JAVA 编程进阶上机报告

学　院　__智能与计算__

专　业　__软件工程__

班　级　__一班__

学　号　__3018216063__

姓　名　__陈宇涛__

一、实验要求

编写程序，统计了不起的盖茨比中各个单词出现的频次（结果按照单词的频次倒序排列），并为输入文件，创建一个 output.txt

二、源代码

```java
package lab2;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.Writer;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.HashMap;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.Map.Entry;
import java.util.TreeMap;

public class readfile {
//判断是否为要求的编码格式
    public static boolean code(File file) throws IOException {
        try {
         InputStream ios=new FileInputStream(file);
         byte[] b=new byte[3];
           ios.read(b);
           ios.close();
           if(b[0]==-17&&b[1]==-69&&b[2]==-65) {
               return true;
           }

    }catch (Exception e) {
        System.out.println("读取文件错误");
        e.printStackTrace();
```

```java
        }
        return false;
    }
    //读入文件并将其按字符串内容、数量用map记录
    public static Map<String,Integer> readFile(String filePath){
        Map<String,Integer> map = new TreeMap<String,Integer>();
        try {
            File file=new File(filePath);
            if(!file.canRead()) {
             System.out.println("文件无读权限");
            }else if(code(file)){
             System.out.println("文件编码错误");
            }else if(file.exists()){

                BufferedReader Bufferedreader = new BufferedReader(new
InputStreamReader(new FileInputStream(filePath)));
                String words = "";
                while((words = Bufferedreader.readLine())!=null){
                    String[] wordline = words.split("\\s+");
                    for(String word:wordline){
                        if(map.containsKey(word)){
                            int n =1 + map.get(word);
                         map.remove(word);
                         map.put(word, (Integer)n);
                        }else{
                            map.put(word, 1);
                        }
                    }
                }

                Bufferedreader.close();                }else{
                System.out.println("没有指定的文件");
            }
        } catch (Exception e) {
            System.out.println("无法读取文件");
            e.printStackTrace();
        }
        return map;
    }
    //借用list排序
    public static List<Map.Entry<String,Integer>>
    sort(Map<String,Integer> map) {
        List<Map.Entry<String,Integer>> list = new
    ArrayList<Map.Entry<String,Integer>>(map.entrySet());
```

```java
        Collections.sort(list,new
    Comparator<Map.Entry<String,Integer>>() {
            public int compare(Entry<String, Integer> o1,
                    Entry<String, Integer> o2) {
                return o2.getValue().compareTo(o1.getValue());
            }

        });
        return list;
    }
//将内容写入output文件中
    public static void write(List<Map.Entry<String,Integer>> list)
    throws IOException {
        try {
        File file = new File("D:" + File.separator +  "output.txt");
        Writer writer = new FileWriter(file);
        Iterator<Entry<String, Integer>> iterator = list.iterator();
        while(iterator.hasNext()){
            Entry<String, Integer> entry = iterator.next();
            String str = entry.getKey()+"  "+entry.getValue()+"\n";
            char[] c = str.toCharArray();
            writer.write(c);
        }
        writer.flush();
            writer.close();    }catch (Exception e) {
            System.out.println("输出文件出错");
            e.printStackTrace();
        }



    }
    public static void main(String argv[]) throws IOException{
    Map<String,Integer> map = new TreeMap<String,Integer>();
    map = readFile("D:/Download/了不起的盖茨比英文.txt");
    List<Map.Entry<String,Integer>> list = sort(map);
        write(list);
    }
}
```
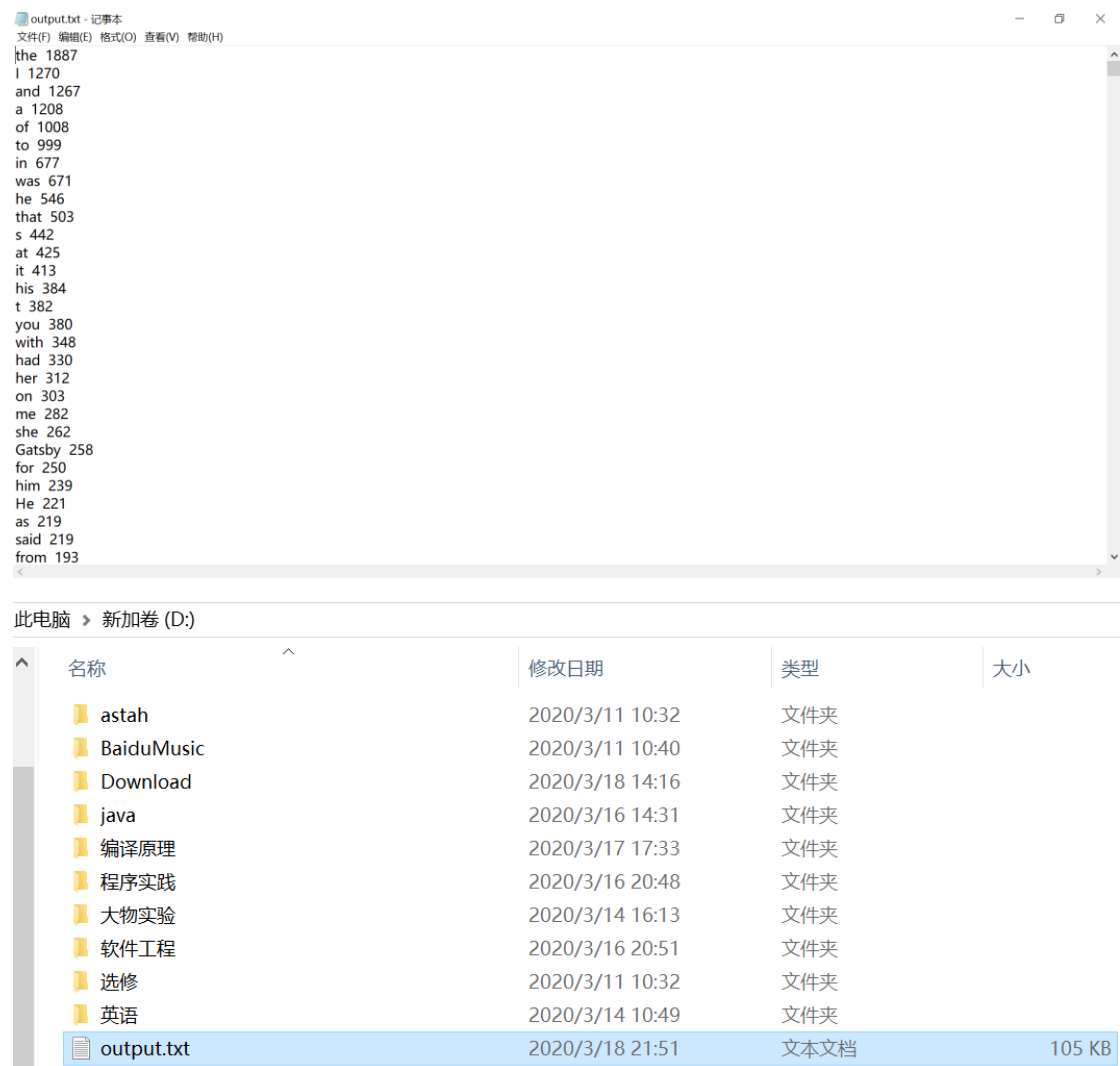
三、实验结果

四、设计

readfile：读入文件，若文件不存在、无读权限、文件编码错误则抛出异常，若正常则一个个读入单词并通过 map<String,Integer>记录次数

code：通过 InputStream 读取文件并通过前 3 个字节判断编码类型

sort：转成 list 通过 Collections.sort 函数按 Value 值倒着排序

write： 遍历 list 并转成 String 通过 writer.write 写入对应文件中