

指针与引用，取地址符与解引用符

指针与引用，取地址符&与解引用^Q符*

引言

探讨一下指针、引用、以及具有两种意思两个符号&和*

引用

引用相当于为对象取一个别名，引用本身不是对象。定义方式：

```
1 int x=4;
2 int &y=x;//y为变量x的引用，变量x有了另外一个名字y，可以通过y来间接访问x
```

指针

指针也可以实现对对象的间接访问，指针与引用有以下区别：

- 指针本身就是对象
- 指针无需在定义时赋初值

1.使用指针获取对象的地址

```
1 int i=12;
2 int *p=&i;//指针p指向i,p的值为i的地址
```

2.使用指针间接访问

```
1 int i=12;
2 int *p=&i;//&为取地址符
3 cout<<*p<<endl;//这里*是解引用符
```

示例

通过下面的实例来理解指针与引用的区别，取地址符&和引用的定义&，指针的定义*与解引用符*。

```
1 #include <iostream>
2 using namespace std;
3 int main(){
4     //整型变量x
5     int x=12;
6
7
8     //x的引用y
9     int &y=x;
10
11     cout<<"x="<<x<<" "<<"y="<<y<<endl;
12     //x=12 y=12
13
14     y++;
15
16     cout<<"x="<<x<<" "<<"y="<<y<<endl;
17     //x=13 y=13
18
19     //指向整型变量的指针p
20     int *p;
21     //p的值为x的地址
22     p=&x;//这里&为取地址符
23
24     cout<<"x="<<x<<" "<<"y="<<y<<" "<<"p="<<p<<endl;
25     //x=13 y=13 p=0x68fedc
26
27     //地址p所指向的值变为10
28     *p=10;//这里为解引用符
29
30     cout<<"x="<<x<<" "<<"y="<<y<<" "<<"p="<<p<<endl;
31     //x=10 y=10 p=0x68fedc
32
33     //p所指对象的引用为z
34     int &z=*p;
35     z=14;
36     cout<<"x="<<x<<" "<<"y="<<y<<" "<<"z="<<z<<" "<<"p="<<p<<endl;
37     //x=14 y=14 z=14 p=0x68fedc
38
39     //整型变量i
40     int i=12;
41     //指针q
42     int *q;
43     //指针q的引用r
44     int *&r=q;
45     //r指向i,也即q指向i
46     r=&i;
47     cout<<"i="<<i<<" "<<"q="<<q<<" "<<"r="<<r<<" "<<endl;
48     //i=12 q=0x68fed8 r=0x68fed8
49
50     //改变r指向的值
51     *r=10;
52     cout<<"i="<<i<<" "<<"q="<<q<<" "<<"r="<<r<<" "<<endl;
53     //i=10 q=0x68fed8 r=0x68fed8
54     return 0;
55 }
```