

Nhập môn Lập trình – IT001

Con trỏ và Mảng

Nội dung

- Sự tương quan mảng 1 chiều và con trỏ
- Sử dụng tham số mảng 1 chiều trong hàm
- Sự tương quan mảng 2 chiều và con trỏ
- Sử dụng tham số mảng 2 chiều trong hàm

SỰ TƯƠNG QUAN MẢNG 1 CHIỀU VÀ CON TRỎ

- Tên của mảng là một hằng địa chỉ, chứa địa chỉ của phần tử đầu tiên.
- Với khai báo: `int a[10]` thì:
 - `a` tương đương với `&a[0]`
 - `a+i` tương đương với `&a[i]`
 - `*(a+i)` tương đương với `a[i]`
- Xét khai báo : `int a[5], *pa=a;` khi đó con trỏ `pa` giữ địa chỉ của phần tử đầu tiên của mảng `a`. Lúc này:
 - `pa+i` tương đương với `&a[i]`.
 - `*(pa+i)` tương đương với `a[i]`.

Nhắc lại: Nhập xuất mảng 1 chiều

```
#include <stdio.h>
#include <conio.h>
#define MAX 100
void main()
{
    int a[MAX], n, i;
    printf("So phan tu cua mang:");
    scanf("%d",&n);

    for(i=0;i<n;i++)
    {
        printf("Nhap phan tu a[%d]: ",i);
        scanf("%d", &a[i]);
    }
    for(i=0;i<n;i++)
        printf("%4d", a[i]);
}
```

Ví dụ: Nhập xuất mảng 1 chiều bằng con trỏ

```
#include <stdio.h>
#include <conio.h>
#define MAX 100
void main()
{
    int a[MAX],n,i;
    printf("So phan tu cua mang:");
    scanf("%d",&n);

    for(i=0;i<n;i++)
    {
        printf("Nhap phan tu a[%d]: ",i);
        scanf("%d", a+i);
    }
    for(i=0;i<n;i++)
        printf("%4d", *(a+i));
}
```

Ví dụ: Nhập xuất mảng 1 chiều bằng con trỏ

```
#include <stdio.h>
#include <conio.h>
#define MAX 100
void main()
{
    int a[MAX],n,i,*pa;
    pa=a;
    printf("So phan tu cua mang:");
    scanf("%d",&n);

    for(i=0;i<n;i++)
    {
        printf("Nhap phan tu a[%d]: ",i);
        scanf("%d", pa+i);
    }
    for(i=0;i<n;i++)
        printf("%4d", *(pa+i));
}
```

Nhắc lại: truyền tham số là mảng cho hàm

Ví dụ:

```
int TinhTong( int pa[], int n);  
void main()  
{  
    int a[10], n;  
    TinhTong(a, n);  
}  
int TinhTong( int pa[], int n)  
{  
    ....  
}
```

Ở ví dụ trên, tham số hình thức của mảng a trong hàm TinhTong khai báo dùng mảng hình thức: **int pa[]**

Tham số hình thức cho hàm là con trỏ

Ví dụ:

```
int TinhTong( int *pa, int n);  
void main()  
{  
    int a[10], n;  
    TinhTong(a, n);  
}  
int TinhTong( int *pa, int n)  
{  
    ....  
}
```

Ở ví dụ trên, tham số hình thức của mảng a trong hàm TinhTong có thể khai báo dùng con trỏ: **int *pa**

SỰ TƯƠNG QUAN MẢNG 2 CHIỀU VÀ CON TRỎ

- Mảng 2 chiều là mảng (1 chiều) của mảng.
- Khi khai báo: `int a[2][3]` thì trong bộ nhớ tạo ra:

Phần tử	<code>a[0][0]</code>	<code>a[0][1]</code>	<code>a[0][2]</code>	<code>a[1][0]</code>	<code>a[1][1]</code>	<code>a[1][2]</code>
Địa chỉ	1	2	3	4	5	6

Tức mảng `a` có 2 phần tử, mỗi phần tử là 1 mảng chứa 3 số nguyên.

- Tên mảng `a` chứa địa chỉ của phần tử đầu tiên.
- Địa chỉ phần tử đầu tiên có kiểu `int[3]`
- Với khai báo: `int a[2][3]` thì:
 - `a` tương đương với `&a[0][0]`
 - `a+1` tương đương với `&a[1][0]`
 -

SỰ TƯƠNG QUAN MẢNG 2 CHIỀU VÀ CON TRỎ

- Xét khai báo : `int a[2][3], *pa;`

`pa = (int*)a;`

Khi đó:

`pa` tương đương với `&a[0][0]`.

`pa+1` tương đương với `&a[0][1]`.

`pa+2` tương đương với `&a[0][2]`.

`pa+3` tương đương với `&a[1][0]`.

`pa+4` tương đương với `&a[1][1]`.

`pa+5` tương đương với `&a[1][2]`.

Chú ý: vì `a` là địa chỉ kiểu `int[3]`, `pa` là con trỏ kiểu `int`.
Nên phải ép kiểu như sau: `pa = (int*)a`

SỰ TƯƠNG QUAN MẢNG 2 CHIỀU VÀ CON TRỎ

- Xét khai báo : `int a[2][3], *pa;`
 `pa = (int*)a;`
 Khi đó:
 - *`pa` tương đương với `a[0][0]`.
 - *`(pa+1)` tương đương với `a[0][1]`.
 - *`(pa+2)` tương đương với `a[0][2]`.
 - *`(pa+3)` tương đương với `a[1][0]`.
 - *`(pa+4)` tương đương với `a[1][1]`.
 - *`(pa+5)` tương đương với `a[1][2]`.

Nhắc lại: Nhập xuất mảng 2 chiều

```
#include <stdio.h>
#include <conio.h>
#define RMax 100
#define CMax 100
void main()
{
    int a[RMax][CMax], n, m, i, j;
    printf("So dong va cot:");
    scanf("%d%d",&n,&m);

    for(i=0;i<n;i++)
        for (j=0;j<m;j++)
        {
            printf("Nhap phan tu a[%d][%d]: ",i,j);
            scanf("%d", &a[i][j]);
        }
    for(i=0;i<n;i++)
    {
        for (j=0;j<m;j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
}
```

Ví dụ: Nhập xuất mảng 2 chiều bằng con trỏ

```
#include <stdio.h>
#include <conio.h>
#define RMax 100
#define CMax 100
void main()
{
    int a[RMax][CMax],n,m, i, j, *pa;
    pa=(int*)a;
    printf("So dong va cot:");
    scanf("%d%d",&n,&m);

    for(i=0;i<n;i++)
        for (j=0;j<m;j++)
        {
            printf("Nhap phan tu a[%d][%d]: ",i,j);
            scanf("%d", pa + i*m +j);
        }
    for(i=0;i<n;i++)
    {
        for (j=0;j<m;j++)
        {
            printf("%d ", *(pa + i*m +j));
        }
        printf("\n");
    }
}
```

Nhắc lại: truyền tham số mảng 2 chiều cho hàm

Ví dụ:

```
int TinhTong( int pa[][100], int n, int m);  
void main()  
{  
    int a[100][100], n,m;  
    TinhTong(a, n, m);  
}  
int TinhTong( int pa[][100], int n, int m)  
{  
    ....  
}
```

Ở ví dụ trên, tham số hình thức của mảng 2 chiều a trong hàm TinhTong khai báo dùng mảng hình thức:

int pa[][100]

Trong thân hàm TinhTong, để truy cập đến phần tử a[i][j] dùng pa[i][j]

Tham số hình thức cho hàm là con trỏ

Ví dụ:

```
int TinhTong( int (*pa)[100], int n, int m);  
void main()  
{  
    int a[100][100], n,m;  
    TinhTong(a, n, m);  
}  
int TinhTong( int (*pa)[100], int n, int m)  
{  
    ....  
}
```

Ở ví dụ trên, tham số hình thức của mảng a trong hàm TinhTong dùng `int (*pa)[100]` là con trỏ kiểu `int[100]`
Trong thân hàm TinhTong, để truy cập đến phần tử `a[i][j]` dùng `pa[i][j]`

Tham số hình thức cho hàm là con trỏ

Ví dụ:

```
int TinhTong( int *pa, int n, int m);  
void main()  
{  
    int a[100][100], n,m;  
    TinhTong(a, n, m);  
}  
int TinhTong( int *pa, int n, int m)  
{  
    ....  
}
```

Ở ví dụ trên, tham số hình thức của mảng a trong hàm TinhTong dùng **int* pa** là con trỏ kiểu int

Trong thân hàm TinhTong, để truy cập đến phần tử a[i][j] dùng ***(pa+i*m+j)**

Nâng cao

- 1) Con trỏ cấp 2
- 2) Mảng con trỏ
- 3) Con trỏ hàm
- 4) Các phép toán trên con trỏ