第九周Python实验报告

**学号**：117060400110 **姓名**：蒙柳双

**班级**：17应用统计学1班 **指导老师**：林卫中老师

**实验目的：**

1. 学会运用字典或者jieba库进行文本字符分析
2. 会安装jieba库
3. 会利用jieba库与字典类型（无顺序）、再利用sort()方法与lamnda函数（对单词出现的次数进行排序）配合、采用函数对获取和整理文本进行封装然后进行人物出场次数统计。
4. 会在前面的基础上加入排除词库excludes和对同一人物不同名字的处理
5. 对于ex6-5.生日悖论，这是一个综合的问题，考验综合能力。

**实验要求：**

1. 上课不玩手机
2. 要认真编码并且思考
3. 实在不会的就向老师求助

**实验内容步骤：**

**程序练习6-4**

**文本字符分析**

编写代码接收字符串，按字符出现频率的降序打印字母。分别尝试录入一写中英文文章片段，比较不同语言之间字符频率的差别。

**核心代码：**

from operator import itemgetter

def analyseText(text):

D={}

tlist = list(text)

for e in tlist:

D[e] = D.get(e,0)+1

return D

text = input("请输入一段文本：")

D = analyseText(text)

D=sorted(D.items(),key=itemgetter(1),reverse=True)

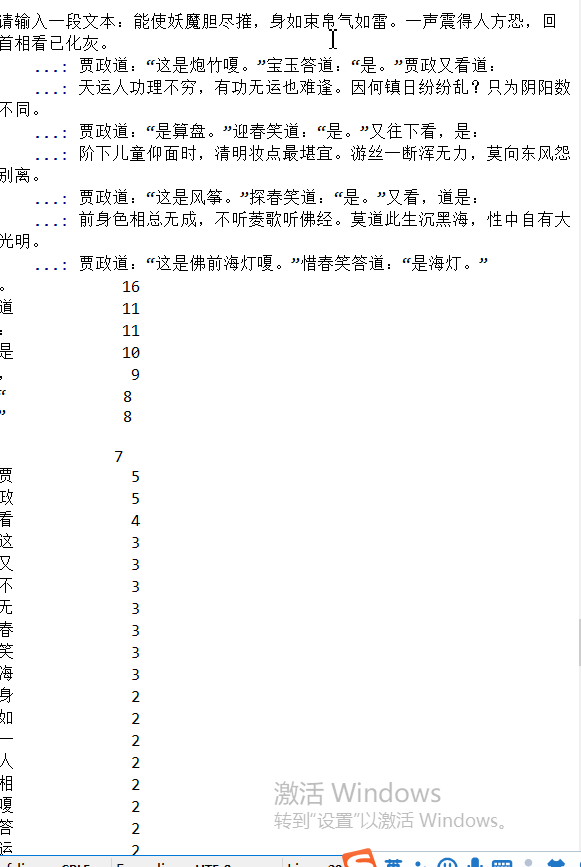
for d in D:

ch,count = d[0],d[1]

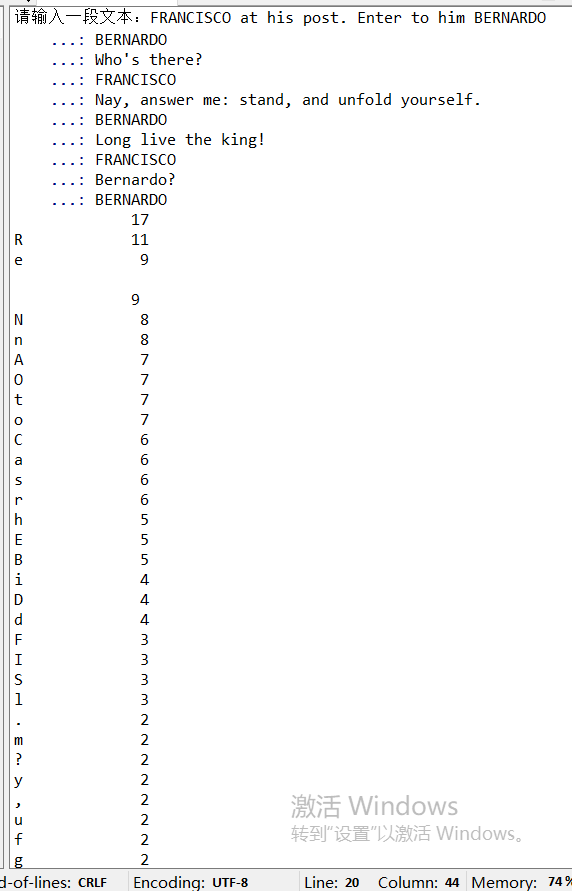
print("{0:<10}{1:>5}".format(ch,count))

**实验结果：**

**中文：**



**英文：**



**程序练习6-6**

《红楼梦》人物统计。编写程序统计《红楼梦》中前20位出场最多的人物。

**核心代码**：

import jieba

excludes = {'什么','一个','我们','你们','如今','说道','知道','起来','这里','出来','众人','那里',

'自己','姑娘','太太','一面','只见','两个','没有','怎么','不是','这个','听见','这样',

'进来','咱们','就是','不知','东西','告诉','回来','只是','大家','老爷','只得','丫头',

'这些','他们','不敢','出去','所以','不过','不好','姐姐','的话','一时','过来','不能',

'心里','她们','如此','银子','今日','二人','答应','这么','几个','还有','只管','说话',

'那边','一回','这话','外头','自然','哪里','打发','袭人','今儿','罢了','鸳鸯','那些',

'屋里','问道','小丫头','如何','听说','人家','看见','媳妇','不用','妹妹','不得','一声',

'姊妹','家里','原来','一句','这会子','到底','别人','过去'}

txt = open("红楼梦.txt","r",encoding='GB18030').read()

words = jieba.lcut(txt)

counts = {}

for word in words:

if len(word) == 1:

continue

elif word == "老太太" or word == "奶奶":

rword = "贾母"

elif word == "凤姐儿" or word == "凤姐":

rword = "王熙凤"

else:

rword = word

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

for word in excludes:

del(counts[word])

items = list(counts.items())

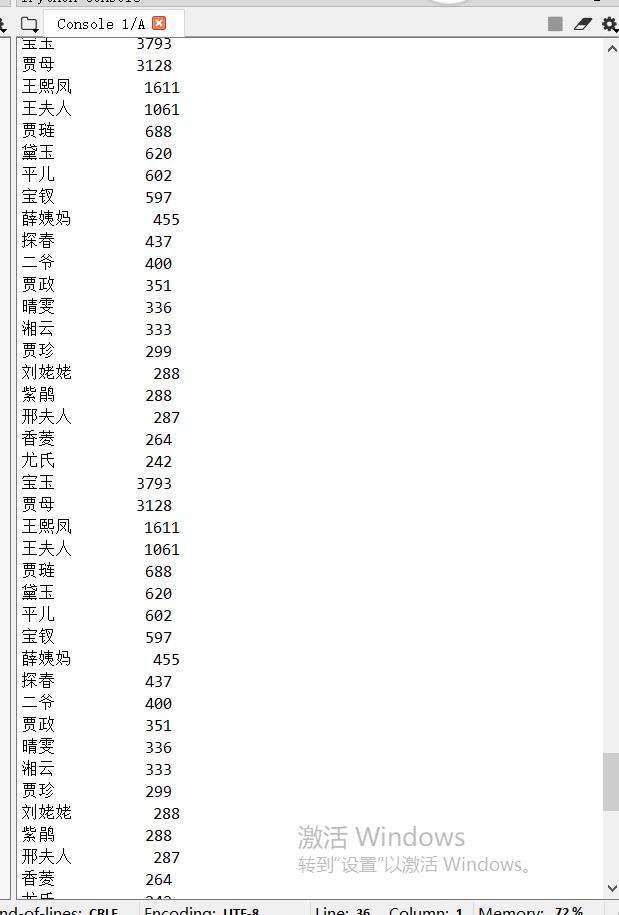
items.sort(key=lambda x:x[1],reverse=True)

for i in range(20):

word,count = items[i]

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

**实验结果：**



**《三国人物》人物统计**

**核心代码：**

import jieba

excludes = {"将军","却说","二人","不可","荆州","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 excludes:

del(counts[word])

items = list(counts.items())

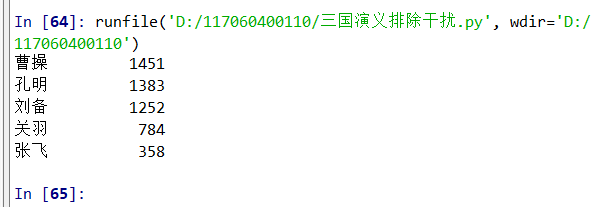
items.sort(key=lambda x:x[1],reverse=True)

for i in range(5):

word,count = items[i]

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

**实验结果：**



**程序练习6-5**

**生日悖论分析。**

生日悖论指如果一个房间里有23人或以上，那么至少有两个人生日相同的概率大于50%。编写程序，输出在不同随机样本数量下，23个人中至少两个人生日相同的概率。

**核心代码：**

from datetime import datetime

from random import\*

def generatesSamles1(n:int):

birthdays = []

days = [31,29,31,30,31,30,31,31,30,31,30,31]

for i in range(n):

month = randint(1,12)

day = randint(1,days[month-1])

someday = (month,day)

birthdays.append(someday)

return birthdays

def genrateSamples2(n,int):

birthdays = []

days =[31,28,31,30,31,30,31,31,30,31,30,31]

year = randint(1950,2000)

for i in range(n):

month = randint(1,12)

if (year%400==0) and (year%4==0 and year%100 !=0):

day[1] = 29

else:

days[1] = 28

day = randint(1,days[montn-1])

someday = (month,day)

birthdays.append(someday)

return birthdays

def calSameBirthdayProb(birthdays:list,n:list):

num =0

for i in range(n):

people = sample(birthdays,23)

pset = set(people)

if len(pset) != len(people):

num = num+1

return num/n

def main():

while True:

n = int(input("请输入一个整数："))

if n <23 :

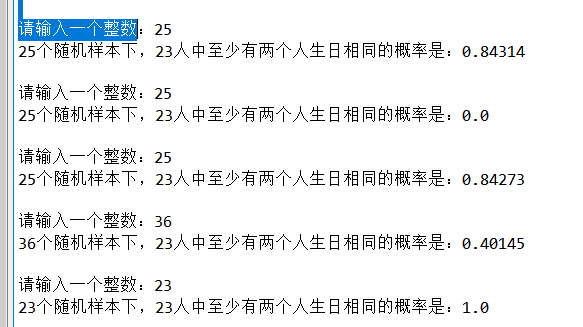
break

birthdays = generatesSamles1(n)

print("{}个随机样本下，23人中至少有两个人生日相同的概率是：{}".format(n,calSameBirthdayProb(birthdays,100000)))

main()

**实验结果：**



**实验总结：**

1. 经过老师的再次分析，对于文本字符分析这一题又再一次有了深入的认识，从而可以理解题目，比前一次的实验了解更加深入了
2. 确实，按理说若课本没有《三国演义》人物出场统计这一例题跟一些排除词库excludes跟同一人物不同名字的处理的例题，ex-6-6题确实是一个难题，感觉自己也不会写得出来
3. 对于生日悖论这题，自己是有思路该怎么去做，但是不会用代码写出来，感觉有很多东西自己不会用代码表达出来。