一．Lambda表达式的构成

C++11中的Lambda表达式**用于定义并创建匿名的函数对象**，以简化编程工作。

Lambda表达式的构成：

[函数对象参数](操作符重载函数参数)mutable ->返回值{函数体}

二．函数对象参数

【】函数对象参数是传递给编译器自动生成的函数对象类的构造函数的。函数对象参数只能使用那些到定义Lambda为止时Lambda所在作用范围内可见的局部变量（包括Lambda所在类的this）。有多种形式：

1. 无参数

没有参数

1. =

通过值传递的方式使用到定义Lambda为止时Lambda所在作用范围内可见的局部变量（包括Lambda所在类的this）。

1. &

通过引用传递的方式使用到定义Lambda为止时Lambda所在作用范围内可见的局部变量（包括Lambda所在类的this）。

1. This

Lambda函数体可以使用Lambda所在类中的成员变量。

1. a

将a按值传递的方式使用，不能修改传递进来的a的拷贝，要修改，需要添加mutable修饰符。

1. &a

将a按引用传递的方式使用。

1. a, &b

a按值传递，b按引用传递。

1. =,&a, &b

除了a,b用引用传递，其他都是值传递。

1. &，a,b

除了a,b用值传递，其他都是引用传递。

三．操作符重载函数参数

标识重载的()操作符的参数，没有参数时，这部分可以省略。参数可以通过按值（如：(a,b)）和按引用（如：(&a,&b)）两种方式进行传递。

四．可修改标识符

Mutable修饰符，加上后可以修改值传递进来的拷贝变量。

五．函数返回值

标识出函数返回值的类型，如：

->int

六．函数体

即使没有实现也不能省略