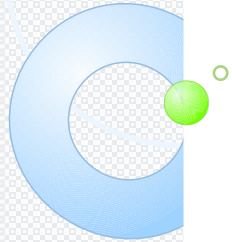


# Chương 4: Mảng (array)

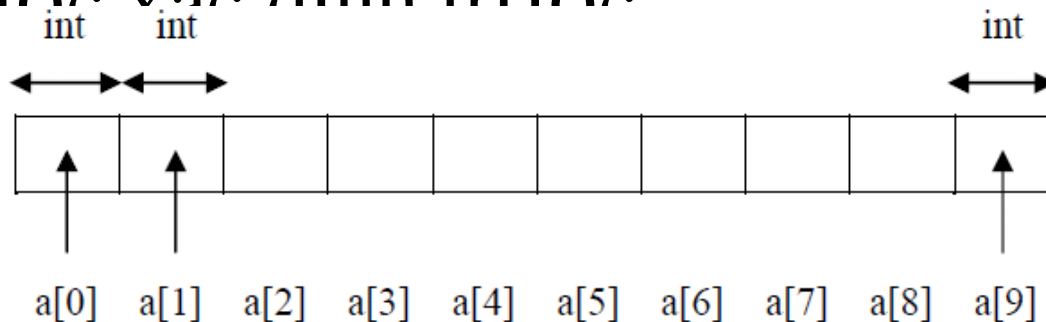


# Agenda

- Khái niệm
- Mảng 1 chiều
- Mảng nhiều chiều

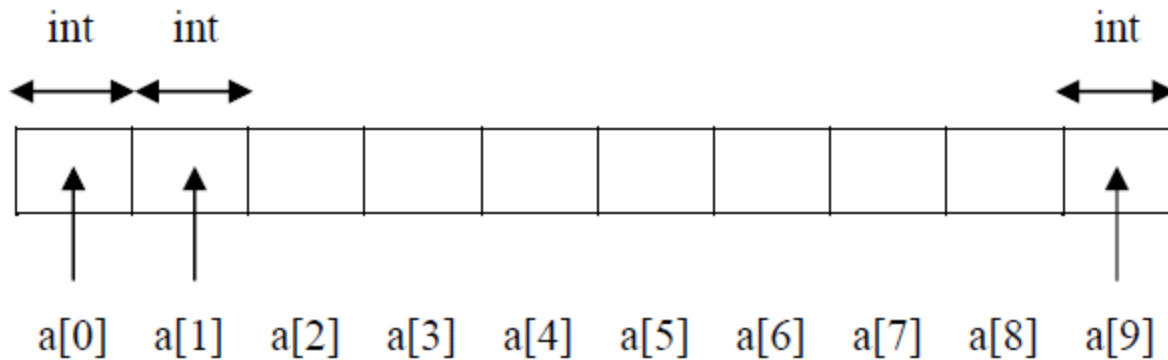
# 1. Khái niệm

- Mảng là một tập hợp các biến có cùng kiểu dữ liệu nằm liên tiếp nhau trong bộ nhớ và được tham chiếu bởi một tên chung (tên mảng).
- Số lượng các phần tử trong mảng được gọi là kích thước của mảng
- Kích thước của mảng là cố định và phải được xác định trước



# 1. Khái niệm

- Mỗi phần tử của mảng được tham chiếu thông qua chỉ mục (index).
- Phần tử đầu tiên có chỉ mục là 0
- Phần tử cuối có chỉ mục là  $n-1$ .



## 2. Mảng 1 chiều

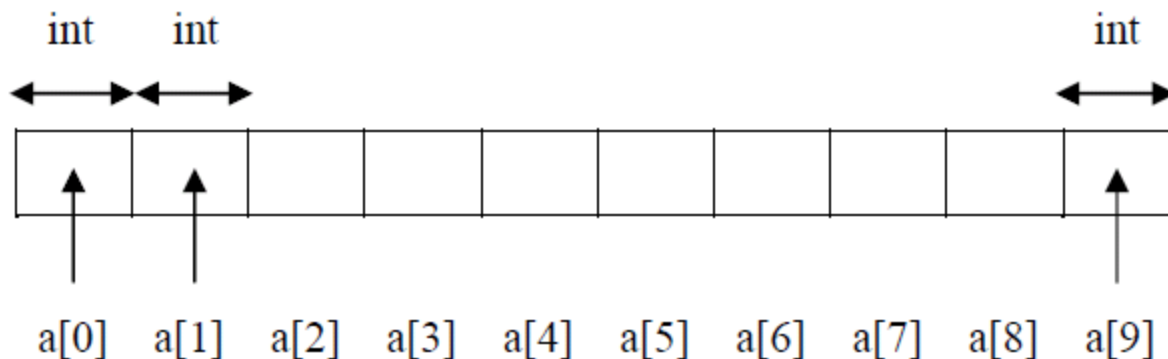
- Khai báo mảng 1 chiều:

**type** **arrayName** [ **arraySize** ];

- **type**: kiểu dữ liệu của mảng
- **arrayName**: tên mảng
- **arraySize**: kích thước của mảng
- Ví dụ:
  - `int c[12];` // c là 1 mảng số nguyên có 12 phần tử
  - `int x[27];` // x là 1 mảng số nguyên có 27 phần tử

## 2. Mảng 1 chiều

- Để truy xuất 1 phần tử trong mảng, ta dùng tên mảng và chỉ mục của phần tử được đặt trong cặp dấu [].
- Ví dụ: Để truy xuất phần tử thứ  $i$  trong mảng  $a \Rightarrow a[i]$



## 2. Mảng 1 chiều

- Mỗi phần tử trong mảng là 1 biến thông thường.
- `int num[3];`
- `num[0] = 2;`
- `num[1] = num[0] + 3`
- `num[2] = num[0] + num[1];`
- `cout << num[1];`

## 2. Mảng 1 chiều – Một số ví dụ

- **Ví dụ 1**: Khởi tạo 1 mảng 10 phần tử. Giá trị của các phần tử = 0

```
#include <iostream.h>
```

```
#include <conio.h>
```

```
void main()
```

```
{
```

```
    int a[10];
```



## 2. Mảng 1 chiều – Một số ví

du

• **Ví dụ 2**: Khởi tạo giá trị các phần tử trong mảng ngay khi khai báo mảng

```
#include <iostream.h>
```

```
#include <conio.h>
```

```
void main()
```

```
{
```

```
    int a[5] = { 5, 6, 7, 8, 9 };
```

## 2. Mảng 1 chiều – Một số ví dụ

- **Ví dụ 3**: Khai báo mảng mà ko xác định trước số phần tử trong mảng

```
int a[] = { 5, 6, 7, 8, 9 };
```

**Note**: Chương trình sẽ tự động đếm số phần tử của mảng từ các giá trị được liệt kê trong phần khai báo phần tử mảng – các giá trị được khai báo trong cặp dấu ngoặc {} và được phân cách bởi dấu “, ”

## 2. Mảng 1 chiều – Một số ví

du

- **Ví dụ 4:** Khai báo mảng có 5 phần tử. Nhập giá trị cho các phần tử trong mảng và tính tổng các phần tử trong mảng.

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
    const int n = 5;
```

```
    int a[n];
```

## 2. Mảng 1 chiều – Một số

**Ví dụ 5:** Khai báo mảng có 10 phần tử, giá trị của các phần tử được gán theo thứ tự từ 1 → 10

```
#include <iostream.h>
#include <iomanip.h>
void main()
{
    const int arraySize = 10;
    int a[arraySize];
```

## 2. Mảng 1 chiều – Một số

**Ví dụ 6:** Khai báo 3 mảng a,b,c kiểu số nguyên có 10 phần tử.

- Giá trị của các phần tử trong mảng 1 là các số lẻ theo thứ tự từ 1 → 19.
- Giá trị của các phần tử trong mảng 2 là các số chẵn theo thứ tự từ 2 → 20.
- $c[i] = a[i] + b[i]$

## 2. Mảng 1 chiều – Một số

- Ví dụ 7: Nhập vào 1 số bất kỳ.  
Tìm trong mảng a vị trí của phần tử có giá trị bằng với số vừa nhập

```
#include <iostream.h>
```

```
#include <stdlib.h>
```

```
void main()
```

```
{
```

```
    randomize();
```

```
    const int arraySize = 10;
```

## 2. Mảng 1 chiều – Một số ví

• Ví dụ 8: Khai báo 1 mảng a kiểu số nguyên có 20 phần tử. Tìm phần tử lớn nhất trong mảng & xuất ra vị trí của phần tử này.

### 3. Mảng nhiều chiều

- C hỗ trợ mảng nhiều chiều. Dạng đơn giản nhất của mảng nhiều chiều là mảng 2 chiều. **Ta có thể xem mảng 2 chiều là một ma trận gồm các hàng và cột.**
- **Mảng 2 chiều thực chất là mảng của những mảng 1 chiều.**



### 3. Mảng 2 chiều

- Ma trận  $a$   $m \times n$  là 1 bảng chữ nhật gồm có  $m * n$  phần tử được viết thành  $m$  dòng (mỗi dòng  $n$  phần tử) và  $n$  cột (mỗi cột  $m$  phần tử).
- Ví dụ: ma trận  $a$   $4 \times 4$

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

# 3. Mảng 2 chiều

- Khai báo mảng 2 chiều

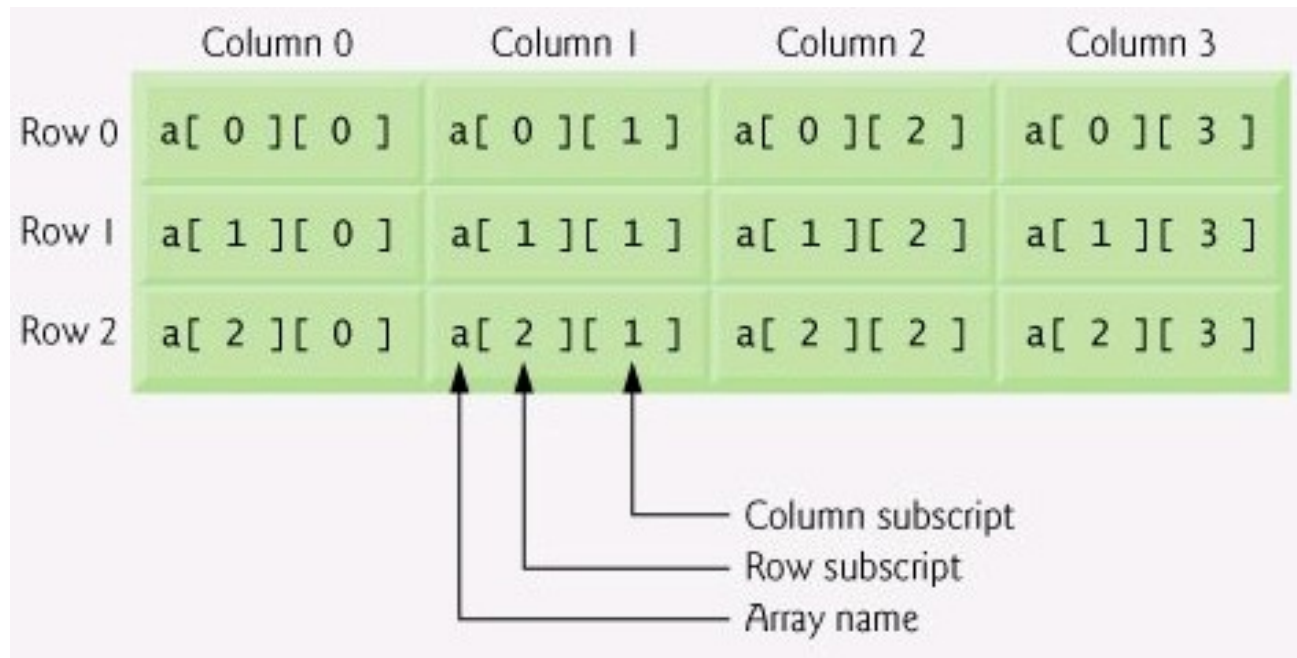
`type arrayName[rows][columns];`

- `type` : Kiểu dữ liệu của mảng
- **arrayName**: tên mảng
- `rows`: số hàng
- `columns`: số cột

# 3. Mảng 2 chiều

- Ví dụ 1: Khai báo 1 mảng 2 chiều a kiểu int có 3 hàng và 4 cột, giá trị các phần tử mảng do người dùng nhập vào từ bàn phím.

**int a[3][4];**



### 3. Mảng 2 chiều

- Ví dụ 2: Khai báo 1 mảng 2 chiều kiểu số nguyên có 4 hàng & 4 cột. Khởi tạo giá trị cho các phần tử trong mảng theo bảng sau & xuất kết quả lên màn hình.

0	1	2	3
0	1	2	3
0	1	2	3
0	1	2	3

### 3. Mảng 2 chiều

```
#include <iostream.h>
```

```
#include <iomanip.h>
```

```
void main()
```

```
{
```

```
    const int n=4;
```

```
    int a[n][n],i,j;
```

```
    for (i = 0;i<n;i++)
```

```
        for (j = 0;j<n;j++)
```

5/4/17

# 3. Mảng 2 chiều

- Có thể thực hiện việc khai báo và khởi tạo các phần tử trong mảng 2 chiều đồng thời với nhau.

```
type arrayName[rows][columns] =  
{ {value1,value2,...,valuen},  
  {value1,value2,...,valuen},  
  {...},  
  {value1,value2,...,valuen}};
```

- Ví dụ: `int b[ 2 ][ 2 ] = { { 1, 2 }, { 3, 4 } };`

# 3. Mảng 2 chiều

- **Lưu ý:** Khi khởi tạo các phần tử mảng đồng thời với việc khai báo mảng, nếu số phần tử trên 1 hàng không được khởi tạo đủ thì chương trình sẽ tự động gán giá trị 0 cho những phần tử chưa được khởi tạo giá trị này.
  - Ví dụ: `int b[ 2 ][ 2 ] = { { 1 }, { 3, 4 } };`
  - `b[ 0 ][ 0 ] = 1`
  - `b[ 0 ][ 1 ] = 0`
  - `b[ 1 ][ 0 ] = 3`
  - `b[ 1 ][ 1 ] = 4`

### 3. Mảng 2 chiều

- Để truy xuất phần tử ở dòng  $i$ , cột  $j$  trong mảng, ta thực hiện như sau: **arrayName[i][j]**
- Ví dụ 2: Khởi tạo 3 mảng 2 chiều lần lượt như sau

```
int array1[ 2 ][ 3 ] = { { 1, 2, 3 }, { 4, 5, 6 } };
```

```
int array2[ 2 ][ 3 ] = { 1, 2, 3, 4, 5 };
```

```
int array3[ 2 ][ 3 ] = { { 1, 2 }, { 4 } };
```



# 3. Mảng 2 chiều

- Output

- Mảng a:

1	2	3
4	5	6

- Mảng b:

1	2	3
4	5	0

- Mảng c:

1	2	0
4	0	0

5/4/17

### 3. Mảng 2 chiều

- Ví dụ 3: Khai báo mảng 2 chiều 4x4 kiểu số nguyên với giá trị của các phần tử do người dùng nhập vào. Tính tổng của từng hàng và của từng cột trong mảng 2 chiều.

### 3. Mảng 2 chiều

- Ví dụ 4: Khai báo mảng 2 chiều kiểu số nguyên với giá trị của các phần tử được khởi tạo ngẫu nhiên. Nhập vào 1 số nguyên bất kỳ. Tìm xem số này xuất hiện ở những vị trí nào trong mảng & xuất hiện bao nhiêu lần.
- Ví dụ 5: Khai báo mảng 2 chiều 3x4 kiểu số nguyên. Viết chương trình để các phần tử trong mảng có giá trị như

sa

2	4	8	16
32	64	128	256
512	1024	2048	5096

# 3. Mảng 2 chiều

- Ví dụ 6: Khai báo mảng 2 chiều kiểu số nguyên với giá trị của các phần tử được khởi tạo ngẫu nhiên tăng dần.