

第12章 异常处理

主要内容

1. Java异常处理机制
2. Java的异常处理语句

1 Java异常处理机制

- Java的异常处理机制是为了及时有效地处理程序运行中的异常错误。
- 在程序中监视可能发生异常的程序块，一个程序中的所有异常被收集起来放到程序的某一段中处理。

1.1 处理机制的结构

- Java中引入了异常和异常类，并且定义了很多异常类。
- 每个异常类代表一类运行错误，类中包含了该运行错误的信息和处理错误的方法等内容。

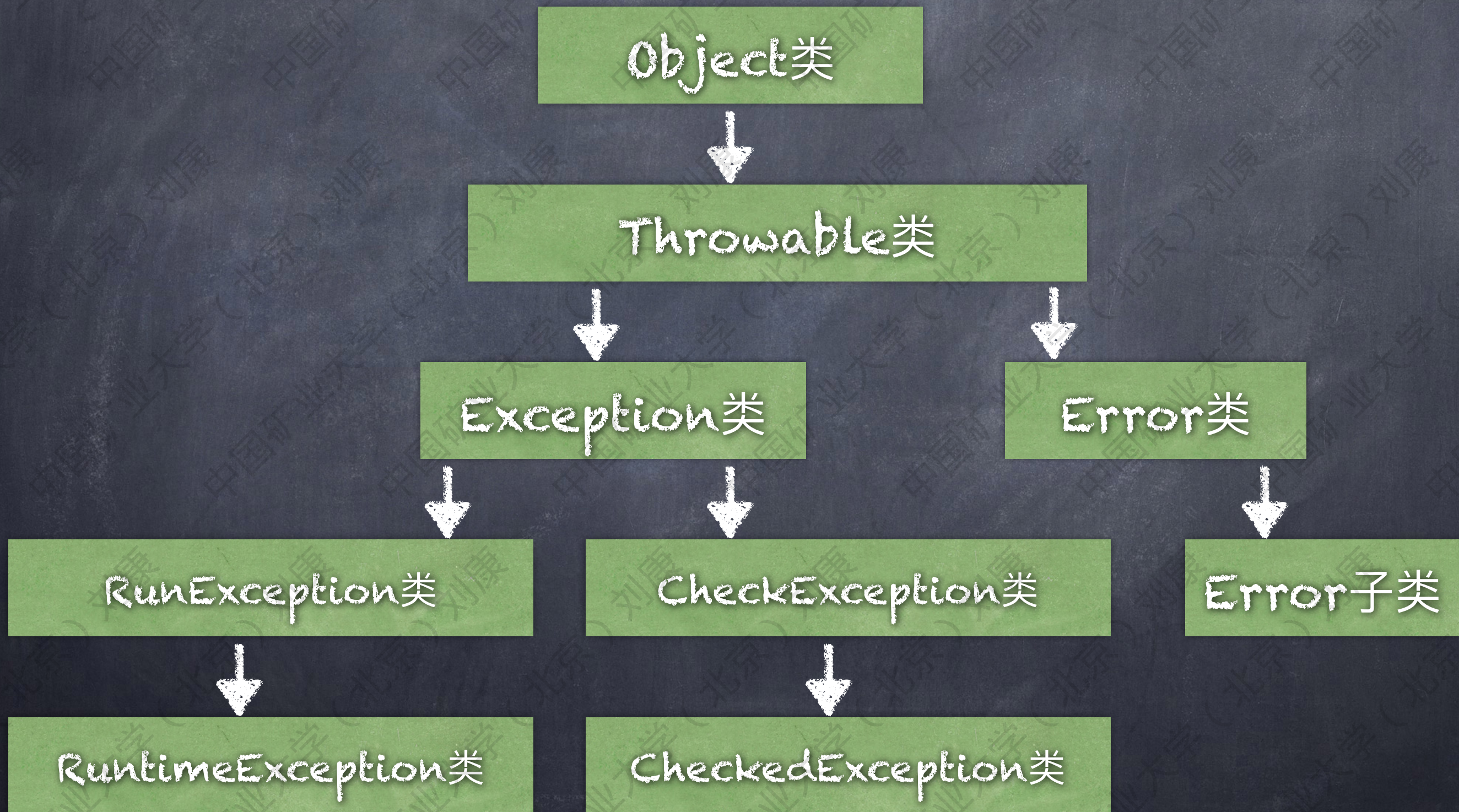
程序运行时期

是否自行处理异常

使用try-catch-finally
语句自行处理

由预设处理方法加以处理

