

考试说明

1. 考试时间：合计 3 小时。选择题部分 40 分钟交卷，允许提前交卷。
2. 考试过程中，不能连接未经指定网站或服务器。
3. 闭卷考试部分，不能查阅任何类型的参考资料。
4. 开卷考试部分，可以查阅纸质文档，不能查阅任何类型的电子文档。
5. 考试过程中，不得使用任何形式的电子存储设备，不可使用手机。
6. 违反上述 2-5 条者，视为考试作弊。

选择题答题方式（25 分，闭卷，严禁使用 python 编程环境进行尝试）

7. 打开浏览器，在地址栏中输入 <http://192.168.125.3>，点击相应链接进入登录页面。
8. 按要求输入两遍自己的学号。
9. 点击“登录”按钮即可进入答题页面。如考试尚未开始，系统会进入等待页面并倒计时。考试开始时间到，系统会自动进入答题页面。
10. 在页面左侧选择题号，页面右侧即会显示相应的题目。考生只需点击选择相应的选项。
11. 答题过程中如关闭浏览器或出现系统故障导致计算机重新启动，系统不会丢失之前已经完成的题目的答案。考生可以打开浏览器重新登录并继续考试。
12. 答题完成后，点击“交卷”按钮即可完成交卷。交卷后不能再次登录系统继续考试。
13. 考试结束时间到，系统会自动收卷。

编程题提交方式（75 分，开卷）

14. 提交前务必关闭 pycharm 编程环境。
15. 所有脚本程序内容必须仅包含在一个脚本程序文件（py 文件）中。
16. 在浏览器的地址栏中输入 <http://192.168.125.3>，点击相应链接进入提交页面。
17. 按要求输入两遍自己的学号。
18. 点击“选择文件”按钮，选择自己的脚本程序文件。点击“提交”按钮提交。
19. 如提交成功，系统会显示相关信息。如果提交不成功，请重复步骤 16-18。
20. 提交成功后，可点击“查看内容”按钮检查提交的内容。

按以下要求编写程序

题目说明

现在有一文本文件 `data.txt`，请将文本文件拷贝到 D 盘根目录，文本文件中包含多行字符串，每行字符串中包含若干用空格、逗号、句号分隔的单词，请编写程序完成如下功能：

1. 编写一个函数，从 `data.txt` 文件中读取所有单词，并保存到单词列表 `wordlst` 中。
2. 编写一个函数，找出 `wordlst` 中存在某个字母至少出现 `num` 次的单词，字母不区分大小写，将符合要求的单词保存到列表 `wordResultLst` 中，其中 `num` 由参数给出。
3. 编写一个函数，删除 `wordResultLst` 中重复单词多余份数，只保留一份，非重复单词保持不变。将结果仍然保存在列表 `wordResultLst` 中。
4. 编写一个函数，输出 `wordResultLst` 中所有单词，要求每个单词占 20 列，每行输出 `count` 个单词，其中 `count` 由参数给出。
5. 编写一个函数，将 `wordResultLst` 中的每一个单词转换成一个整数，保存到列表 `numLst` 中。转换规则：整数为单词的所有字母的 ASCII 值的累加和，例如：`sum` 对应的整数就是 `s`、`u`、`m` 三个字母的 ASCII 值之和。
6. 编写一个函数，对 `numLst` 中的所有整数按整数的数字累加和进行降序排序，例如整数：`923,456,134` 对应的整数数字累加和为 14，15，8，则排序结果 `456, 923, 134`。
7. 编写一个函数，输出排序后的 `numLst`，要求每个整数占 8 列，每行输出 `count` 个整数，其中 `count` 由参数给出。
8. 编写一个函数，统计 `numLst` 中每个数字出现的次数，将统计结果保存到字典 `resultDic` 中。
9. 编写一个函数，将统计结果 `resultDic` 中出现次数最多的数字及其出现次数输出到 D 盘根目录的文本文件 `result.txt` 中，输出格式：
数字(占 2 列,左对齐)：出现次数(占 3 列，右对齐)

测试程序如下(不允许修改测试程序，修改一处扣 2 分，直到扣满 10 分)：

```
if __name__ == "__main__":  
    # ----从 data.txt 文件中读取所有单词-----  
    wordlst = readWordsFromFile("d:\\data.txt")
```

```

print("文件中单词个数:", len(wordlst)) # 输出单词个数

# ----找出单词中, 存在某个字母重复 num 次的单词-----
wordResultLst = findMultiAlphaWords(wordlst, 2)
print("至少含有重复 2 次的字母的单词: ", len(wordResultLst))

# ----删除 wordResultLst 中重复单词的多余份数, 只保留一份-----
delMultiData(wordResultLst)
print("===删除重复单词的多余单词后的结果===")
printWordLst(wordResultLst, 4) #输出所有单词, 每行输出 4 个单词

# ----将 wordResultLst 中的所有单词转换为整数-----
numlst = getNumberOfWords(wordResultLst)

# ----对 numlst 中的所有整数进行根据数字累加和进行降序排序----
sortByDigitalSum(numlst)
print("===整数降序排序的结果===")
printNumLst(numlst, 5) #输出整数列表, 每行输出 5 个整数

# ----统计数字出现的次数-----
resultDic = staticDigitalTimes(numlst)
print("===出现次数最多的数字===")
printDicToFile("d:\\result.txt", resultDic)

```

一个简单示例结果(不是标准答案):

```

spyderWorkspace/testOfLesson')
文件中单词个数: 65
至少含有重复2次的字母的单词: 13
===删除重复单词的多余单词后的结果===
That          statements      program      packaged
procedure     function      parameters  that
serve         diffEr       Operation
===整数降序排序的结果===
 969  549  945 1096  592
 816  870 1076  760  433
 401
===出现次数最多的数字===
9 :  6

```

评分标准

(编程题满分为 75 分)

大项	子项	评分项	应得分	实得分
正确性 65 分	结果（65 分） （程序无法运行则此项不得分）	读取数据文件	10	
		查找字母指定重复次数的单词	5	
		删除重复单词的多余单词	10	
		将单词输出到屏幕	5	
		将单词转换为整数	5	
		对整数根据数字累加和降序排序	5	
		将整数输出到屏幕	5	
		统计数字出现的次数	10	
		将统计结果输出到文件	10	
		修改测试代码(一处扣 2 分，最多扣满 10 分)		
可读性 10 分	注释（6 分）	有详细且正确的注释	6	
		有注释，但不够详细	4	
		完全没有注释	0	
	变量命名（4 分）	变量命名有规则	4	
		变量命名有规则、但规则使用不一致	2	
		变量命名无规则	0	
总分（满分 75 分）				