

```

//bind(less<int>(), 40, placeholders::_1);
//less<int>的operator()的第一个参数绑定为40，那这样当调用less<int>()这个可调用对象时，
//第二个参数，就用这里的 placeholders::_1表示，在调用这个函数时，被传入的第一个参数所取代；
//auto bf = bind(less<int>(), 40, placeholders::_1);
//bf(19);

cishu = count_if(myvector.begin(), myvector.end(), bind(
    less<int>(), //临时对象
    40,
    placeholders::_1
));

//a)bind:函数适配器中的绑定七；
//b)less<int>():是个函数对象（仿函数），这里是个临时对象
//c)count_if:是个算法；
cout << cishu << endl;

```

bind是标准库函数，定义在头文件functional中，可以将bind函数看作一个通用函数适配器，它接受一个可调用对象，生成一个新的可调用对象来适应元对象的参数列表。