# 计算机程序设计实验报告9

**学号： 117060400104姓名 熊骏班级：应统1班指导老师： 林卫中**

**实验要求：如何了解使用jieba词库对中文文档进行分词并进一步统计文档词频**

**实验心得：运用列表管理采集的信息，构建数据结构。运用组合数据类型进行文本词频统计**

实验题目：**文本字符分析 红楼梦人物统计 生日悖论分析**

## 实验代码：

**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 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','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','这话','一回','外头','自然','打发','哪里','今儿','罢了','那些','屋里'}**

**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 == "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(20):**

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

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

**3、**

**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()**

## 实验结果截图:





