**数据结构实验报告9**

**学号：** 117060400106 **姓名**： 尹华剑 **班级：**应用统计一班  **指导老师：** 林卫中

**实验名称**：  **第六章程序练习题6.4-6.6**

**实验要求：**

**1、掌握组合数据类型中元组、数组、列表、集合和字典等类型及基本操作**

**2、掌握jieba词库对中文文档进行并进一步统计文档词频。**

**实验题目一：**

1. **文本字符分析。**
2. **生日悖论分析。**
3. **《红楼梦》人物统计。**

**算法实现：1.**

**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)**

**items = list(D.items())**

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

**for item in items:**

**ch, count = item**

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

**2.import random**

**def generateSamples(n:int): #生成n个人的生日，并放到一个列表中**

**birthdays = []**

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

**for i in range(n):**

**month = random.randint(1, 12)**

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

**someday = (month, day) #生成月，日**

**birthdays.append(someday)**

**return birthdays**

**def sameBirthday(birthdays:list, n:int):**

**num = 0**

**for i in range(n):**

**people = random.sample(birthdays, 23)**

**pset = set(people) #将列表变为集合，计算相同生日的人的个数**

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

**num = num + 1**

**return num/n**

**def main():**

**while True:**

**n = int(input("please enter a number: "))**

**if n < 23:**

**break**

**birthdays = generateSamples(n)**

**print("{}个随机样本数量下，23个人中至少有两人生日相同的概率是：{}".format(n, sameBirthday(birthdays, 100000)))**

**break**

**main()**

**3.import jieba**

**excludes = {'什么','一个','我们','你们','如今','说道','老太太','知道','姑娘','起来',**

**'这里','出来','众人','那里','奶奶','自己','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'**

**}**

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

**words = jieba.lcut\_for\_search(txt)**

**counts = {}**

**for word in words:**

**if len(word) == 1:**

**continue**

**elif word == '凤姐' or word == '熙凤':**

**rword = '凤姐'**

**elif word == '元春' or word == '贵妃':**

**rword = '元春'**

**else:**

**rword = word**

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

**for word in excludes:**

**del(counts[word])**

**items = list(counts.items())**

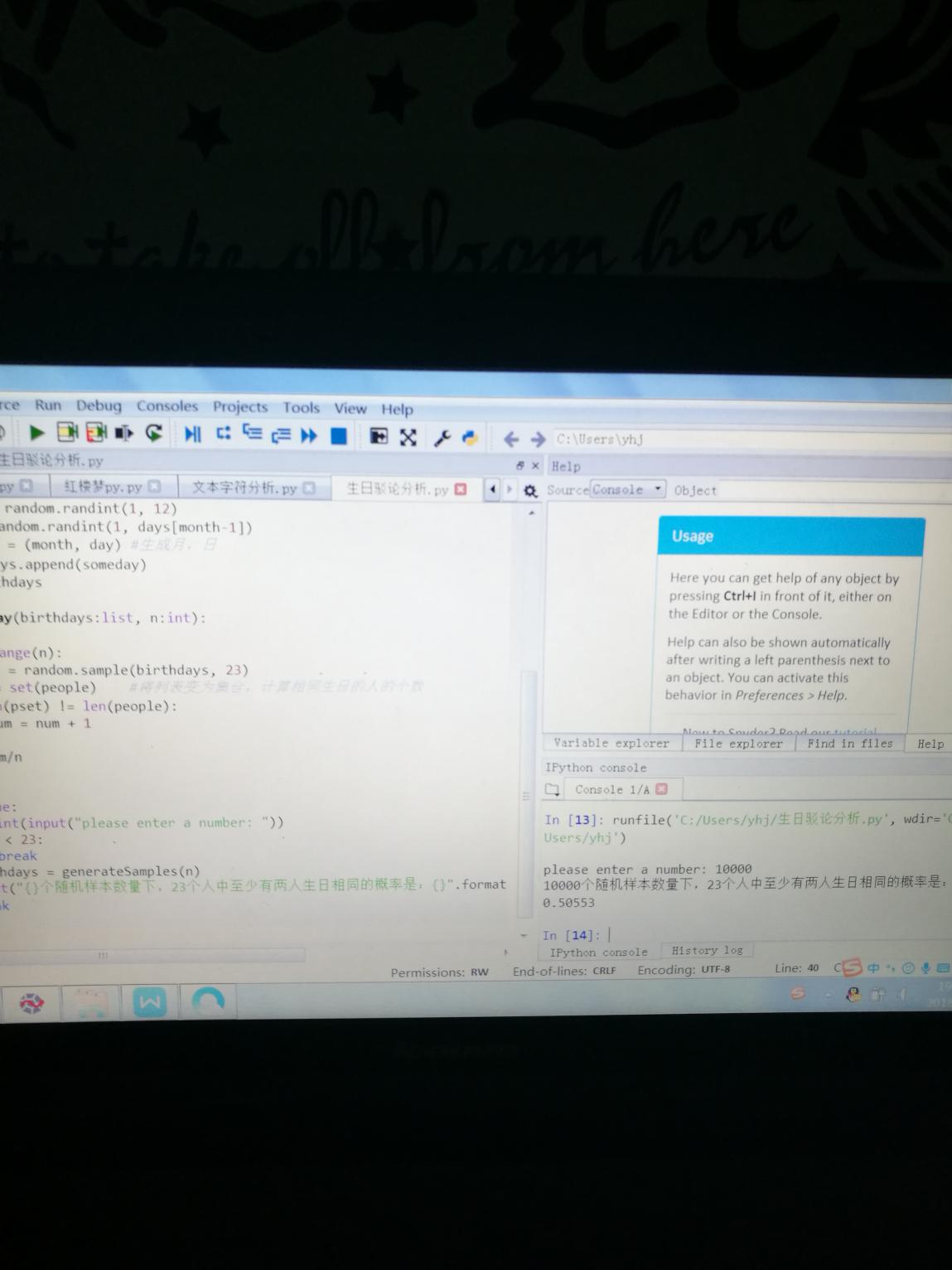
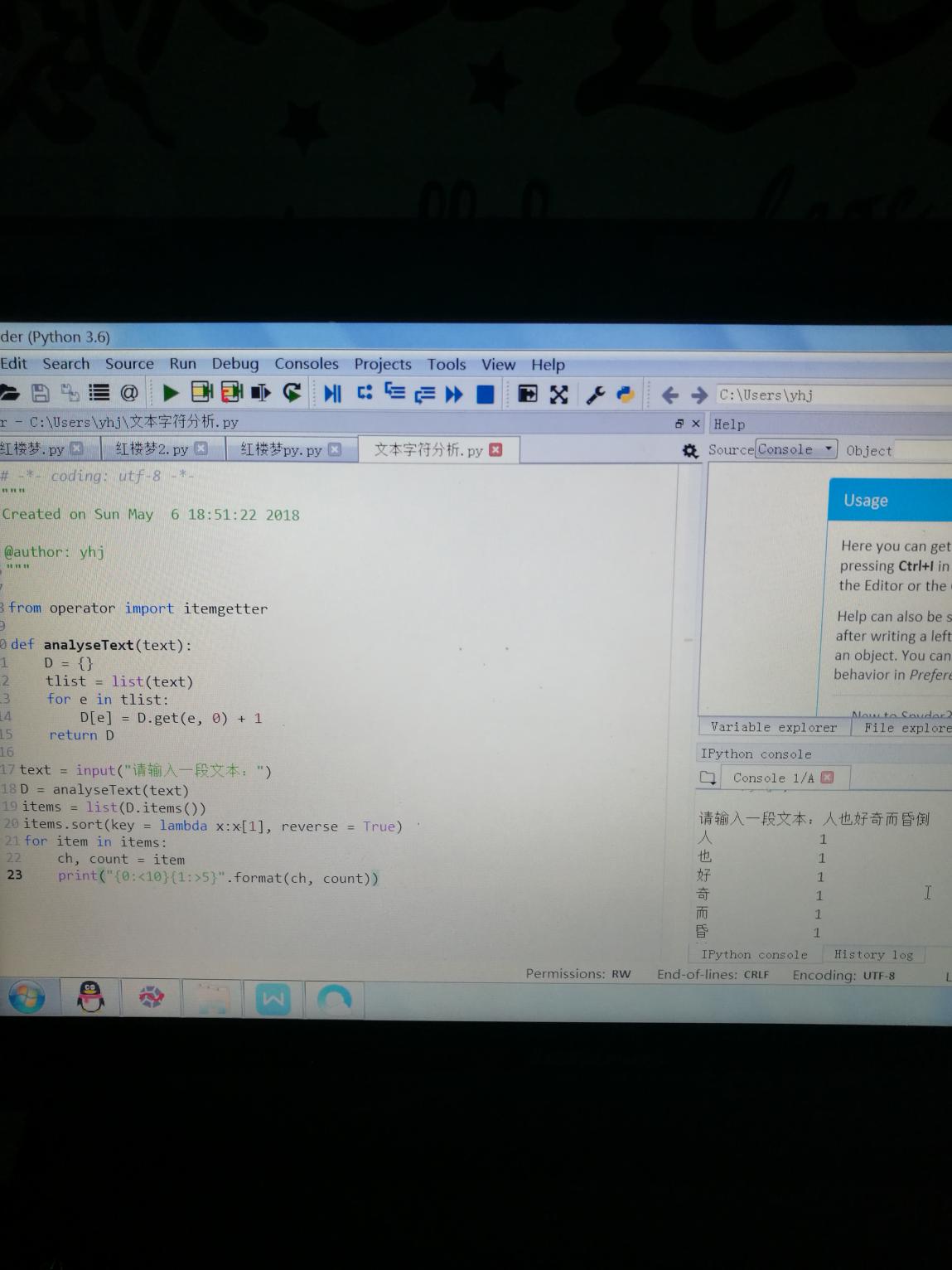
**items.sort(key=lambda x: x[1], reverse = True)**

