西南民族大学

**实验报告**

2019 ------2020 学年第 2 学期

课程名称：软件工程课程设计

学 院：计算机科学与技术

专业：计算机科学与技术

年级：2017级 班级：1702班

学号：201731102132 姓名：关佳平

同组人：201730206181 岳欣

|  |
| --- |
| 西南民族大学学生实验报告  教学单位：计算机科学与技术 实验室名称：BS-226 实验时间： 2020年3月20日  专业：计算机科学与技术 班级：1702班 |
| 实验项目名称：第5次实验 实验成绩： 教师签名：周绪川 |
| **一、实验目的与背景** 目的：把个人项目的三个的功能封装为独立模块，并设计单元测试 **1.单独运行yq.exe或者yq.java，给出全省输入文件的输出；**  **2.运行yq [指定输入文件] [指定输出文件]；（按要求排序）**  **3.运行yq [指定输入文件] [指定省份]；（按要求排序）**  **二、材料与方法**  材料：pc机，jdk1.8环境  方法： BufferedReader; FileOutputStream; FileReader;  **三、实验主要过程与结果**  **1、读取yq\_in.txt文件**  FileReader file = new FileReader(args[0]);  BufferedReader br = new BufferedReader(file);  2、对读取到的数据进行分割处理  String line = br.readLine();  String[] word = line.split("\t");  String data = word[1] + "\t" + word[2] +"\n";  3、输出为yq\_out.txt文件  fos.write(data.getBytes());  4、定义一个core类，对三种方法进行封装  5、全部代码实现   |  | | --- | | import java.io.BufferedReader;  import java.io.FileOutputStream;  import java.io.FileReader;  import java.util.Arrays;  //封装类  class core{    //**单独运行yq.exe或者yq.java，给出全省输入文件的输出；**  public void getAllProvince() throws Exception {  FileReader file = new FileReader("E:\\testData\\yq\_in.txt");  BufferedReader br = new BufferedReader(file);  FileOutputStream fos = new FileOutputStream("E:\\testData\\yq\_out.txt");    String line = br.readLine();  String shen = "";  int count = 1;    while(line != null) {    String[] word = line.split("\t");  String data = "";    if(!shen.equals(word[0]) && !word[2].equals("0")) {  if(count == 1) {  data = word[0] + "\n" + word[1] + "\t" + word[2] + "\n";  fos.write(data.getBytes());  count++;  }else {  data = "\n" +word[0] + "\n" + word[1] + "\t" + word[2] + "\n";  fos.write(data.getBytes());  }  shen = word[0];  }else if(shen.equals(word[0]) && !word[2].equals("0")) {  data = word[1] + "\t" + word[2] + "\n";  fos.write(data.getBytes());  }  line = br.readLine();  }  br.close();  file.close();  fos.close();  }    //**运行yq [指定输入文件] [指定输出文件]；（按要求排序）**  public void getOneProvince(String inpath, String outpath, String shen) throws Exception {  FileReader file = new FileReader(inpath);  BufferedReader br = new BufferedReader(file);  FileOutputStream fos = new FileOutputStream(outpath);    String line = br.readLine();    String data = shen + "\n";  fos.write(data.getBytes());    String[] city = new String[1000];  while (line != null) {  String[] word = line.split("\t");  if (word[0].equals(shen) && !word[2].equals("0")) {  int con = Integer.parseInt(word[2]);  if (city[con] == null)  city[con] = word[1];  else  city[con] = city[con] + "," + word[1];  }  line = br.readLine();  }  for (int i = 999; i >= 0; i--) {  if (city[i] != null) {  String[] split = city[i].split(",");  if (split.length == 1) {  data = city[i] + "\t" + i + "\n";  fos.write(data.getBytes());  } else {  Arrays.sort(split);  for (String string : split) {  data = string + "\t" + i + "\n";  fos.write(data.getBytes());  }  }  }  }  br.close();  file.close();  fos.close();  }    //**3.运行yq [指定输入文件] [指定省份]；（按要求排序）**  public void getAllProvinceToFile(String inpath, String outpath) throws Exception {  FileReader file = new FileReader(inpath);  BufferedReader br = new BufferedReader(file);  FileOutputStream fos = new FileOutputStream(outpath);    String line = br.readLine();  String shen = line.split("\n")[0];  int sum = 0;  String[] shenfen = new String[3000];    while(line != null) {  String[] word = line.split("\t");  if(word[0].equals(shen)) {  sum += Integer.parseInt(word[2]);  }else {  shenfen[sum] = shen;  sum = 0;  shen = word[0];  }  line = br.readLine();  }  br.close();  shenfen[sum] = shen;  for(int i = 2999; i > 0; i--) {  if(shenfen[i] != null) {  write\_for\_count(shenfen[i],inpath,fos);  }  }  file.close();  br.close();  fos.close();  }  //指定输入省份按排序输出  private void write\_for\_count(String shen, String inpath, FileOutputStream fos) throws Exception {  FileReader file = new FileReader(inpath);  BufferedReader br = new BufferedReader(file);  String line = br.readLine();  String data = "\n"+shen+"\n";  fos.write(data.getBytes());  String[] city = new String[1000];  while (line != null) {  String[] word = line.split("\t");  if (word[0].equals(shen) && !word[2].equals("0")) {  int con = Integer.parseInt(word[2]);  if (city[con] == null)  city[con] = word[1];  else  city[con] = city[con] + "," + word[1];  }  line = br.readLine();  }    for (int i = 999; i > 0; i--) {  if (city[i] != null) {  String[] split = city[i].split(",");  if (split.length == 1) {  data = city[i] + "\t" + i + "\n";  fos.write(data.getBytes());  } else {  Arrays.sort(split);  for (String string : split) {  data = string + "\t" + i + "\n";  fos.write(data.getBytes());  }  }  }  }  br.close();  file.close();  }      }  public class yq {  //main方法测试  public static void main(String[] args) throws Exception {  core c = new core();  if(args.length == 0) c.getAllProvince();  else if(args.length == 2) c.getAllProvinceToFile(args[0], args[1]);  else if(args.length == 3) c.getOneProvince(args[0], args[1], args[2]);  }  } |   代码思路：首先通过读取文件的每行分割，将每个省的总人数统计起来，按人数为下标省份为值，存进字符数组中，再从大下标像小下标遍历。      定义方法，将人数从大到小的省份，输入路径，输出路径，输出流当参数传入方法中，再用同样的方法以人数为下标，城市为值存入字符串数组中，如果人数一样，则通过“，”隔开，再遍历字符串数组，如果有超过两个的值，分割，再排序，就可以实现从大到小排序，并且人数相同可以按拼音排序。      同理也可以用以上方法操作输出指定省份的信息。  5、过程截图  创建py.java文件    编辑打开输入上述代码    在当前目录打开终端    编译java文件//这里因为编码问题报错，所有固定为UTF-8来编译  出现.class文件  执行文件.class文件    得到yq.txt文件    打开得到全部省的感染人数，没有进行排序    再执行文件.class文件    得到文件    内容为省份人数从大到小，省内人数也是从大到小    改变参数继续执行    得到输出文件    其内容是输出指定省份的信息并排序    四、实验心得  在本次实验中选择使用java编程语言，在编程时引用输入输出流以及String字符串的split方法对字符串进行分割，再将字符串转换成byte数组输出到文本种。这里完全使用终端和文本进行操作，不用编译器就可执行。注意的是要对数据进行筛选，对于感染人数为0的地区要删除。还需要判断输入的数据是否为3个或者为2个分别进行操作。以人数为下标，城市为值存入字符串数组中，从大到小遍历，就可以得到排序的城市。如果人数相同，使用Arrays.sort(split);方法进行排序，这里需要注意的是，这个排序对拼音有效，对汉字无效。在对三种不同处理的封装上，我选择定义一个core类，分别写出三种方法，在主函数中，针对参数个数进行不同的操作，得出不同的结果。 |

注：实验报告的内容及格式可由学院根据学科专业特点确定；全校各专业必须使用学校统一封面。