

教学目标

1. 二维数组的定义
2. 二维数组的初始化
3. 二维数组的元素获取
4. 二维数组的输入输出
5. 总结回顾

教学重点

二维数组的基本概念

二维数组：相同数据类型的一维数组的集合

行下标：决定了数组元素所在的行

列下标：决定了数组元素所在的列

越界：超出数组存储的范围

二维数组的定义方式：数据类型 数组名 [行长度][列长度];

二维数组遍历：通过使用二维数组下标，依次获取数组里面的所有元素

总结回顾

二维数组的初始化

- 1、全部初始化为0

```
int a[2][3] = {};
```

```
int a[2][3]; //定义成全局数组
```

- 2、依次初始化每个元素

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

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

- 3、初始化部分元素

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

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

部分初始化中未被初始化的部分会被直接初始化为0

二维数组的遍历输入

```
for (int i = 0; i < 行数; i++)  
{  
    for (int j = 0; j < 列数; j++)  
    {  
        cin >> a[i][j];  
    }  
}
```

二维数组的遍历输出

```
for (int i = 0; i < 行数; i++)  
{  
    for (int j = 0; j < 列数; j++)  
    {  
        cout << a[i][j] << " ";  
    }  
    cout << endl;  
}
```

二维数组和一维数组对比

	二维数组	一维数组
全部初始化为0	<code>int a[10][10] = {};</code> <code>int a[10] = {};</code>	
遍历方式	for循环嵌套	for循环
数组容量	行长度*列长度	数组长度

演示内容

1. 二维数组的定义
2. 二维数组的初始化
3. 二维数组的元素获取
4. 二维数组的输入输出