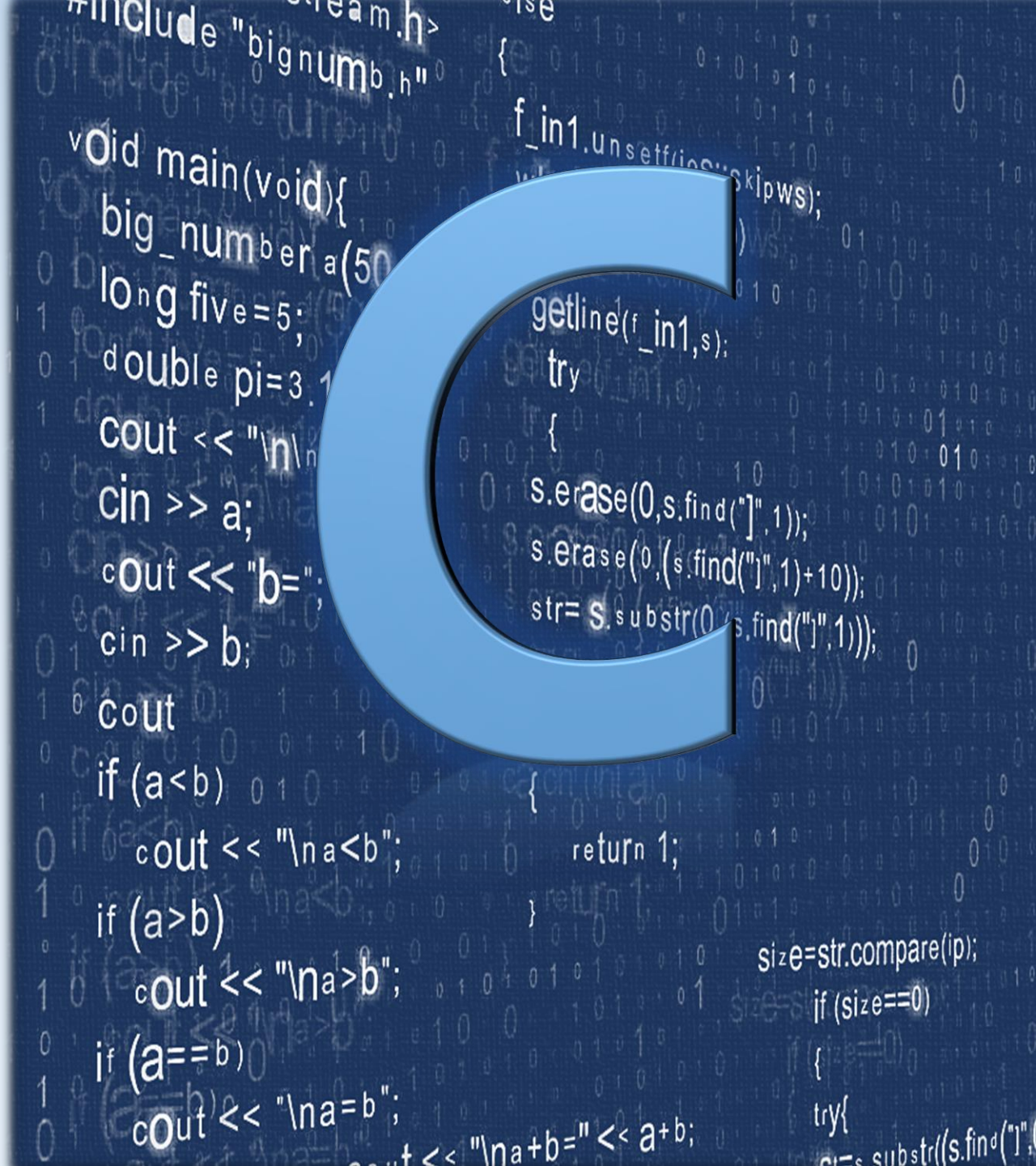



# 《C语言程序设计》

C语言课程组



# 本章授课内容



指针与二维数组



# 指针与二维数组

❖ 指向由m个元素组成的一维数组的指针

定义：数据类型 (\*指针名)[m];

❖ 示例

➤ int a[4];

int (\*p)[4] = &a;

➤ 含义：p是一个指向由4个整型元素构成的数组a的指针;

❖ 错误的赋值方式：

p = a;

p = &a[0]



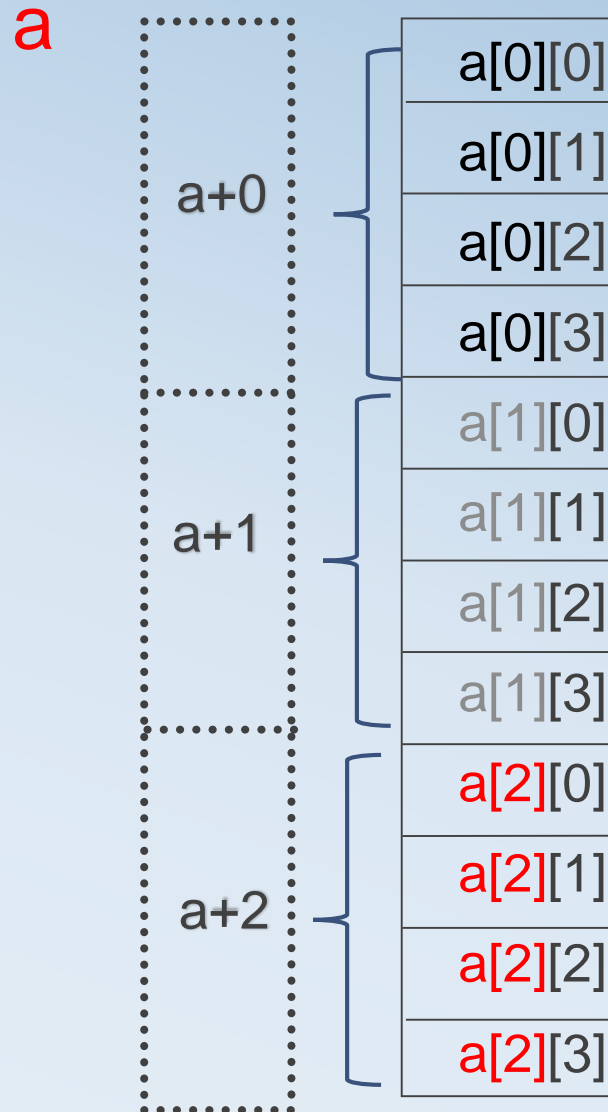
# 指针与二维数组

❖ `int a[3][4];`

- `a` 代表二维数组的首地址，第0行的地址，**行地址**
- `a + i` 代表第*i*行的地址，但并非增加*i*个字节！
- 行地址表示的是一个指向由4列元素组成的一维数组的指针

