**安徽工程大学**

Python语言程序设计**实验报告**

班级:人工智能191姓名:武新纪学号:3190707121成绩\_\_\_\_\_\_\_\_\_\_\_

日期:2020年11月16日指导教师:陈富春

**实验名称**实验四 Python函数操作

**【目的和要求】**

1. 掌握文件操作基本方法；
2. 掌握文件操作模式；
3. 掌握文本文件及二进制文件相关概念；
4. 掌握中英文词频统计方法；
5. 掌握图像处理方法。

**【实验准备】**

1. 列表、元组、字典、集合
2. 文件操作方法
3. Jieba库
4. PIL库的使用
5. **【实验内容】**下载Python科学计算发行版https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/
6. **Anaconda介绍**

Anaconda 是一个开源的用于科学计算的 Python 发行版，支持 Linux, Mac, Windows, 包含了众多流行的科学计算、数据分析的 Python 包和依赖项。因为包含了大量的科学包，Anaconda 的下载文件比较大（约 500-600MB）。如果只需要某些包，或者需要节省带宽或存储空间，也可以使用Miniconda这个较小的发行版（仅包含conda和 Python）。 Anaconda通过conda管理工具包、开发环境、Python版本，大大简化了科学计算与数据分析的工作流程。不仅可以方便地安装、更新、卸载工具包，而且安装时能自动安装相应的依赖包，同时还能使用不同的虚拟环境隔离不同要求的项目。 Anaconda 包含了720多个数据科学相关的开源包，在数据可视化、机器学习、深度学习等多方面都有涉及。不仅可以做数据分析，甚至可以用在大数据和人工智能领域。

anaconda安装完成后包括以下应用程序：

1）Anaconda Navigtor ：用于管理工具包和环境的图形用户界面，后续涉及的众多管理命令也可以在 Navigator 中手工实现。

2）Jupyter notebook ：Jupyter Notebook（此前被称为 IPython notebook）是一个基于web的交互式计算环境，支持运行 40 多种编程语言。Jupyter Notebook 便于创建和共享文学化程序文档，支持实时代码，数学方程，可视化和 markdown。

3）spyder ：一个使用Python语言、跨平台的、科学运算集成开发环境

4）打开spyder 进行如下练习：

**【练习】**

1.编写程序，创建一个文本文件，输入该内容（“中华人民共和国中国人民”）（存放D盘根目录下）打开该文本文件，读出其中内容，统计“中国”关键字出现的次数。

import jieba

a=input("请输入:\n")

with open("D:/text.txt",'w') as f:

    f.write(a)

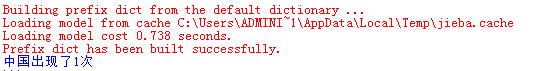
with open("D:/text.txt",'r') as f:

    str=f.read()

str1=jieba.lcut(str,*cut\_all*=True)

print("中国出现了{}次".format(str1.count("中国")))

**运行结果:**



2、将下列列表ls=[“清华大学”,“北京大学”,“中国人民大学”,“北京航天大学”，“北京师范大学”]写入文件，并对该文件进行如下操作：一个大学名存放一行，进行行遍历，并输出；在指定的位置产生一个相同副本的文件，完成文件复制功能。

**程序:**

ls = ["清华大学", "北京大学", "中国人民大学", "北京航天大学","北京师范大学"]

with open("test.txt",'w') as f:

    for i in ls:

        f.write(i+'\n')

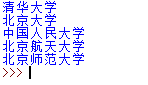
with open("test.txt",'r') as f1,open("test副本.txt",'w') as f2:

    for line in f1.readlines():

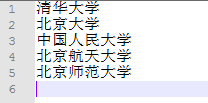
        f2.write(line)

        print(line.strip('\n'))

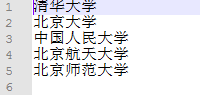
**运行结果:**



**test.txt文件内容**：



**test副本.txt文件内容:**



3.用WINDOWS记事本创建一个文本文件，其中每行包含一段英文。试读出文件的全部内容，并判断：

1）该文本文件共有多少行？

1. 包含最多和最少字符在哪几行？

**程序:**

i = 0

a=[]

with open("test2.txt",'r') as f:

    print("文件内容如下:")

    for line in f.readlines():

        a.append(len(line))

        print(line.strip('\n'))

        i+=1

dict1=dict(zip([j+1 for j in range(i)],a))

items = list(dict1.items())

items.sort(*key*=*lambda* *x*: *x*[1], *reverse*=True)

