实验报告

1 预习内容

* Python是一种高级编程语言，最初由Guido van Rossum于1980年代末期开发，并于1991年首次发布。其设计哲学强调代码的可读性和简洁性。Python具有简单易学、支持多范式编程（如面向对象、函数式编程等）、拥有丰富的标准库以及强大的社区支持等特性。Python被广泛应用于Web开发、数据分析、人工智能、自动化运维、科学计算等领域。常用的开发工具包括PyCharm、Visual Studio Code、Jupyter Notebook等。Python环境的搭建通常通过下载官方提供的安装包或使用Anaconda进行一键安装。
* Python标识符与数据类型标识符是用来给变量、函数、类、模块等命名的名字。Python中标识符的命名规则如下：字母和数字：标识符可以包含字母（A-Z 或 a-z）、数字（0-9）以及下划线（\_）。不能以数字开头：标识符不能以数字开头。区分大小写：例如，variable 和 Variable 是两个不同的标识符。关键字限制：标识符不能是Python的关键字。例如，class, def, if等不能作为标识符使用。特殊字符限制：标识符不能包含空格或特殊字符（如 @, $, % 等）。
* Python数据类型

Python支持多种内建数据类型，常见的包括：

* 数值类型：

整数（int）：如 5, -10

浮点数（float）：如 3.14, -0.001

复数（complex）：如 1 + 2j, -3 + 0.5j

* 字符串（str）：

用单引号或双引号括起来的字符序列，如 'hello', "world"

布尔类型（bool）：

仅有两个值：True 和 False

序列类型：

列表（list）

元组（tuple）

范围（range）

* 集合类型：

集合（set）

冻结集合（frozenset）

在Python中，变量的定义非常简单，只需通过赋值操作即可创建变量

2 程序代码

（2）def is\_armstrong\_number(n, base):  
 digits = [int(digit) for digit in str(n)]  
 sum\_of\_powers = sum(digit \*\* base for digit in digits)  
 return sum\_of\_powers == n  
  
def find\_armstrong\_numbers(limit):  
 armstrong\_numbers = []  
 for num in range(limit + 1):  
 num\_digits = len(str(num))  
 if is\_armstrong\_number(num, num\_digits):  
 armstrong\_numbers.append(num)  
 return armstrong\_numbers  
  
def check\_if\_armstrong\_number(k):  
 num\_digits = len(str(k))  
 return is\_armstrong\_number(k, num\_digits)  
  
# 查找0到10000之间的自幂数  
armstrong\_numbers = find\_armstrong\_numbers(10000)  
print("0到10000之间的自幂数有：", armstrong\_numbers)  
  
# 用户输入  
k = int(input("请输入一个整数k："))  
if check\_if\_armstrong\_number(k):  
 print(f"{k} 是一个自幂数。")  
else:  
 print(f"{k} 不是一个自幂数。")

（3）def generate\_special\_matrix(k):  
 if k % 2 == 0:  
 raise ValueError("k must be an odd number")  
  
 matrix = [[0] \* k for \_ in range(k)]  
 num = 1  
  
 for i in range(k):  
 if i % 2 == 0:  
 for j in range(k):  
 matrix[i][j] = num  
 num += 1  
 else:  
 for j in range(k - 1, -1, -1):  
 matrix[i][j] = num  
 num += 1  
  
 return matrix  
  
# 示例  
k = 3  
special\_matrix = generate\_special\_matrix(k)  
for row in special\_matrix:  
 print(row)

3 成果展示

0到10000之间的自幂数有： [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407, 1634, 8208, 9474]

请输入一个整数k：66

66 不是一个自幂数。

[1, 2, 3]

[6, 5, 4]

[7, 8, 9]

4 心得体会

通过本次课程设计，我加深了对Python编程的理解和应用能力。特别是在实现自幂数检测和特殊矩阵生成的过程中，我掌握了Python中函数定义、循环结构、列表操作等基本编程技巧。同时，我也体会到编写高质量代码的重要性，代码的可读性和简洁性不仅提高了开发效率，也方便了后期的维护和扩展。在今后的学习和工作中，我将继续努力提升自己的编程能力，争取在更多的项目中取得更好的成绩。

说明：所有的蓝色字都属于说明性文字，请在提交版本中删除。（1）标题使用小四号黑体；（2）正文使用五号宋体；（3）行间距使用1.5倍行距；（4）正文字体颜色请使用黑色，插图等颜色不限；（5）图必须有图题，表必须有表头。

课程设计参考格式

1 功能设计

此部分内容：请描述课程设计实现的功能。注意：是已经完成的功能，不是构思的功能。建议：如果有拓展功能，请以清晰的表达形式提请评阅老师。

2 软件框架

此部分内容：描述软件的各个功能块及各个功能块之间的关系。

3 功能模块的实现

此部分内容：对各个功能块的具体实现描述，（注意：不允许使用代码描述，可使用自然语言或者流程框图等进行描述）。

4 成果展示

此部分内容：对所编程序成果展示（注意：不要超过3页）。如果对成果有需要分析之处，请分析相应成果。

5 心得体会

此部分内容：课程设计中的体会。

6 参考文献

此部分内容非必需。请列出相关参考文献。

附录

此部分内容：提交必要的关键程序代码。

# 封面格式

出于保留格式的需求，相应封面在后续页。

|  |  |
| --- | --- |
| **成 绩** |  |

****

**中 国 矿 业 大 学**

**China University of Mining and Technology**

**20 - 20 学年**

**《Python编程实践》**

**课 程 设 计 报 告**

院 系： 信息与控制工程学院

班 级：

学 号：

日 期：

姓 名：

签 名：

注：若要求交打印稿，（1）请在姓名处手写签名 ，（2）并删除本行文字

|  |  |
| --- | --- |
| **成 绩** |  |

****

**中 国 矿 业 大 学**

**China University of Mining and Technology**

**20 - 20 学年**

**《Python编程实践》**

**第 次实验报告**

院 系： 信息与控制工程学院

班 级：

学 号：

日 期：

姓 名：

签 名：

注：若要求交打印稿，（1）请在姓名处手写签名 ，（2）并删除本行文字