❖ 如果有 `int (*p)[4];` 可以这样赋值：

`p = a;`

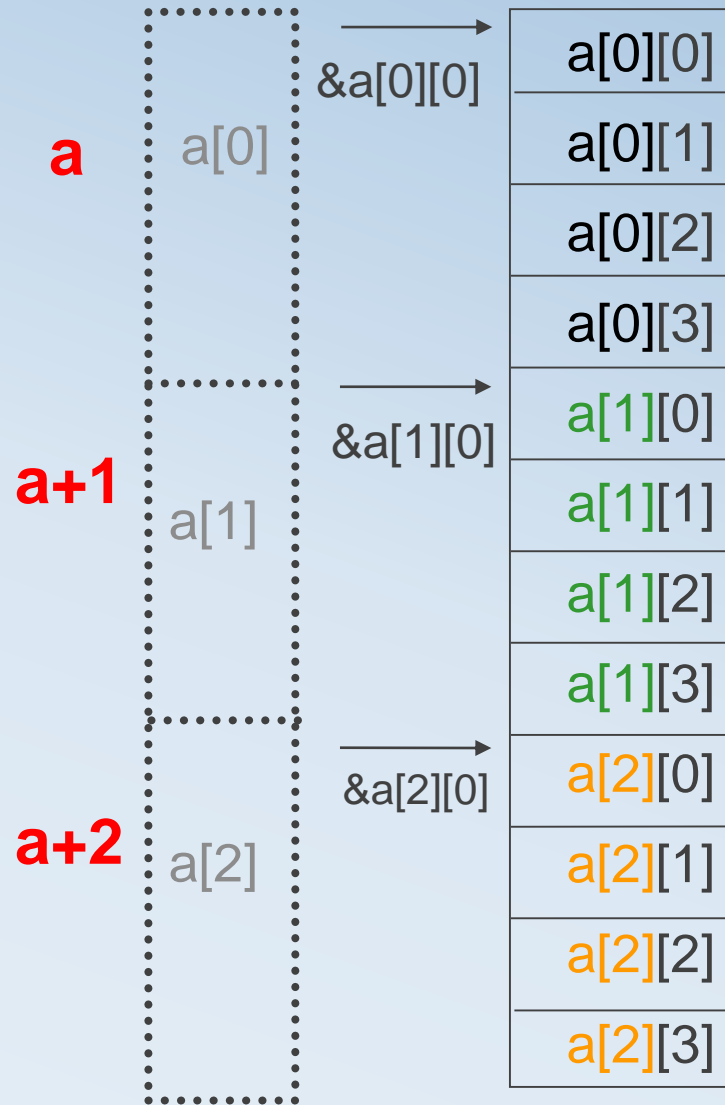


# 指针与二维数组

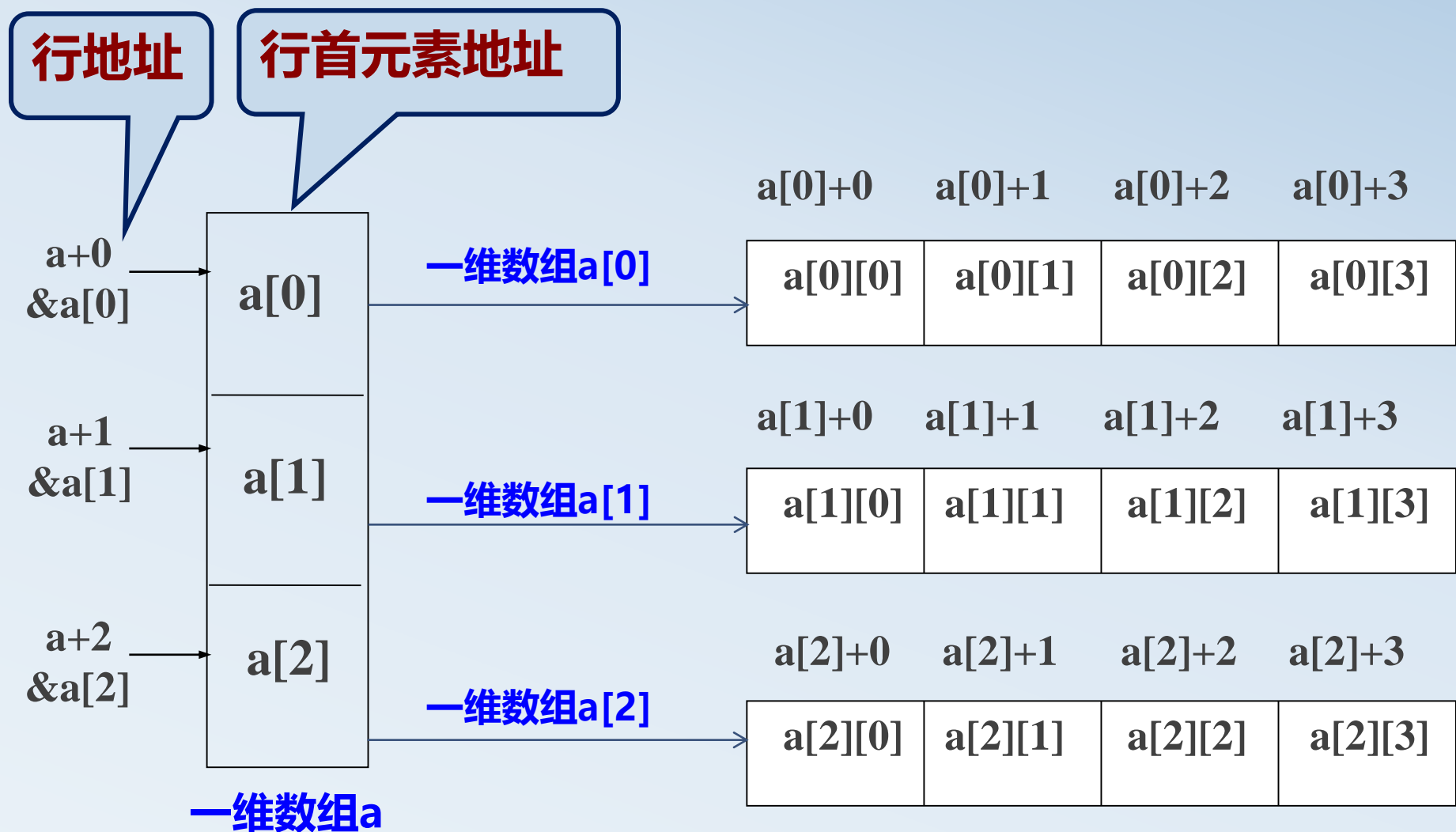
❖ `int a[3][4];`

- **`a[i]`** 即 `*(a+i)` 代表第 *i* 行第 0 列元素的地址，**行首元素地址**
- `*(a+i)+j` 即 `a[i]+j` 代表第 *i* 行第 *j* 列的地址  
`&a[i][j]`
- `*(*(a+i)+j)` 即 `a[i][j]` 代表第 *i* 行第 *j* 列的内容

$a[i] \Leftrightarrow *(a+i) \Leftrightarrow \&a[i][0]$



# 指针与二维数组





# 指针与二维数组

❖ 已知：int a[2][3]; 下列各表达式的含义：

- &a
- a
- &a[0]
- a[0]
- a[0][0]
- &a[0][0]

# 指针与二维数组

❖ 已知：`int a[3][4]={{1,3,5,7},{9,11,13,15},{17,19,21,23}};`

则：

- `a[1]+2`
- `*(a+1)+2`
- `*(a[1]+2)`
- `*(*a+1)+2`
- `(*(a+1))[2]`

以上表达上表示什么？



# 指针与二维数组

❖ 已知：`int a[3][4] = {{1,3,5,7}, {9,11,13,15}, {17,19,21,23}};`

`int (*p)[4] = a;`

`int *q = &a[0][0];` 请用 `p, q` 分别表示下面的表达式：

- `a[1]`
- `a+2`
- `a[2][2]`
- `a[3][1]`

Thank You !