1.2 异常类的继承关系



1. java.util包中的异常

Exception

└─ RuntimeException

└─ EmptyStackException

└─ NoSuchElementException

2. java.io包中的异常

Exception

└─ IOException

├─ EOFException

├─ FileNotFoundException

├─ InterruptedIOException

└─ UTFDataFormatException

3. java.awt包中的异常

Exception

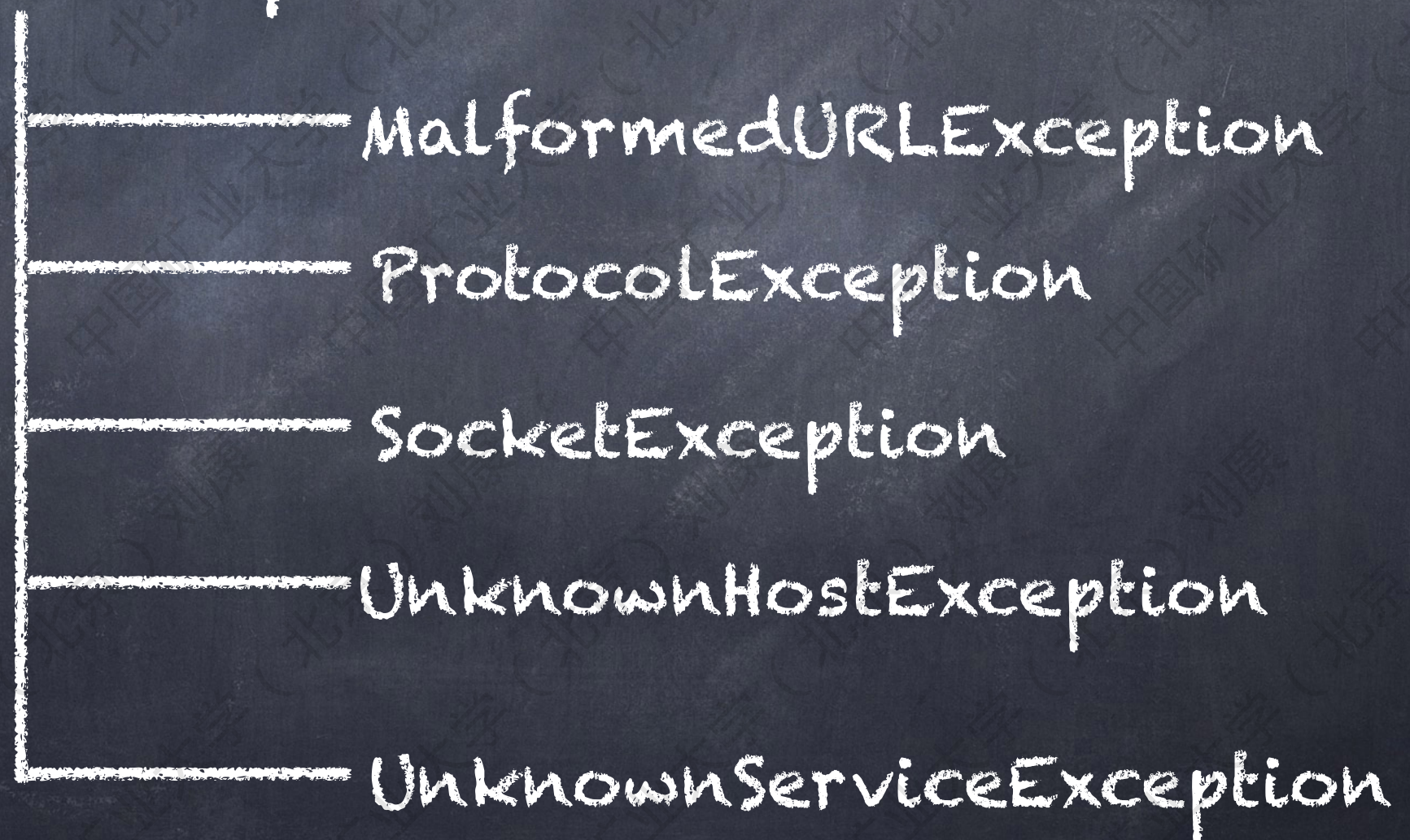
