

# C 语言程序设计

## 第六章 数组

## 第六章 数组

### ➤ 6.2 二维数组

**二维数组定义：**

1. 是一个一维数组，但是
2. 其每个数组元素本身，又是一个一维数组  
即是：**一维数组的数组**

**相应地，一个n维数组：**

1. 是一个一维数组，但是
2. 其每个数组元素本身，又是一个n-1维数组

# 二维数组的形式

也必须是 **常量**

```
int sz [5][4];
```

数组的类型

数组名

第1维  
具有5个元素  
的一维数组

第2维  
每个元素又是一个  
具有4个元素  
的一维数组

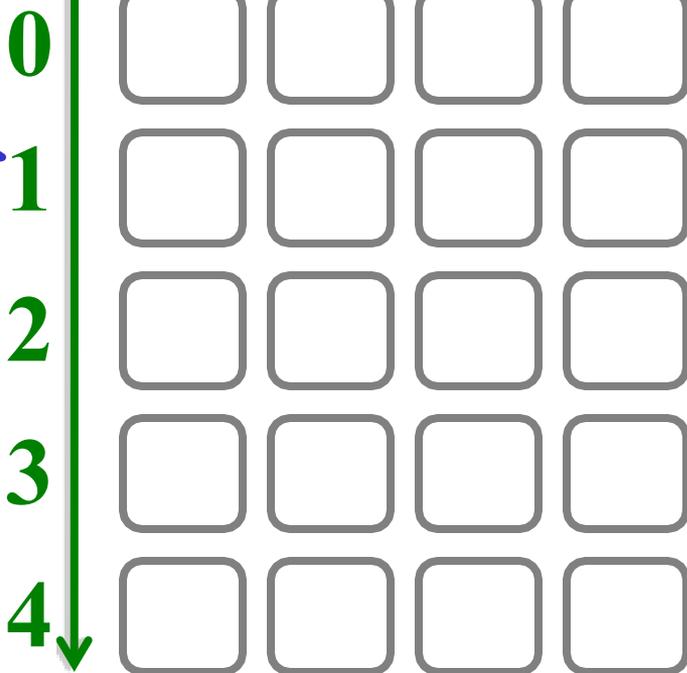
二维数组 可以看作是一个由变量构成的矩阵

```
int sz[5][4];
```

列的编号

0 1 2 3

行的编号



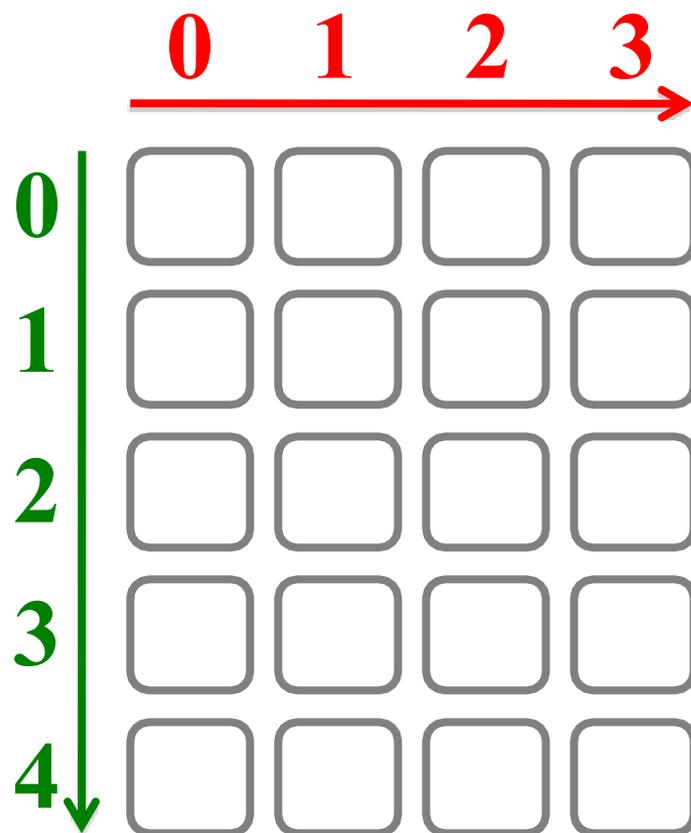
变量的行数

变量的列数

# 二维数组中变量的编号

二维数组中  
变量的编号，由：  
行的编号，和  
列的编号  
两部分组成。

```
int sz[5][4];
```



# 如何访问二维数组中的变量

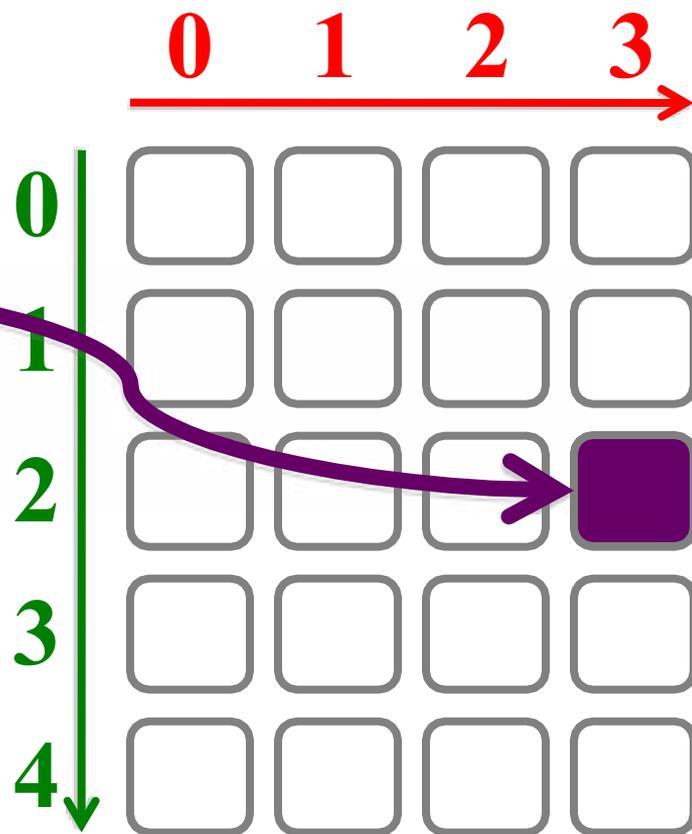
行的编号

...SZ[2][3]...;

数组名

列的编号

```
int sz[5][4];
```

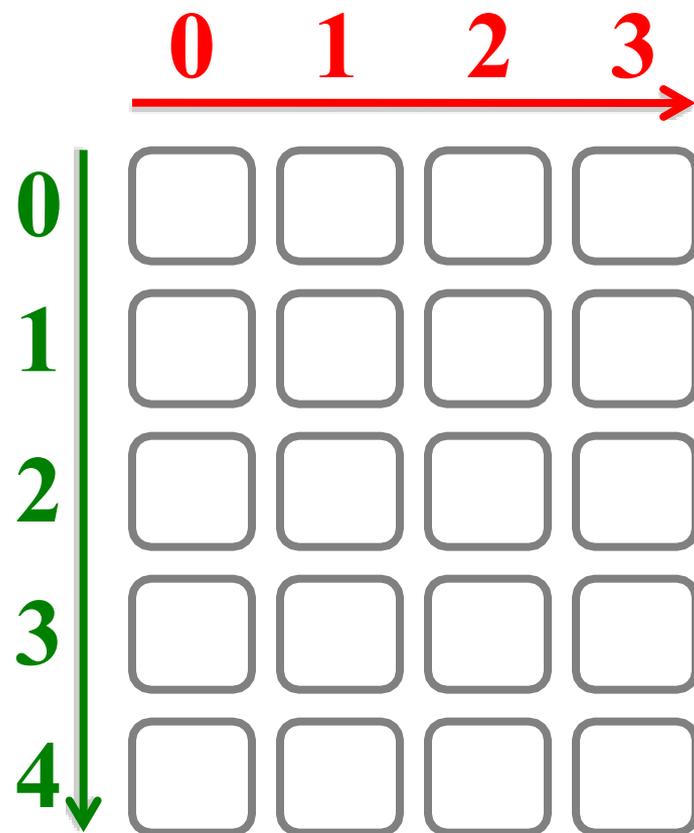


# 二维数组变量赋值的一种特殊方式

- 声明时赋值

```
int sz[5][4] = {  
    {0, 1, 2, 3},  
    {4, 5, 6, 7},  
    {8, 9, 0, 1},  
    {2, 3, 4, 5},  
    {6, 7, 8, 9}  
};
```

```
int sz[5][4];
```



# 二维数组变量赋值的一种特殊方式

## • 声明时赋值

```
int sz[5][4];
```

```
int sz[5][4] = {
```

```
{0, 1, 2, 3},
```

```
{4, 5, 6, 7},
```

```
{8, 9, 0, 1},
```

```
{2, 3, 4, 5},
```

```
{6, 7, 8, 9}
```

```
};
```

第0行变量的值

第1行变量的值

第2行变量的值

第3行变量的值

第4行变量的值

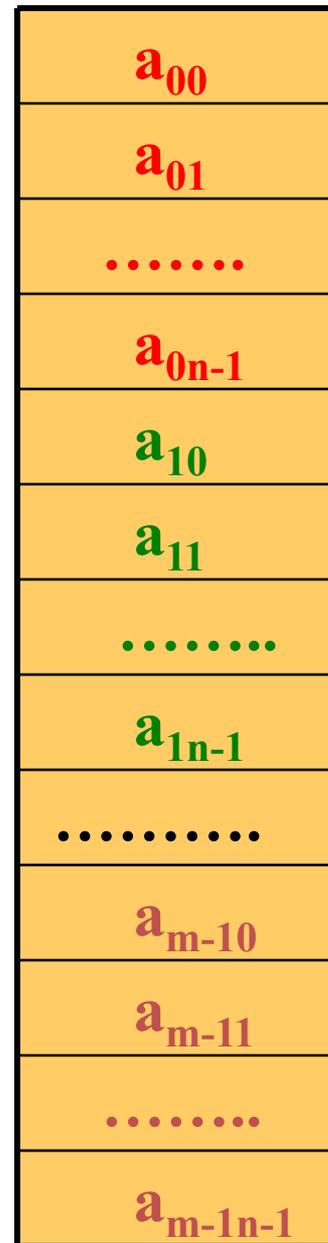
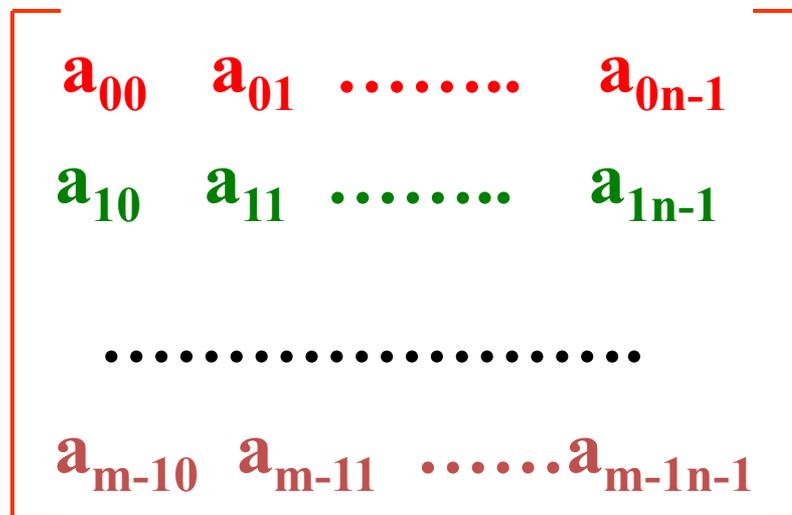
	0	1	2	3
0	0	1	2	3
1	4	5	6	7
2	8	9	0	1
3	2	3	4	5
4	6	7	8	9

# 二维数组：按行的顺序存放

。

按行优先顺序存放

$A[m][n]=$



2000

2002

2004

⋮

注：在这里， $a_{mn}$  代表  $a[m][n]$ 。  
]。

若有数组b[2][3]; 其中数据为:  $\begin{bmatrix} 11 & 12 & 13 \\ 21 & 22 & 23 \end{bmatrix}$

则存放情况如下:

b[0][0]	b[0][1]	b[0][2]	b[1][0]	b[1][1]	b[1][2]
11	12	13	21	22	23
2000	2002	2004	.....		

# 二维数组的输入和输出

数组的输入和输出只能逐个对数组元素进行操作  
(字符数组例外)

```
int b[2][3], i, j;
```

## 输入方法:

输入第i行第j列元素:

```
scanf("%d",&a[i][j]);
```

输入整个数组元素:

```
for (i=0;i<2;i++)  
    for(j=0;j<3;j++)  
        scanf("%d",&a[i][j]);
```

## 输出方法:

输出第i行第j列元素:

```
printf("%d",a[i][j]);
```

输出整个数组元素:

```
for (i=0;i<2;i++)  
    for(j=0;j<3;j++)  
        printf("%d",a[i][j]);
```

# 二维数组 应用举例

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/068004064071007001>