文本处理 实验报告

宁晨然 17307130178

一、问题一

说明以下的正则表达式匹配的字符串类：[a-zA-Z]+；[A-Z][a-z]*；p[aeiou]{,2}t；\d+(\.\d+)?；([^aeiou][aeiou][^aeiou])*；\w+|[^\w\s]+。

ANS:

①[a-zA-Z]+

匹配 1 个或多个的英文字符（包含小写与大写）序列

②[A-Z][a-z]*

匹配首字母大写后面字符小写（0 个或多个）的英文字符序列

③p[aeiou]{,2}t

匹配 p-t 字符序列，-用 0 到 2 个元音字符填充的英文小写字符序列

④\d+(\.\d+)?

\d+表示 1 个或多个数字，(\.\d+)?表示出现 0 或 1 次小数（至少一位小数）。

匹配整数或小数（包括 0 开头的数字）

⑤([^aeiou][aeiou][^aeiou])*

匹配 3 位循环字符串序列（0 次或多次），循环节的首尾字符非元音、中间字符元音。

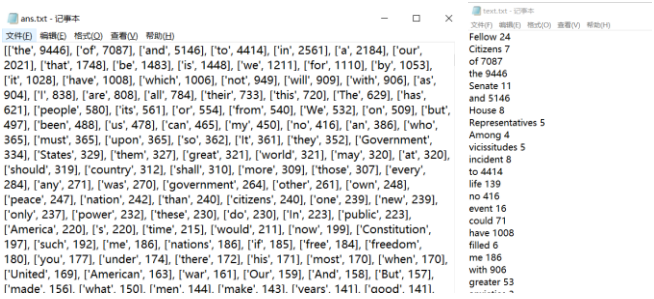
⑥\w+|[^\w\s]+

匹配 1 个或多个 字母数字下划线 或 除字母数字下划线空白符的字符序列。所有字符必须同时属于一个集合，例如'a*a'不能匹配，可以匹配'12a'和'*……'。

二、问题二

创建一个文件，包含词汇和（任意指定）频率，其中每行包含一个词，一个空格和一个正整数，如：fuzzy 53。使用 open(filename).readlines()将文件读入 Python 链表。接下来，使用 split()将每一行分成两个字段，并使用 int()将其中的数字转换为一个整数。结果要求是链表形式：[['fuzzy', 53], ...]。

ANS：



先使用 freqdist 构造一个词频 list 如左图，然后使用 split 来改变每行的 str，最后格式化输出。代码如下，关键部分由红色标出。

```
from nltk.book import text4
import re, pprint,nltk

word = [w for w in text4 if re.search('^[a-zA-Z]+$',w)]
fdist = nltk.FreqDist(word)
```

```
data = open("text.txt","r+")
for sample in fdist:
    string = sample + ' ' + str(fdist[sample])
    print(string,file=data)

def change(s):
    s = s.split()
    print(s)
    s[1] = int(s[1])
    return s

f = open('text.txt').readlines()
f = [change(s) for s in f]
f = sorted(f,key = lambda x:x[1],reverse=True)
data2 = open('ans.txt','w')
print(f,file=data2)
```

三、问题三

定义一个变量 silly 包含字符串：'newly formed bland ideas are inexpressible in an infuriating way'。编写代码执行以下任务：分割 silly 为一个字符串链表，每一个词一个字符串，使用 Python 的 split()操作，并保存到叫做 bland 的变量中；提取 silly 中每个词的第二个字母，将它们连接成一个字符串，得到'eoldrnnnna'；使用 join()将 bland 中的词组合成一个单独的字符串。确保结果字符串中的词以空格隔开。

ANS：

```
silly = 'newly formed bland ideas are inexpressible in an infuriating way'
bland = silly.split()
t = ''
for s in bland:
    t = t + str(s[1])
a = ' '.join(bland)
print(t)
print(a)
```

输出结果

```
eoldrnnnna
newly formed bland ideas are inexpressible in an infuriating way
```