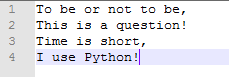
print("该文件有{}行".format(i))

print("包含字符数目最多的是{}行".format(items[0][0]))

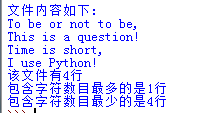
print("包含字符数目最少的是{}行".format(items[-1][0]))

**调试结果:**

test2.py文件内容:



**运行结果:**



4.向文本文件data.txt中写入整数，（文件中每行存放一个整数），将其读出后按升序排序后再写入文本文件data\_asc.txt中。

a=[10,34,23,21,54,34,23,23]

with open("data.txt",'w') as f:

    for i in a:

        f.write(str(i)+'\n')

with open("data.txt",'r') as f:

    b=list(map(int,f.readlines()))

    b.sort()

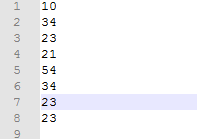
with open("data\_asc.txt",'w') as f:

    for i in b:

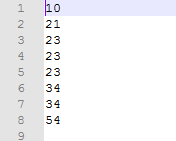
        f.write(str(i)+'\n')

**运行结果:**

**data.txt文件内容：**



**data\_asc.txt文件内容:**



5.打开hamlet.txt文本，统计文章中前20位实词出现的次数。

**程序:**

excludes = {"the", "and", "of", "you", "a", "i", "my", "in"}

*def* getText():

    txt = open("hamlet.txt", "r").read()

    txt = txt.lower()

    for ch in '!"#$%&()\*+,-./:;<=>?@[\\]^\_‘{|}~':

        txt = txt.replace(ch, " ")  # 将文本中特殊字符替换为空格

    return txt

hamletTxt = getText()

words = hamletTxt.split()

counts = {}

for word in words:

    counts[word] = counts.get(word, 0) + 1

for word in excludes:

    del(counts[word])

items = list(counts.items())

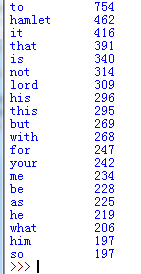
items.sort(*key*=*lambda* *x*: *x*[1], *reverse*=True)

for i in range(20):

    word, count = items[i]

    print("{0*:<10*}{1*:>5*}".format(word, count))

**运行结果:**



**【思考题】**

1、文本文件和二进制的不同点？

**答:文本文件一般由特定编码的字符组成，如UTF-8编码，内容容易统一展示和阅读。大部分文本文件可以通过文本编辑软件或者文字处理软件创建、修改和阅读。**

**二进制文件直接由比特0和比特1组成，没有统一的字符拜尼马，文件内部数据的组织形式和文件用途有关。二进制是信息按照非字符但特定格式形成的文件。二进制文件和文本文件最主要的区别在于是否有统一的字符编码。**

* 1. 打开三国演义.txt文本，统计文章中主要人物出现的次数。

import jieba

excudes={"将军","却说","荆州","二人","不可","不能","如此"}

txt=open("三国演义.txt",'r',*encoding*='utf-8').read()

words=jieba.lcut(txt)

counts={}

for word in words:

    if len(word)==1:

        continue

    elif word=="诸葛亮" or word=="孔明曰":

        rword="孔明"

    elif word == "关公" or word == "云长":

        rword = "关羽"

    elif word == "玄德" or word == "玄德曰":

        rword = "刘备"

    elif word == "孟德" or word == "丞相":

        rword = "曹操"

    else:

        rword=word

    counts[rword]=counts.get(rword,0)+1

for word in excudes:

    del (counts[word])

items=list(counts.items())

items.sort(*key*=*lambda* *x*:*x*[-1],*reverse*=True)

for i in range(5):

    word,count=items[i]

    print("{0*:<10*}{1*:>5*}".format(word,count))

运行结果:

