信息处理技术 作业 4

From: 梁鑫宇 3160104494

题目：运用词典法进行中文自动分词

思路分析：

由于词典法需要对分词词表进行处理，考虑使用 json 文件便于排序，查找，同时降低模块之间的耦合性，降低代码重复，提高程序扩展性

源代码：

ConvertToJson.py

```
import json

words = []

# 打开 txt 格式的分词词表为 file_obj
with open("wordslist.txt") as file_obj:
    for word in file_obj: # 逐行压入 word 列表中
        words.append(word.rstrip()) # 利用 retrip 函数除去行尾可能有的空格

# 为便于二分查找，利用 sort 函数对词表进行排序
words.sort()

# 利用 dump 函数将排序后的 word 词表编码成 JSON 字符串存入新建的 words_list.json 文件中
with open("words_list.json",'w') as store:
    json.dump(words,store)
```

load.py

```
import json

# 此函数用于求出词表中最长的词的长度
def MaxSize(json_name):
    MaxSize = 0

    # 根据函数接收的参数文件名打开 json 文件为 file_obj
    with open(json_name) as file_obj:
        words = json.load(file_obj) # 用 words 列表接收 load 函数解码的 JSON 数据

    # 遍历 words 中的元素，返回最长的元素的长度
    for word in words:
```

```
        if len(word) > MaxSize:
            MaxSize = len(word)
    return MaxSize

# 此函数用于读取 json 文件，返回存有文件内容的列表
def GetList(json_name):
    with open(json_name) as file_obj:
        words = json.load(file_obj)
    return words
```

search.py

```
# 二分查找
def check(word,words,left,right):
    while left<=right:
        mid = int((left+right)/2)
        if word == words[mid]:
            return True
        elif word < words[mid]:
            right = mid - 1
        else:
            left = mid + 1
    if left > right:
        return False
    else:
        return True
```

