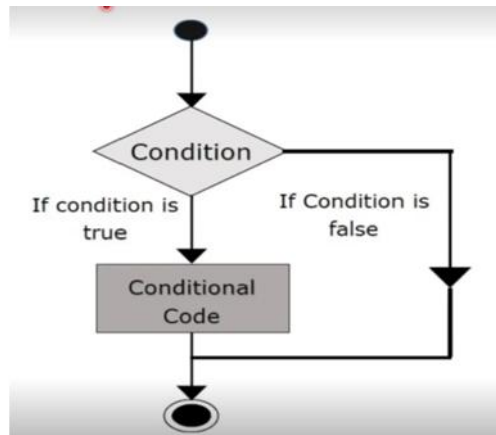
Các bài toán làm vì c v i m ng là truy c p các ph n t c a m ng, tìm ki m, thêm b t ph n t trong m ng, s p x p ph n t trong m ng, s d ng m ng character và hay là chu i ký t .

Khi nào s d ng m ng ng v i các bài toán yêu c u nh l u tr l s li u nh t nh các d li u có cùng ki u và ko yêu c u ph i thêm b t d li u trong quá trình s d ng.

Tu thu c vào bài toán có th l a ch n s d ng các c u trúc d li u khác nh liên k t chu i, stack, heap, Q, tree, (ây là các c u trúc d li u)

Cấu trúc điều kiện rẽ nhánh trong C

-) Giới thiệu về cấu trúc điều kiện rẽ nhánh
-) Làm thế nào xây dựng biểu thức điều kiện
-) Các câu lệnh if, else, các câu lệnh switch case



Các biểu thức hiển thị bài toán: chia phần nguyên vào tổng, chia số với 2 và lấy phần dư. Nếu phần dư bằng 0 thì số đó là số chẵn, khác 0 là số lẻ.

Vòng lặp trong C

-) Cách vào và thoát vòng lặp
-) Làm cách nào sử dụng vòng lặp

Vòng lặp trong C là 1 phần mã code cho phép thực hiện nhiều lần cho đến khi điều kiện kết thúc thỏa mãn.

Các loại vòng lặp trong C là for, while, do ... while

Vòng lặp while thực hiện nhiều lần khi điều kiện ở vế còn đúng thỏa mãn.

Syntax: The do...while loop

```
do{  
    statement;  
} while (condition);
```

Khác vòng lặp while chỉ vòng lặp do while sẽ thực hiện ít nhất 1 lần

Từ khoá trong vòng lặp

-) **Goto**
-) **Continue**
-) **Exit()**
-) **Break**
-) **Return**

Goto label

Đặc điểm của chuyển ngôn ngữ lập trình có nhãn, không nên sử dụng goto trong chương trình bởi vì goto phá vỡ chương trình có cấu trúc và công việc khi đang maintain chương trình

Continue

Giúp chúng ta bỏ qua các câu lệnh phía sau và thể hiện vòng lặp mới

Tuần tự vào chương trình chúng ta có thể sử dụng continue khi mà lần tiếp theo khi xảy ra mà chúng ta không muốn thể hiện những cái khi lần còn lại trong vòng lặp

Lệnh exit() giúp chúng ta thoát khỏi toàn bộ chương trình

Lệnh break giúp chúng ta thoát khỏi vòng lặp chương trình và tiếp tục

Lệnh return giúp chúng ta thoát khỏi chương trình con và quay trở lại chương trình đã gọi chương trình con. Ví dụ hàm main gọi n hàm con/ function, lệnh return sẽ giúp chúng ta thoát khỏi function và trở lại hàm main

Câu hỏi: có 1 con trỏ trỏ vào phần tử của mảng thì vì nó con trỏ y và ký tự a y có thể thao tác 1 vài thứ khá là giống nhau nhưng mà có gì phải làm dc mà a ko làm dc và i khác nhau gì 2 thứ đó? Chứ, hì chung chung quá