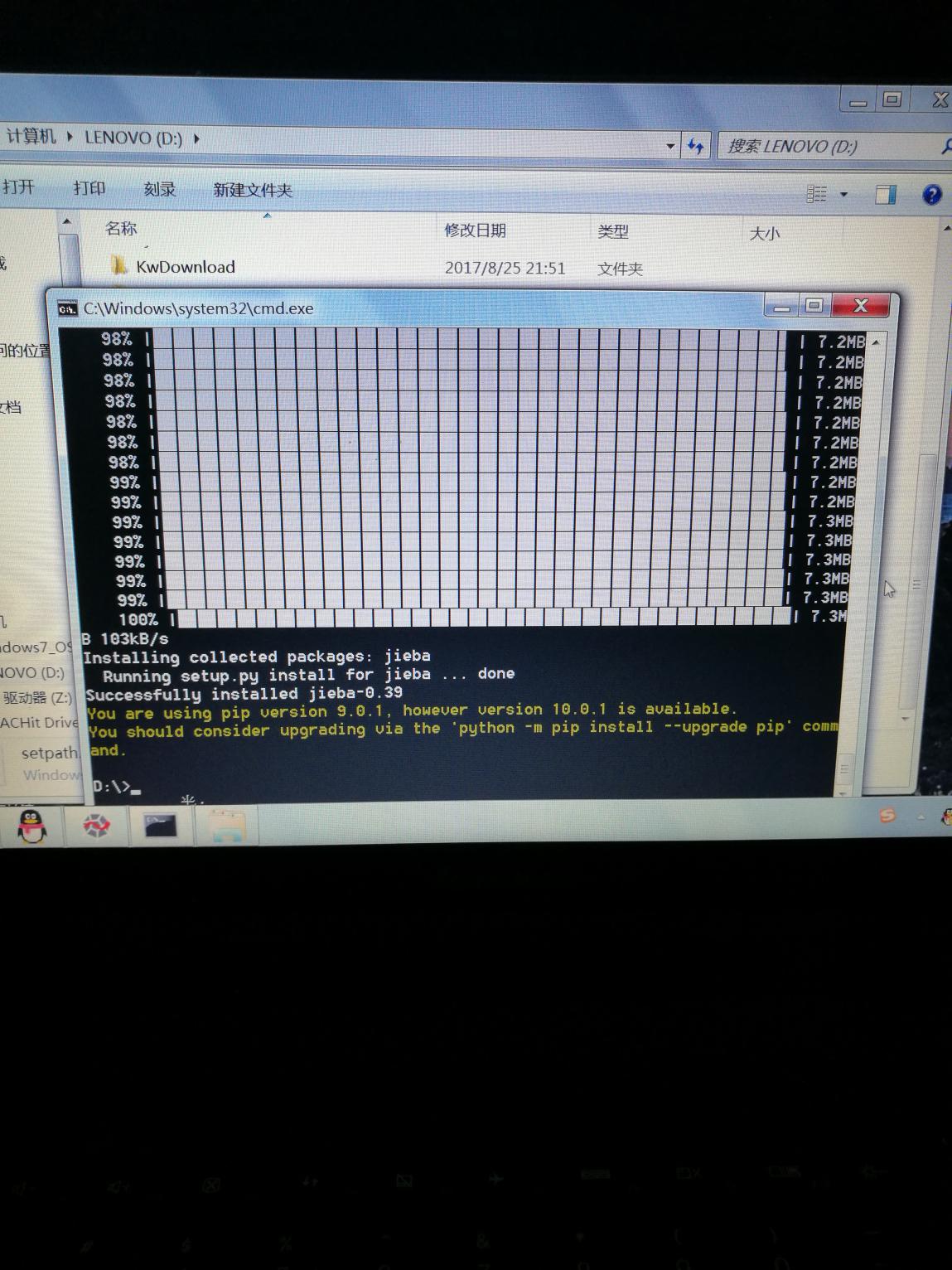
**for i in range(50):**

**word, count = items[i]**

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

**实验结果：**

**：**

****

**反思：**

**本次自我安装jieba 库出现了一点错误，感觉像是安装好了，但还是用不了，导致6.6题的《红楼梦》人物分析做不出来！**