**第一部分**

**编写一个Java源程序，其中包含编写一个包含主方法main的公共类（访问权限为public的类），主方法main中完成的任务是：**

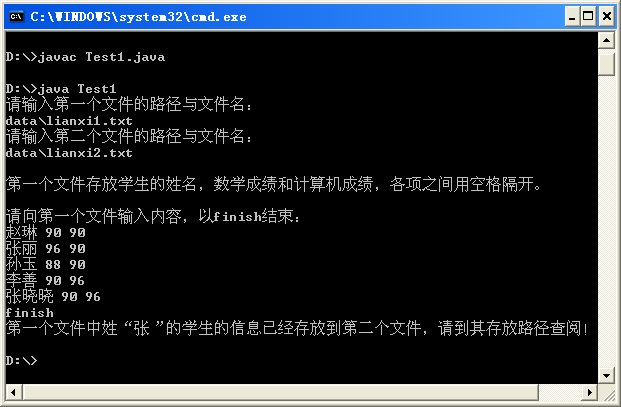
**（1）从键盘上输入两个文件的路径名及其文件名。**

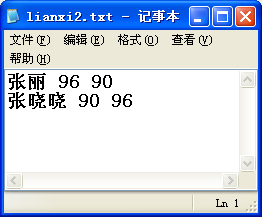
**（2）从键盘上输入若干学生的信息（姓名，数学成绩，计算机成绩，各项之间用空格隔开），以finish结束，将这些学生的信息写入第一个文件。**

**（3）将第一个文件作为输入流的源，找出姓“张”的学生，将他们的信息写入第二个文件。**

**提示：将输入的每一行学生的信息作为一个字符串来处理。**

**程序的运行结果如下图所示（供参考）：**

****

****

**第二部分**

**编写一个Java源程序，其中的文件读写部分使用RandomAccessFile类来完成，程序实现的功能如下：**

**编写一个包含主方法main的公共类（访问权限为public的类），主方法main中完成的任务是：**

**（1）将第一部分产生的文件中的姓名“李善”改为“王山”。**

**（2）之后，在文件的最后继续添加新的学生信息。**

**（如果没有保存第一部分的文件，也可以在记事本中输入若干同学的姓名和成绩等信息作为第二部分的原文件。）**

**提示：**

**（1）用RandomAccessFile来对文件进行操作时，读写模式应该是“rw”。**

**（2）假设有RandomAccessFile类的对象raf，则**

**String s=raf.readLine();**

**表示从文件中读取一行，将读取的内容放入字符串对象s中。但是，如果读入的内容有汉字时，可能会读入乱码，导致读入的内容出错。通常采用如下语句**

**byte[] b=s.getBytes("ISO-8859-1");**

**String s1=new String(b);**

**进行转换，之后，用数组b创建的新字符串s1就没有乱码问题。**

**（3）在改名字的程序段中，应该存取两个指针的位置，一个是读完上一行之后的文件指针的位置，另一个是读完当前行之后的文件指针的位置（想一想，为什么？）。使用RandomAccessFile类的对象调用getFilePointer()和seek()方法实现上述任务。**

**（4）向文件中添加信息时，假设从键盘上输入的学生信息存放在字符串s中，则 raf.write(s.getBytes());**

**可以向文件写入字符串。**

**程序的运行结果如下图所示（供参考）：**

