实验八 异常处理和名空间

**一、异常处理（try、throw、catch）**

**实验内容1:** 求一元二次方程ax2+bx+c=0的实根。如果方程没有实根，则使用C++的异常处理机制输出有关警告。请编程实现。

**实验内容2:**

1. 例程example1.cpp，输出结果是什么？请运行程序验证你的想法。理解throw的数据类型和catch捕获异常类型之间的关系。然后分别做 2) 3) 4)。

OK0!

end0

1. 将f3()中的catch(float)改为catch(double)，思考输出结果是什么？请运行程序验证你的想法。

OK3!

end3

end2

end1

end0

1. 将f2()中的catch(int)改为catch(double)，思考输出结果是什么？请运行程序验证你的想法。

OK2!

end2

end1

end0

1. 将f1()中的catch(char)改为catch(double)，思考输出结果是什么？请运行程序验证你的想法。

OK1!end1

end0

**实验内容3：**

1. 例程example2.cpp，输出结果是什么？请运行程序验证你的想法。

调用fun1(stud1):

构造Tan

构造Li

拷贝构造Tan

1101 Tan

fun1

析构Tan

构造Tan

构造Li

1101 Tan

fun2

析构Li

析构Tan

num=0,error!

析构Li

析构Tan

调用fun0():

调用fun2(stud2):

构造Tan

构造Li

num=0,error!

析构Li

析构Tan

构造Tan

构造Li

构造Tan

1101 Tan

构造Li

析构Li

析构Tan

num=0,error!

析构Li

析构Tan

△调用fun1(stud1);时void fun1(Student stud)中调用了拷贝构造函数，调用fun2(stud2);时void fun2(Student& stud)没用拷贝构造函数，直接传递对象

Class X

{public:

X();

X(const X&);//拷贝构造函数

}

1.为什么拷贝构造参数是引用类型？

当一个对象以传递值的方式传一个函数的时候，拷贝构造函数自动被调用来生成函数中的对象（符合拷贝构造函数调用的情况）。如果一个对象是被传入自己的拷贝构造函数，它的拷贝构造函数将会被调用来拷贝这个对象，这样复制才可以传入它自己的拷贝构造函数，这会导致无限循环直至栈溢出（Stack Overflow）。

2.拷贝构造函数调用的三种形式

1)一个对象作为函数参数，以值传递的方式传入函数体；

2)一个对象作为函数返回值，以值传递的方式从函数返回；

3)一个对象用于给另外一个对象进行初始化（常称为复制初始化）。

总结：当某对象是按值传递时（无论是作为函数参数，还是作为函数返回值），编译器都会先建立一个此对象的临时拷贝，而在建立该临时拷贝时就会调用类的拷贝构造函数。

1. 使用下列语句 ：

Student stud1(1101, "Tan"); //建立对象stud1

Student stud2(0, "Li"); //建立对象stud2

fun1(stud1);

fun1(stud2);

替换主函数中fun0的调用，输出结果是什么？请运行程序验证你的想法。

结果：

构造Tan

构造Li

构造Tan

构造Li

拷贝构造Tan

1101 Tan

fun1

析构Tan

拷贝构造Li

析构Li

析构Li

析构Tan

num=0,error!

析构Li

析构Tan

1. 使用下列语句 ：

Student stud1(1101, "Tan"); //建立对象stud1

Student stud2(0, "Li"); //建立对象stud2

fun2(stud1);

fun2(stud2);

替换主函数中 fun0的调用，输出结果是什么？请运行程序验证你的想法。

结果：

构造Tan

构造Li

构造Tan

构造Li

1101 Tan

fun2

析构Li

析构Tan

num=0,error!

析构Li

析构Tan

**二、名空间（namespace）**

**\*实验内容4:** 组装example3中的文件，观察错误信息。使用namespace 去除名冲突。理解namesapce 的含义和用法。给出修改后的程序。