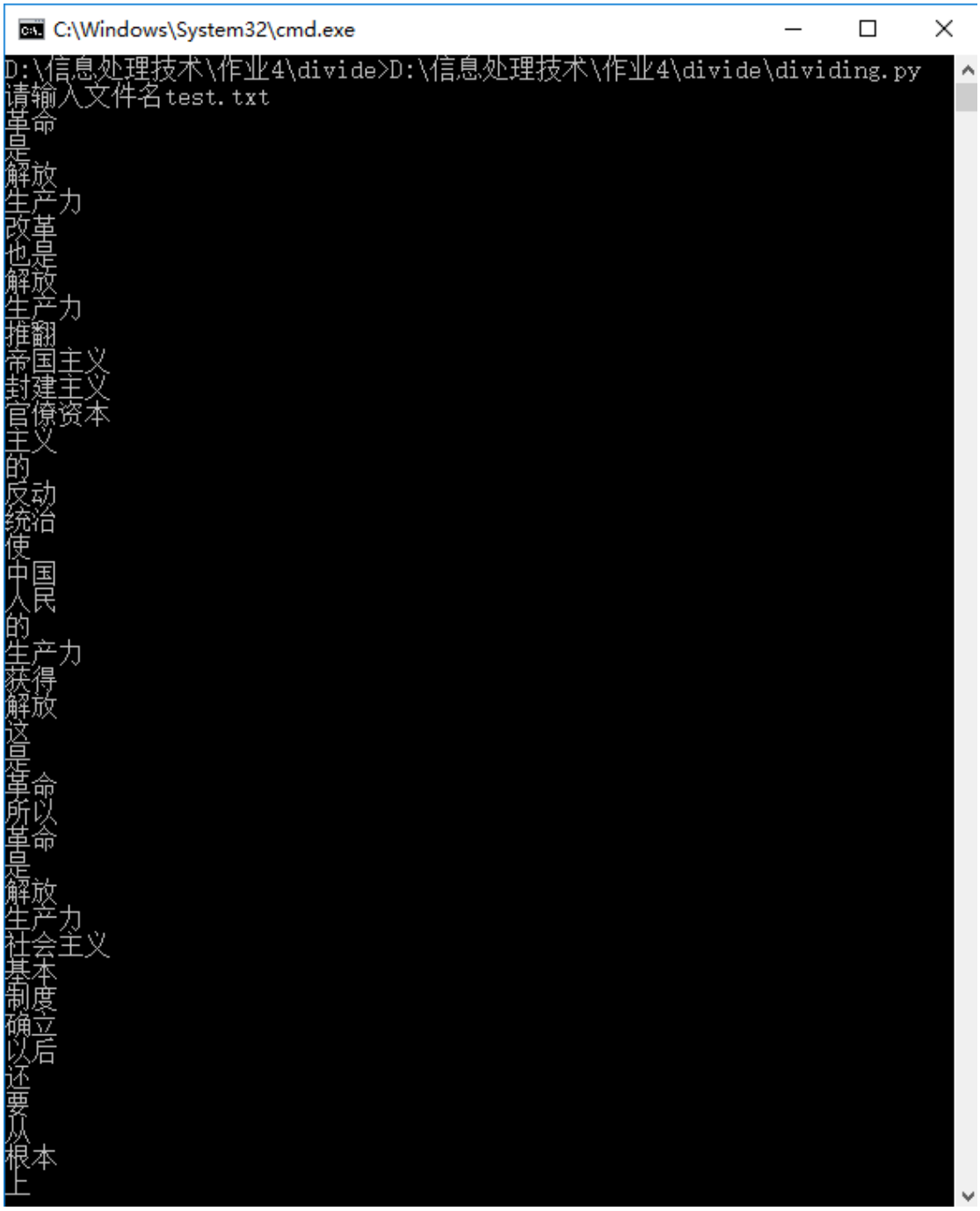
dividing.py

```
import load
from search import check

def GetWord(UserStr,MaxSize,words):
    list_size = len(words) # 获得分词列表的长度

    temp_word = UserStr[0:MaxSize] #从左边截取 MaxSize 长度的字符串
    while len(temp_word) > 1: #循环查找直到截取的字符串长度 = 1 为止
        if check(temp_word,words,0,list_size-1): # 调用 check 函数二分查找 temp_word 是否在词表中
            return temp_word
        else: # 若不在，则缩短截取的字符串的长度
            temp_word = temp_word[0:len(temp_word)-1]

    return temp_word
```

```
# 此函数用于展示分词结果
def show(word):
    # 接收的字符串长度 >1 时说明分词词表中包含此词，直接输出
    if len(word) > 1:
        print(word)
    # 接受的字符串长度 =1 时，此字符可能是汉字单字也可能是标点符号或英文等其他字符，故筛选后再输出
    elif word >= u'\u4e00' and word <= u'\u9fa5':
        print(word)

# 运行分词程序前先运行 ConvertToJson.py 将分词词表转换为 json 文件
json_name = "words_list.json"
MaxSize = load.MaxSize(json_name) # 得到分词词表最长词的长度
words = load.GetList(json_name) # 接收分词词表为列表 words
divided_file = input("请输入文件名")

# 利用 try-except 代码块处理用户输入无效文件名的情况
try:
    with open(divided_file) as file_obj:
        for UserStr in file_obj: # 按行遍历文件
            while(len(UserStr)): # 循环分词直至一行长度被截取至 0
                if len(UserStr)>=MaxSize:
                    # 调用 GetWord 函数从一行的左侧开始获得分词
                    word = GetWord(UserStr,MaxSize,words)
                    # 展示分词结果
                    show(word)
                    # 将获得的分词从这一行中截去
                    UserStr = UserStr[len(word):]

                # 当一行本身或经历遍历被截取至长度小于词表中的最长词时，直接将行的长度作为分词长度最大值传给 GetWord 函数
                else:
                    word = GetWord(UserStr,len(UserStr),words)
                    show(word)
                    UserStr = UserStr[len(word):]

except FileNotFoundError:
    print("File Not Found") # 用户输入无效文件名时发出异常提示
```

测试样例：

测试文本：邓小平南方讲话节选


C:\Windows\System32\cmd.exe
D:\信息处理技术\作业4\divide>D:\信息处理技术\作业4\divide\dividing.py
请输入文件名test.txt
革命
是
解放
生产力
改革
也是
解放
生产力
推翻
帝国主义
封建主义
官僚资本
主义
的
反动
统治
使
中国
人民
的
生产力
获得
解放
这
是革命
所以
革命
是
解放
生产力
社会主义
基本
制度
确立
以后
还
要
从
根本
上