

数组

► 什么是数组

■ 为什么要有数组

需要统计某公司员工的工资情况，例如计算平均工资、找到最高工资等。

■ 数组的概念

数组是存储同一种数据类型多个元素的集合。

■ 数组的特点

1. 数组中的元素按线性方式排列，可以通过编号来访问数组中的每个元素；
2. 每个值都存储在一个独立的数组元素中，计算机在内存中依次存储数组的各个元素。

► 数组的声明

- 要创建数组，可以使用声明语句。数组声明应该指出以下三点：

1. 存储在每个元素中的值的类型；
2. 数组名；
3. 数组中的元素数。

- 声明数组的格式：

数据类型 数组名[数组大小]；

- 举例：

```
double salaries[20];
```

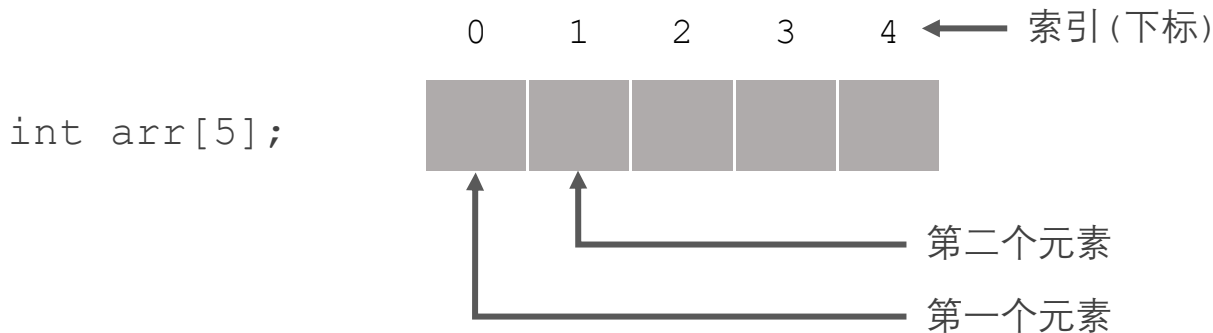
► 数组的访问

■ 访问元素

```
arrayName[index];
```

index 是数组的索引(下标), 是一个从0开始的整数;

数组的第一个元素是 `arrayName[0]`, 最后一个元素是 `arrayName[length-1]`。



► 数组的初始化

- C++ 允许在声明语句中初始化数组元素，只需提供一个用逗号分隔的值列表（初始化列表），并将它们用花括号括起来，可省略等号（=）。

```
int arr[5] = {10, 21, 35, 80};
```

```
int arr[5] {10, 21, 35, 80} // C++11新增
```

- 只有在定义数组时才能使用初始化，此后就不能使用了，也不能将一个数组赋给另一个数组：

```
int cards[4] = {3, 6, 8, 10};
```

```
int hand[4];
```

```
hand[4] = {3, 6, 8, 10}; 错误
```

```
hand = cards; 错误
```

► 数组的初始化

- 初始化数组时，提供的值可以少于数组的元素数目。如果对数组的一部分进行初始化，则编译器将把其他元素设置成0。

```
int arr[5] = {10, 21};
```

```
int arr[5] = {0};
```

```
int arr[5] = {};    // C++11新增
```

- 如果初始化数组时方括号内 ([]) 为空，C++ 编译器将计算元素个数。

```
int things[] = {1, 5, 3, 8};
```

- 列表初始化禁止缩窄转换：

```
int arr[] = {25, 92, 3.0};
```

错误

```
char chs[4] { 'h', 'i', 11220, '\0' };
```

错误

```
char chs[4] { 'h', 'i', 112, '\0' };
```

Thanks

