**《Java语言程序设计基础教程》**

**上机实验指导手册**

**实验十 文件与输入输出流**

## 【目的】

1. 熟悉文件File类的方法；
2. 熟悉常见的文件输入流和输出流；
3. 理解字节流和字符流的联系与区别；
4. 了解对象序列化的方法。

## 【内容】

### 写入文本信息

在D盘下创建文件“hello.txt”，并使用FileWriter写入以下内容：

*枯藤老树昏鸦，*

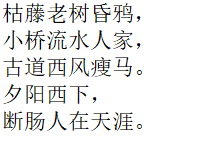
*小桥流水人家，*

*古道西风瘦马。*

*夕阳西下，*

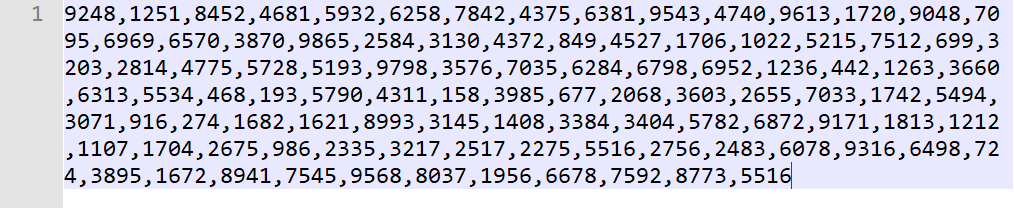
*断肠人在天涯。*

然后再使用FileReader从文件“hello.txt”中读取文件，并打印至控制台：



### 读取文件中数字并排序

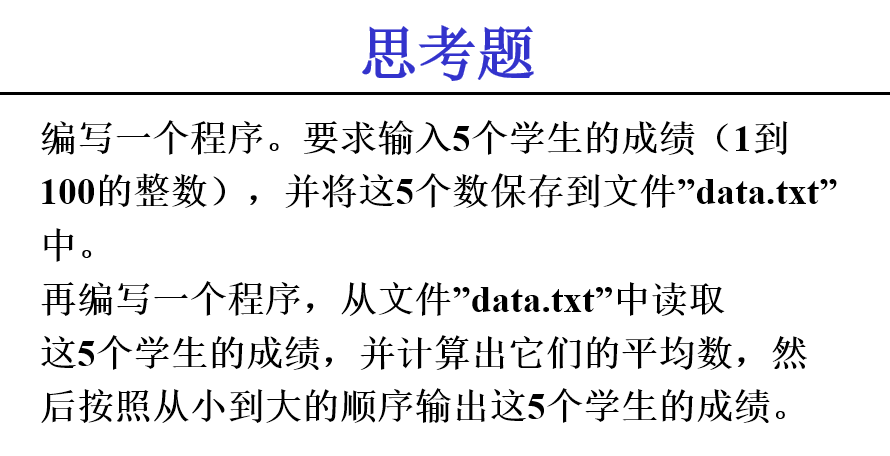
已给定文件“Trial\_10\_num.txt”，其中包含100个数字，如下图所示



请使用BufferedReader读取该文件，按照从大到小进行排序，将排序结果保存至新的文件（文件名自拟）并输出至控制台。

提示：

参考课堂案例



<https://github.com/imwyw/Java/blob/master/PPT/Codes/javaCh06_StuScore.java>

### 查找所有java源代码文件

查找D盘下所有java源代码文件，并将所有java代码文件的绝对路径保存至文件“find.txt”

提示如下：

1、需要使用File的实例方法listFiles()获取所有文件，代码如下：

File rootFile = **new** File("D:\\");

// 获取目录下所有文件

File[] fileArray = rootFile.listFiles();

2、需要判断当前文件是否为文件，并且是否为java代码文件，判断代码如下：

File testFile = **new** File("D:\\Hello.java");

// 是否为文件

System.***out***.println(testFile.isFile());

// 文件后缀是否是.java

System.***out***.println(testFile.getName().endsWith(".java"));

3、在寻找文件时，如包含目录，需要一直查找下去，所以需要使用递归方式查找，合理设计方法进行递归调用

4、写入文件建议使用缓冲文件输出方式，即BufferedWriter

## 【注意事项】

1. 提交产物为相关问题的回答以及java源代码（xxx.java文件）
2. 提交方式，通过博思平台进行提交，截止时间详见平台

**提升编码效率之eclipse智能提示的设置：**

(1). 打开Eclipse，选择打开" Window － Preferences"。

(2). 在目录树上选择"Java－Editor－Content Assist"，在右侧的"Auto-Activation"找到"Auto Activation triggers for java"选项。默认触发代码提示的就是"."这个符号。

(3). 在"Auto Activation triggers for java"选项中，将"."更改：.abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ

