

21计科03-B20210302301杨韬-实验五

实验五 Python数据结构与数据模型

班级： 21计科3

学号： B20210302301

姓名： 杨韬

Github地址： <https://github.com/bairimenglin/yangtao/tree/main/experiment>

CodeWars地址： <https://www.codewars.com/users/bairimenglin>

实验目的

1. 学习Python数据结构的高级用法
2. 学习Python的数据模型

实验环境

1. Git
2. Python 3.10
3. VSCode
4. VSCode插件

实验内容和步骤

第一部分

在[Codewars网站](#)注册账号，完成下列Kata挑战：

第一题：停止逆转我的单词

难度： 6kyu

编写一个函数，接收一个或多个单词的字符串，并返回相同的字符串，但所有5个或更多的字母单词都是相反的（就像这个Kata的名字一样）。传入的字符串将只由字母和空格组成。只有当出现一个以上的单词时，才会包括空格。

例如：

```
spinWords( "Hey fellow warriors" ) => returns "Hey wollef sroirraw"

spinWords( "This is a test") => returns "This is a test"

spinWords( "This is another test" )=> returns "This is rehtona test"
```

代码提交地址：

<https://www.codewars.com/kata/5264d2b162488dc400000001>

提示：

- 利用str的split方法可以将字符串分为单词列表

例如：

```
words = "hey fellow warrior".split()

# words should be ['hey', 'fellow', 'warrior']
```

- 利用列表推导将长度大于等于5的单词反转(利用切片word[::-1])
- 最后使用str的join方法连结列表中的单词。

第二题：发现离群的数(Find The Parity Outlier)

难度：6kyu

给你一个包含整数的数组（其长度至少为3，但可能非常大）。该数组要么完全由奇数组成，要么完全由偶数组成，除了一个整数N。请写一个方法，以该数组为参数，返回这个"离群"的N。

例如：

```
[2, 4, 0, 100, 4, 11, 2602, 36]

# Should return: 11 (the only odd number)


[160, 3, 1719, 19, 11, 13, -21]

# Should return: 160 (the only even number)
```

代码提交地址：

<https://www.codewars.com/kata/5526fc09a1bbd946250002dc>

第三题： 检测Pangram

难度：6kyu

pangram是一个至少包含每个字母一次的句子。例如，"The quick brown fox jumps over the lazy dog"这个句子就是一个pangram，因为它至少使用了一次字母A-Z（大小写不相关）。

给定一个字符串，检测它是否是一个pangram。如果是则返回 `True`，如果不是则返回 `False`。忽略数字和标点符号。

代码提交地址：

<https://www.codewars.com/kata/545cedaa9943f7fe7b000048>

第四题： 数独解决方案验证

难度：6kyu

数独背景

数独是一种在 9x9 网格上进行的 game。游戏的目标是用 1 到 9 的数字填充网格的所有单元格，以便每一列、每一行和九个 3x3 子网格（也称为块）中的都包含数字 1 到 9。更多信息请访问：

<http://en.wikipedia.org/wiki/Sudoku>

编写一个函数接受一个代表数独板的二维数组，如果它是一个有效的解决方案则返回 true，否则返回 false。数独板的单元格也可能包含 0，这将代表空单元格。包含一个或多个零的棋盘被认为是无效的解决方案。棋盘总是 9 x 9 格，每个格只包含 0 到 9 之间的整数。

代码提交地址：

<https://www.codewars.com/kata/63d1bac72de941033dbf87ae>

第五题： 疯狂的彩色三角形

难度： 2kyu

一个彩色的三角形是由一排颜色组成的，每一排都是红色、绿色或蓝色。连续的几行，每一行都比上一行少一种颜色，是通过考虑前一行中的两个相接触的颜色而产生的。如果这些颜色是相同的，那么新的一行就使用相同的颜色。如果它们不同，则在新的一行中使用缺失的颜色。这个过程一直持续到最后一行，只有一种颜色被生成。

