

8.1 C++指向数组元素的指针

原创 闫小林 C语言入门到精通 2020-12-15 08:30

收录于合集

#小林C++基础

103个 >

点击上方“C语言入门到精通”，选择置顶
第一时间关注程序猿身边的故事



作者

闫小林

白天搬砖，晚上做梦。我有故事，你有酒么？

C++指向数组元素的指针

在C++中，一个变量有地址，一个数组包含若干元素，每个数组元素都在内存中占用存储单元，它们都有相应的地址；指针变量既然可以指向变量，也可以指向数组元素，所谓数组元素的指针就是数组元素的地址。

定义一个整型数组array，它有10个元素：

```
int array[10];
```

定义一个基类型为整型的指针变量p：

```
int *p;
```

将元素array[0]的地址赋给指针变量p，使p指向array[0]：

```
p=&array[0];
```

和C语言一样，在C++中，数组名代表数组中第一个元素的地址，因此也可以这样写：

```
p=&array[0];  
p=array;
```

在定义指针变量时可以直接赋初值，p的初值为array[0]的地址：

```
int *p=&array[0];
```

同样，也可以写成：

```
int *p=array;
```

可以通过指针引用数组元素，假设p已定义为一个基类型为整型的指针变量，并已将一个整型数组元素的地址赋给了它，使它指向某一个数组元素。

如果指针变量p已指向数组中的一个元素，则p+1指向同一数组中的下一个元素，p+i和array+i就是array[i]的地址，或者说，它们指向array数组的第i个元素。

(p+i)或(array+i)是p+i或array+i所指向的数组元素，即array[i]，可以看出，[]实际上是变址运算符。对array[i]的求解过程是：先按array+i×d计算数组元素的地址，然后找出此地址所指向的单元中的值。

指向数组元素的指针变量也可以带下标，如p[i]与*(p+i)等价，引用一个数组元素，可用以下方法：

- 下标法，如array[i]形式
- 指针法，如*(a+i)或*(p+i)。

其中array是数组名，p是指向数组元素的指针变量。

经典案例：C++输出数组中的10个元素。

```
#include<iostream> //预处理  
using namespace std; //命名空间  
int main() //主函数  
{  
    int array[10]; //定义整型数组  
    int i; //定义整型变量  
    int *p=array; //定义指针变量，指针变量p指向数组array的首元素  
    for(i=0;i<10;i++) //for循环键盘输入10个数  
    {  
        cin>>*(p+i);  
    }  
    cout<<"输出结果："<<endl; //提示语句  
    for(p=array;p<(array+10);p++)  
    {  
        cout<<*p<<" "; //p先后指向array[0]-array[10]  
    }  
    cout<<endl; //换行  
    return 0; //函数返回值为0;  
}
```

执行以上程序会输出：

```
2 3 5 6 1 4 5 8 9 4  
输出结果：  
2 3 5 6 1 4 5 8 9 4  
-----  
Process exited after 27.55 seconds with return value 0  
请按任意键继续. . .
```

以上，如果你看了觉得对你有帮助，就给小林点个赞叭，这样小林也有更新下去的动力，跪谢各位父老乡亲们啦~



此人是你关注列表的漏网之鱼



收录于合集 #小林C++基础 103

< 上一篇

7.9 C++指针作函数参数

下一篇 >

8.2 C++指针变量作函数参数接收数组地址