

C++标准程序库读书笔记

仿函数（函数对象）

仿函数的定义方法

```
class FunctionObjectType
{
public:
    void operator()()
    {
        statements
    }
};
```

这种定义方式的好处：

- 仿函数比一般函数更灵巧，因为它可以拥有状态。对于仿函数，可以同时拥有两个状态不同的实体，一般函数则不能这样。
- 每个仿函数都有其型别，因此你可以将仿函数的型别当作**template**参数来传递，从而指定某种行为模式，此处还有一个好处，容器型别也会因为仿函数的不同而不同。
- 执行速度上，仿函数通常比函数指针更快。

仿函数是传值，不是传址。因此算法并不会改变岁参数而来的仿函数的标志。

将仿函数以传值方式传递的优点和缺点：

- 优点：可以传递常量和暂时表达式。
- 缺点：无法改变仿函数的状态，算法可以改变仿函数的状态，但你无法存取并改变最终状态，因为改变的只不过是仿函数的副本。

有两个办法可以从运用了仿函数的算法中获取结果或反馈：

1. 以**by reference**的方式传递仿函数
2. 运用**for_each()**算法的回返回值。

判断式和仿函数

判断式：返回布尔值（可转换为**bool**）的一个函数或仿函数。对**STL**而言，并非所有返回布尔值的函数都是合法的判断式。