例如：

Colour here:	G G	B G	R G	B R
Becomes colour here:	G	R	B	G

一个更大的三角形例子：

R R G B R G B B
R B R G B R B
G G B R G G
G R G B G
B B R R
B G R
R B

你将得到三角形的第一行字符串，你的工作是返回最后的颜色，这将出现在最下面一行的字符串。在上面的例子中，你将得到 "RRGBRBBB"，你应该返回 "G"。

限制条件： $1 \leq \text{length}(\text{row}) \leq 10^5$

输入的字符串将只包含大写字母'B'、'G'或'R'。

例如：

```
triangle('B') == 'B'

triangle('GB') == 'R'

triangle('RRR') == 'R'

triangle('RGBG') == 'B'

triangle('RBRGBRB') == 'G'

triangle('RBRGBRBGGRRRBGBBBGG') == 'G'
```

代码提交地址：

<https://www.codewars.com/kata/5a331ea7ee1aae8f24000175>

提示：请参考下面的链接，利用三进制的特点来进行计算。

<https://stackoverflow.com/questions/53585022/three-colors-triangles>

第二部分

使用Mermaid绘制程序流程图

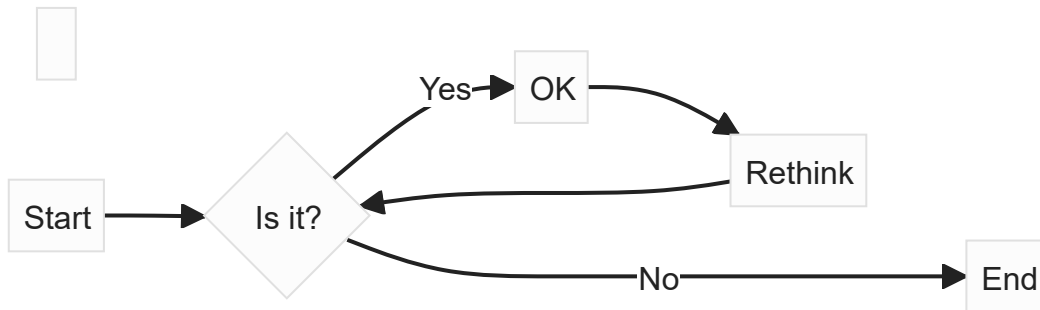
安装VSCode插件：

- Markdown Preview Mermaid Support
- Mermaid Markdown Syntax Highlighting

使用Markdown语法绘制你的程序绘制程序流程图（至少一个），Markdown代码如下：

"/Experiments/img/2023-08-05-22-00-00.png" 未创建，点击以创建。

显示效果如下：



查看Mermaid流程图语法-->[点击这里](#)

使用Markdown编辑器（例如VScode）编写本次实验的实验报告，包括[实验过程与结果](#)、[实验考查](#)和[实验总结](#)，并将其导出为 **PDF格式** 来提交。

实验过程与结果

请将实验过程与结果放在这里，包括：

- [第一部分 Codewars Kata挑战](#)
- **第一题：停止逆转我的单词**

```
def spin_words(sentence):  
  
    result = sentence.split()  
  
    for i in range(len(result)):  
  
        t=result[i]  
  
        if len(t)>=5:  
  
            t1=t[::-1]  
  
            result[i]=t1  
  
    sentence = ' '.join(result)
```

```
return sentence
```

输入:"Welcome"

输出:"emocleW"

- **第二题：发现离群的数(Find The Parity Outlier)**

```
def find_outlier(integers):  
  
    odd_count = sum(1 for num in integers if num % 2 != 0)  
  
    even_count = sum(1 for num in integers if num % 2 == 0)  
  
    # 找到不符合数组特性的数并返回  
  
    for num in integers:  
  
        if odd_count > even_count:  
  
            if num % 2 == 0:  
  
                return num  
  
        else:  
  
            if num % 2 != 0:  
  
                return num
```

输入:[2, 4, 0, 100, 4, 11, 2602, 36]

输出:11

- **第三题：检测Pangram**

```
import string
```

```
def is_pangram(s):

    s = s.lower()

    for char in 'abcdefghijklmnopqrstuvwxyz':

        if char not in s:

            return False

    return True
```

输入:"The quick brown fox jumps over the lazy dog "

输出:True

- **第四题： 数独解决方案验证**

```
def validate_sudoku(board):

    elements = set(range(1, 10))

    # row

    for b in board:

        if set(b) != elements:

            return False

    # column

    for b in zip(*board):

        if set(b) != elements:

            return False

    # magic squares

    for i in range(3, 10, 3):

        for j in range(3, 10, 3):
```



```

        if elements != {(board[q][w]) for w in range(j-3, j) for q in range(i-3, i)}:

            return False

    return True

```

- **第五题： 疯狂的彩色三角形**

```

def triangle(row):

    while len(row) > 1:

        new_row = ""

        for i in range(len(row) - 1):

            new_row += get_missing_color(row[i], row[i + 1])

        row = new_row

    return row


def get_missing_color(color1, color2):

    if color1 == color2:

        return color1

    colors = {'R', 'G', 'B'}

    colors.remove(color1)

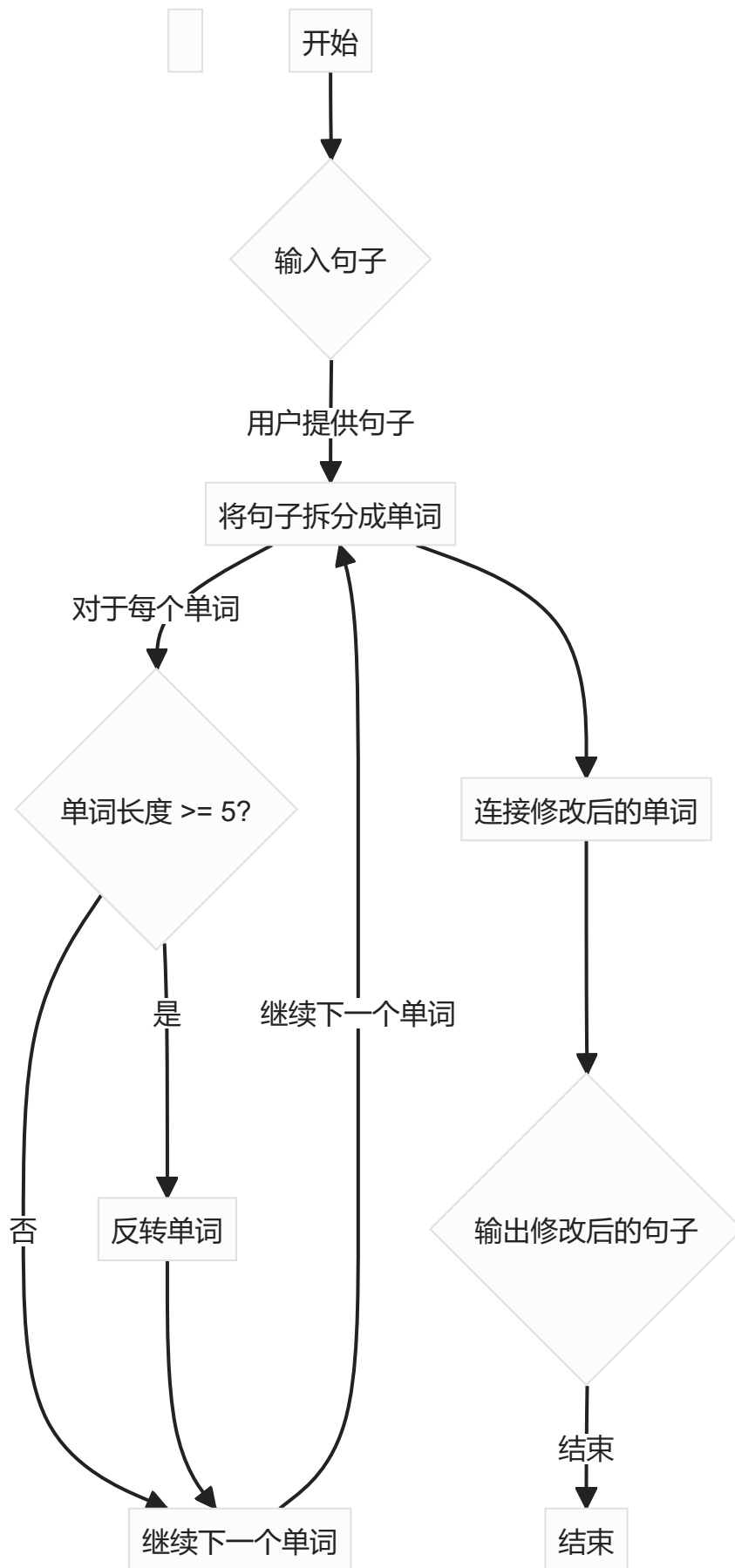
    colors.remove(color2)

    return colors.pop()

```

- [第二部分 使用Mermaid绘制程序流程图](#)

• 第一题：停止逆转我的单词



注意代码需要使用markdown的代码块格式化，例如Git命令行语句应该使用下面的格式：

"/Experiments/img/2023-07-26-22-48.png" 未创建，点击以创建。

显示效果如下：

```
git init

git add .

git status

git commit -m "first commit"
```

如果是Python代码，应该使用下面代码块格式，例如：

"/Experiments/img/2023-07-26-22-52-20.png" 未创建，点击以创建。

显示效果如下：

```
def add_binary(a,b):

    return bin(a+b)[2:]
```

代码运行结果的文本可以直接粘贴在这里。

注意：不要使用截图，因为Markdown文档转换为Pdf格式后，截图会无法显示。

实验考查

请使用自己的语言并使用尽量简短代码示例回答下面的问题，这些问题将在实验检查时用于提问和答辩以及实际的操作。

1. 集合（set）类型有什么特点？它和列表（list）类型有什么区别？

特点：

集合是无序的（无顺序）；集合中的元素是唯一的；集合类型支持集合数学运算（如并集、交集、差集等）；集合类型支持元素查找（通过 in 关键字）。

区别：

集合类型是无序的，列表类型是有序的。集合类型支持集合运算，如并集、交集、差集等，而列表类型不支持集合运算。集合类型支持元素查找，而列表类型不支持元素查找。集合类型在创建时不需要指定元素个数，而列表类型在创建时需要指定元素个数。

2. 集合 (set) 类型主要有那些操作？

添加元素 (add)：向集合中添加一个元素。删除元素 (discard)；更新元素 (update)：集合并集 (union)；集合差集 (difference)；集合长度 (len)；集合排序 (sorted)；集合 (clear)；集合 (pop)；集合 (count)；集合 (copy)；集合 (extend)；集合 (remove)；集合 (set)。

3. 使用 * 操作符作用到列表上会产生什么效果？为什么不能使用 * 操作符作用到嵌套的列表上？使用简单的代码示例说明。

操作符可以用来创建一个新的列表，该列表是将原来的列表重复指定的次数。如果要将操作符作用到嵌套的列表上，则会导致语法错误。例如，以下代码将引发错误：

```
nested_list = 1, 2, [3, 4]
```

```
result = nested_list * 3
```

4. 总结列表，集合，字典的解析 (comprehension) 的使用方法。使用简单的代码示例说明。

列表、集合和字典都可以使用解析来创建。解析是一种简洁的创建列表、集合或字典的方法，它允许你使用 for 循环和 if 语句来筛选和转换数据。如：

列表

```
my_list = [1, 2, 3, 4, 5]
```

```
result = [x * 2 for x in my_list]
```

```
print(result)
```

集合

```
my_list = [1, 2, 3, 4, 5]
```

```
result = {x * 2 for x in my_list}
```

```
print(result)
```

字典解析

```
my_list = ['a', 'b', 'c', 'd', 'e']
```

```
result = {x: x.upper() for x in my_list}
```

```
print(result)
```

实验总结

在本次实验中，我深入学习了Python中数据结构的高级用法以及数据模型的相关概念。列表解析能够以一行代码生成新的列表，使代码更为紧凑。集合解析是创建集合的快速方式，对于生成无重复元素的集合非常方便。字典解析使得创建字典变得简单，通过一行代码就能够生成键值对。这些让我更深入地理解了Python中高级数据结构的使用和数据模型的概念，使我能够更加灵活地处理和操作数据。这些知识不仅提高了我的代码效率，也为我理解更复杂的Python程序打下了基础。通过实际的代码练习，我对这些概念有了更深刻的理解。