

**实 验 报 告**

# 课程名称 Python 语言程序设计

**批阅签名**

**院 系 信息工程学院**

**专 业 数据科学与大数据技术**

**班 级 1801班**

**学 号 180808011022**

**姓 名 刘锐**

**实验名称 实验四 自定义函数的设计和使用**

**实验地点 个人计算机**

**实验成绩**

**指导教师 徐鲁辉**

**学年学期 2019～2020 学年第 2 学期**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 实验6异常处理的编写 | | | | |
| 实验类型 | 设计型 | 实验学时 | 2 | 日期 | 2020年5月24日 |
| 实验目的 | （一）理解什么是异常。  （二）掌握Python内置异常类的使用，掌握如何自定义异常类。  （三）掌握Python中的异常处理结构的使用。  （四）了解断言的使用方法。 | | | | |
| 实验设备 | PC，Windows 7以上版本操作系统，Anaconda。 | | | | |
| （一）异常基本概念  简单地说，异常是指程序运行时引发的错误，引发错误的原因有很多，例如除零、下标  越界、文件不存在、网络异常、类型错误、名字错误、字典键错误、磁盘空间不足等。如果  这些错误得不到正确的处理将会导致程序终止运行，而合理地使用异常处理结果可以使得程  序更加健壮，具有更强的容错性，不会因为用户不小心的错误输入或其他运行时原因而造成  程序终止。  语法错误和逻辑错误不属于异常，但有些语法错误往往会导致异常，例如由于大小写拼  写错误而访问不存在的对象。异常是指因为程序出错而在正常控制流以外采取的行为。当  Python检测到一个错误时，解释器就会指出当前流已无法继续执行下去，这时候就出现了异  常。当程序出现错误，Python会自动引发异常，也可以通过raise显式地抛出异常。  异常处理分为三个阶段：抛出异常、捕获异常、处理异常。  注意：  （1）不建议使用异常来代替常规的检查，如if...else判断。  （2）应避免过多使用异常处理机制，只在确实需要时才使用。  （3）捕捉异常时，应尽量精准，并针对不同类型的异常设计不同的处理代码。  （二）内置异常类和自定义异常类  Python所有异常的基类是BaseException，下面完整展示了Python内置异常类的继承层  次。  （三）异常处理结构  1. try...except结构  try子句中的代码块放置可能出现异常的语句，except子句中的代码块处理异常。  2. try...except…else结构  如果try范围内捕获了异常，就执行except块；如果try范围内没有捕获异常，就执行  else块。  3. 带有多个except的try结构  也可以将要捕获的异常写在一个元组中，可以使用一个except语句捕获多个异常。  4. try...except…finally结构  在该结构中，finally子句无论是否发生异常都会执行，常用来做一些清理工作以释放try  子句中申请的资源。  （四）断言与上下文管理  断言语句的语法是：  assert 表达式, "错误信息"  当判断表达式expression为真时，什么都不做；如果表达式为假，则抛出异常。  assert语句一般用于开发程序时对特定必须满足的条件进行验证，仅当\_\_debug\_\_为True  时有效。当Python脚本以-O选项编译为字节码文件时，assert语句将被移除以提高运行速  度。  二、实验具体过程及结果  （一）编写函数模拟猜数游戏。系统随机产生一个数，并且指定玩家最多可以猜的次数  和数字范围例如[1, 100)，系统会根据玩家的猜测进行提示，玩家则可以根据系统的提示对  下一次的猜测进行适当调整。请使用异常处理机制来编写程序。      （二）用户输入若干个分数，求所有分数的平均分。每输入一个分数后询问是否继续输  入下一个分数，回答“yes”就继续输入下一个分数，回答“no”就停止输入分数。请使用异  常处理机制来编写程序。    （三）编写一个自定义异常类，程序执行过程如下：  判断输入的字符串长度是否小于5，如果小于5，例如输入长度为3，则输出“'The length  ofinputis3,expecting at least 5'”，如果大于5，则输出“'print success'”。    **三、实验心得**  **在编程过程中，对异常处理有了更深刻的理解，对之前所学知识也进行了整体复习，自己的**  **代码水平又有了一定的提高** | | | | | |