└

AWTException

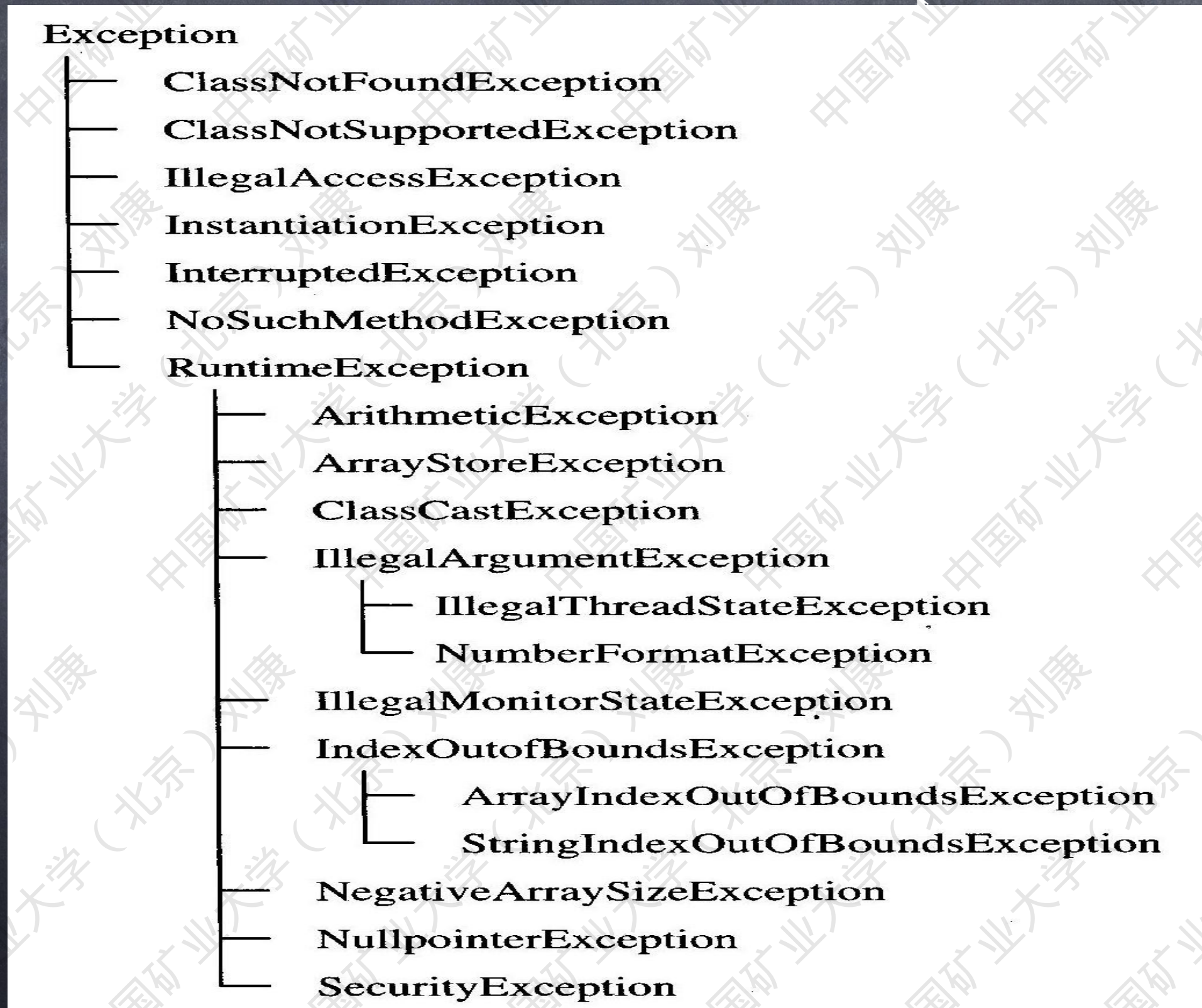
4. java.net包中的异常

Exception

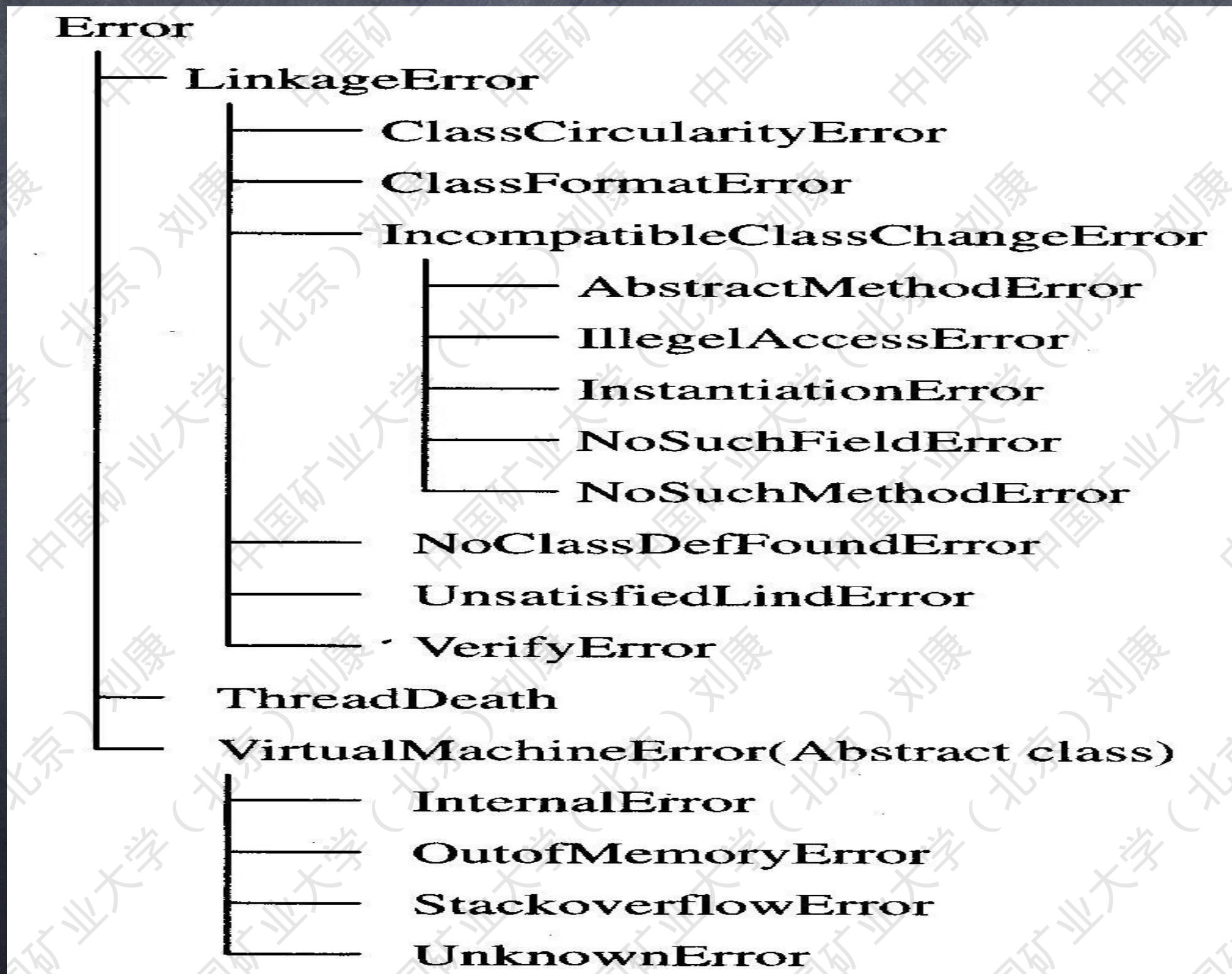
└─ IOException



5. 系统定义的运行异常Exception类



6. 系统定义的运行异常Error类

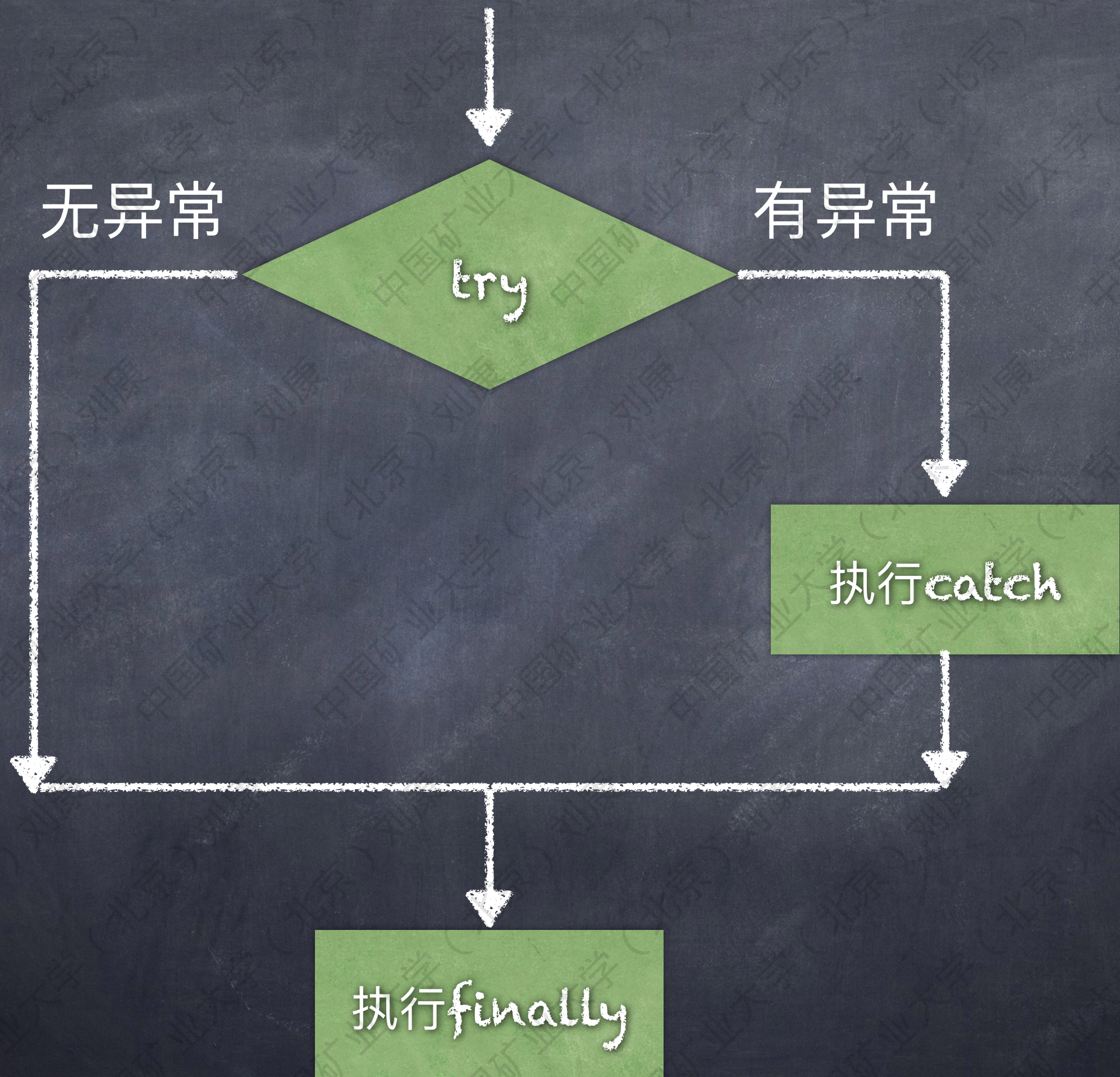


2 异常处理语句

2.1 try-catch-finally语句

在大多数情况下，系统预设的异常处理方法只会输出一些简单的信息到显示器上，然后结束程序的执行。

Java提供try-catch-finally语句，明确地捕捉某种类型的异常，并按要求加以适当的处理，发挥异常处理机制的优点。



- `try-catch-finally` 组合语句来实现抛出异常和捕获异常的功能，其格式如下：

```
try
{
    statements //可能发生异常的代码
}
catch(ExceptionType1 ExceptionObject)
{
    Exception Handling //处理异常代码
}
finally
{
    Finally Handling //无论发生异常都执行的代码
}
```


2.2 抛出异常的throw语句与throws语句

Java的异常是系统在程序运行时抛出的，但也可根据实际情况在程序中抛出一个异常。

Java可以使用throw语句，与throws语句抛出异常。

2.2.1 throw语句

throw语句用来明确地抛出一个异常。throw语句的作用是改变程序的执行流程，使程序跳到相应的异常处理语句中执行。throw语句格式如下：

```
throw ExceptionObject
```


2.2.2 throws语句

在有些情况下，不需要一个方法本身来处理异常，而是希望把异常向上移交给调用这个方法的方法来处理，在这种情况下则需要通过throws语句来实现。throws语句的格式如下：

```
returnType methodName(para1,para2,...) throws exceptionList
```