## 第7章

一、选择题

1．关于异常的含义，下列描述中最正确的一个是（ D ）。

A．程序编译错误 B．程序语法错误

C．程序自定义的异常事件 D．程序编译或运行时发生的异常事件

【解析】异常就是程序编译或运行时发生的异常事件。

2．自定义异常时，可以通过对下列哪一项进行继承？（ C ）

A．Error类 B．Applet类

C．Exception类及其子类 D．AssertionError类

【解析】自定义异常类时，该类必须继承Exception类及其子类。

3．对应try和catch子句的排列方式，下列哪一项是正确的？（ A ）

A．子类异常在前，父类异常在后 B．父类异常在前，子类异常在后

C．只能有子类异常 D．父类和子类不能同时出现在try语句块中

【解析】对应try和catch子句的排列方式，要求子类异常（范围小的异常）在前，父类异常（范围大的异常）在后。

4．运行下面程序时，会产生什么异常？（ A ）

public class X7\_1\_4 {

public static void main(String[] args) {

int x = 0;

int y = 5/x;

int[] z = {1,2,3,4};

int p = z[4];

}

}

A．ArithmeticException B．NumberFormatException

C．ArrayIndexOutOfBoundsException D．IOException

【解析】当程序执行到“int y = 5/x”语句时，发生异常，程序中止执行，因此发生ArithmeticException异常。

5．运行下面程序时，会产生什么异常？（ C ）

public class X7\_1\_5 {

public static void main(String[] args) {

int[] z = {1,2,3,4};

int p = z[4];

int x = 0;

int y = 5/x;

}

}

A．ArithmeticException B．NumberFormatException

C．ArrayIndexOutOfBoundsException D．IOException

【解析】当程序执行到“int p = z[4]”语句时，发生异常，程序中止执行，因此发生ArrayIndexOutOfBoundsException异常。

6．下列程序执行的结果是（ B ）。

public class X7\_1\_6 {

public static void main(String[] args) {

try{

return;

}

finally{

System.out.println("Finally");

}

}

}

A．程序正常运行，但不输出任何结果 B．程序正常运行，并输出Finally

C．编译通过，但运行时出现异常 D．因为没有catch子句，因此不能通过编译

【解析】在执行try-catch-finally语句块时，最后必须执行finally语句块中的内容，而本程序没有异常发生，因此程序正常运行，并输出Finally。

7．下列代码中给出正确的在方法体内抛出异常的是（ B ）。

A．new throw Exception(" "); B．throw new Exception(" ");

C．throws IOException(); D．throws IOException;

【解析】在方法体内抛出异常时只能使用throw，而不能使用throws，另外，“new Exception(" ")”是创建一个异常，因此B是正确的。

8．下列描述了Java语言通过面相对象的方法进行异常处理的好处，请选出不在这些好处范围之内的一项（ C ）

A．把各种不同的异常事件进行分类，体现了良好的继承性

B．把错误处理代码从常规代码中分离出来

C．可以利用异常处理机制代替传统的控制流程

D．这种机制对具有动态运行特性的复杂程序提供了强有力的支持

【解析】异常处理机制不能代替传统的流程控制。

二、填空题

1．异常是在程序编译或运行中所发生的可预料或不可预料的异常事件，出现在编译阶段的异常，称之为 编译时异常 ，出现在运行阶段的异常，称之为 运行时异常 。

2．根据异常的来源，可以把异常分为两种类型： 系统定义的运行时异常 和 用户自定义异常 。

3．所有的Java异常类都是系统类库中的 Exception 类的子类。

4．抛出异常分为 由系统自动抛出异常 、 通过throw抛出异常 以及 通过throws抛出异常 三种情况。

5．Java语言为我们提供了 try…catch 语句和try…catch…finally 语句捕捉并处理异常。

6．一个try块后面可能会跟着若干个 catch 块，每个 catch 块都有一个异常类名作为参数。

7．如果try语句块产生的异常对象被第一个catch块所接收，则程序的流程将 转向第一个catch块 ，catch语句块执行完毕后就 退出当前方法 ，try块中尚未执行的语句和其他的catch块将被 忽略 ；如果try语句块产生的异常对象与第一个catch块不匹配，系统将自动转到 第二个catch块 进行匹配。

8．由于异常对象与catch块的匹配是按照catch块的 先后 顺序进行的，所以在处理多异常时应注意认真设计各catch块的排列顺序。

9．throws语句抛出的异常实际上是由throws语句修饰的方法内部的 throw 语句抛出的，使用throws的主要目的是为了 通知所有预调用此方法的方法 。