《Python程序设计》课程实验报告

信息 学院 智能科学与技术 专业 2020 级

实验时间 2021 年 12 月 1 日

姓名 ***Steven*** 学号

实验名称 作业九

实验成绩

一、实验目的

练习使用Python打开文件、读取内容、修改内容的方法。

学会使用第三方库处理二进制文件。

学会利用编程思想处理文档及内容。

二、实验仪器设备及软件

电脑，Pycharm ，Anaconda Python 3.8

三、实验方案

1. 交叉合并文件，若有一个文件提前合并完了，则之后都合并那个没合并完的。

这个题可以利用队列的思想：依次让字符串出队列，且只有当两个字符串队列不为空的时候才执行出队、写入文件。

2. 读取第一题的代码，在每一行的后面都添加行号，并使行号对齐。

其实就是设定一个长度，这个长度比每行都长，长度不够的以空格补齐，然后在最长长度之后添加一个行号数字即可。

3. 读取并输出一个Word文件里的所有单元格的内容信息。

其实就是简单应用一下python的第三方库docx。

四、实验步骤

1.

|  |
| --- |
| **merge.py** |
| content = []   **def** processContent():  **with** open(**'file1.txt'**) **as** file1:  text1 = [i.replace(**'\n'**, **''**) **for** i **in** file1.readlines()]  **with** open(**'file2.txt'**) **as** file2:  text2 = [i.replace(**'\n'**, **''**) **for** i **in** file2.readlines()]  *# 读取两个文件内容，忽略掉换行符* **for** i **in** range(2 \* max(len(text1), len(text2))): *# 用出队的方式读取字符串，可以免于调整下标* **if** i % 2 == 0: *# 行号为偶数，实为奇数行，进行出队* **if** len(text1) != 0:  content.append(text1.pop(0))  **else**: *# 行号为奇数，实为偶数行，进行出队* **if** len(text2) != 0:  content.append(text2.pop(0))   **return** content   **def** writeInFile(file): *# 写入文件* **with** open(**'result.txt'**, mode=**'w'**) **as** newFile:  newFile.write(**'\n'**.join(file)) |

2.

|  |
| --- |
| **createNewMerge.py** |
| content = []   **def** addLineIndex():  **with** open(**'merge.py'**, mode=**'r'**) **as** f:  text = f.read().split(**'\n'**)  length = max([len(i) **for** i **in** text]) + 4 *# 以每句的长度生成列表，再取最大值+4得设定的语句长度上限* **for** i **in** range(len(text)):  content.append(text[i] + **' '** \* (length - len(text[i])) + **'# '** + str(i + 1)) *# 为每句内容添加行号* **def** writeInFile(): *# 写入文件* **with** open(**'merge\_new.py'**, mode=**'w'**) **as** newFile:  newFile.write(**'\n'**.join(content)) |

3.

|  |
| --- |
| **readWord.py** |
| **from** docx **import** Document *# 导入库* path = **'1.docx'** *# 文件路径* content = Document(path) *# 读入文件* tables = content.tables *# 获取文件中的表格的集合* print(**"此文件中有"**, len(tables), **"个表格"**) *# 获取文件中的表格数量* **for** table **in** tables: *# 遍历每一个表格* **for** row **in** table.rows: *# 从表格第一行开始循环读取表格数据* **for** cell **in** row.cells: *# 遍历每一个单元格* print(cell.text, end=**' '**) *# 获取单元格的内容* print(**''**) *# 每一行的输出完了，增加一个换行* |

五、实验结果及分析

1.

图形用户界面, 应用程序

描述已自动生成图形用户界面, 应用程序

描述已自动生成图形用户界面, 应用程序, Word

描述已自动生成

2.

图形用户界面, 文本, 应用程序

描述已自动生成

3.

表格

描述已自动生成

图片包含 图示

描述已自动生成

六、实验总结及体会

Python本身就对文件内容的读取、修改的功能就已经十分完善且易于操作了，对于文件内容的操作本身不用设定太多的可选项，而且还不用接触如“文件流”之类的晦涩的概念。除此之外还有很多Python第三方库可供导入处理原本无法读取的二进制文件。由此更深刻感受到Python在功能实现方面强大的性能和极低的学习成本。

此外本题中利用队列的思想获取字符串，降低了代码复杂度提高了效率，也是彰显了数据结构的重要性。

七、教师评语