编号



数据结构课程设计报告

二级学院 计算机科学与工程学院

专 业 软件工程

学生姓名 周翔辉

学 号 11603080122

指导教师 王森

时 间 2017.1

### 

### 数据结构课程设计报告

*三号、黑体、居中*

*空一行*

### 1. 题目1

###### 1.1 题目说明

利用哈夫曼编码进行信息通讯可以大大提高信道利用率，缩短信息传输时间，降低传输成本。但是，这要求在发送端通过一个编码系统对待传数据预先编码；在接收端将传来的数据进行译码(复原)。对于双工信道 (即可以双向传输信息的信道)，每端都需要一个完整的编/译码系统。试对于任意的一段文本（可能是直接输入的，也可能是保存在本地文件中或者网络上的)，写一个哈夫曼码的编译码系统。

[基本要求]  
一 个完整的系统应具有以下功能：  
(l)I:初始化 (Initialization)。从终端读入字符集大小 n，及 n 个字符和 m 个权值，建立哈夫曼树，并将它存于文件 hfmtree 中。  
(2)C:编码 (Coding)。利用已建好的哈夫曼树(如不在内存，则从文件 hfmtree 中读入)，对文件 tobetrans 中的正文进行编码，然后将结果存入文件 codefile 中。  
(3)D:解码(Decoding)。利用已建好的哈夫曼树将文件 codefile 中的代码进行译码，结果存入文件 textfile 中。  
(4)P:打印代码文件 (Print)。将文件 codefile 以紧凑格式显示在终端上，每行 50 个代码。同时将此字符形式的编码文件写入文件 codeprint 中。  
(5)T:打印哈夫曼树 (Tree printing)。将已在内存中的哈夫曼树以直观的方式 (树或凹入  
表形式)显示在终端上，同时将此字符形式的哈夫曼树写入文件 treeprint 中。

[实现提示]  
可以根据题目要求把程序划成 5 个模块，设计成菜单方式，每次执行一个模块后返回菜单。除了初始化(I)过程外，在每次执行时都经过一次读取磁盘文件数据。这是因为如果在程序执行后一直没有进行初始化(I)过程，为了能使后面的操作顺利进行，可以通过读取旧的数据来进行工作。比如：如果程序的工作需要的字符集和权值数据是固定的，只要在安装程序时进行一次初始(I)化操作就可以了。再次运行程序时，不管进行哪项操作都可以把需要的数据读入到内存。

对住房的属性问题，……，具有五大属性：

①区位固定性。建筑物通常和土地一样，……

*小四号、宋体，首行缩进两字*

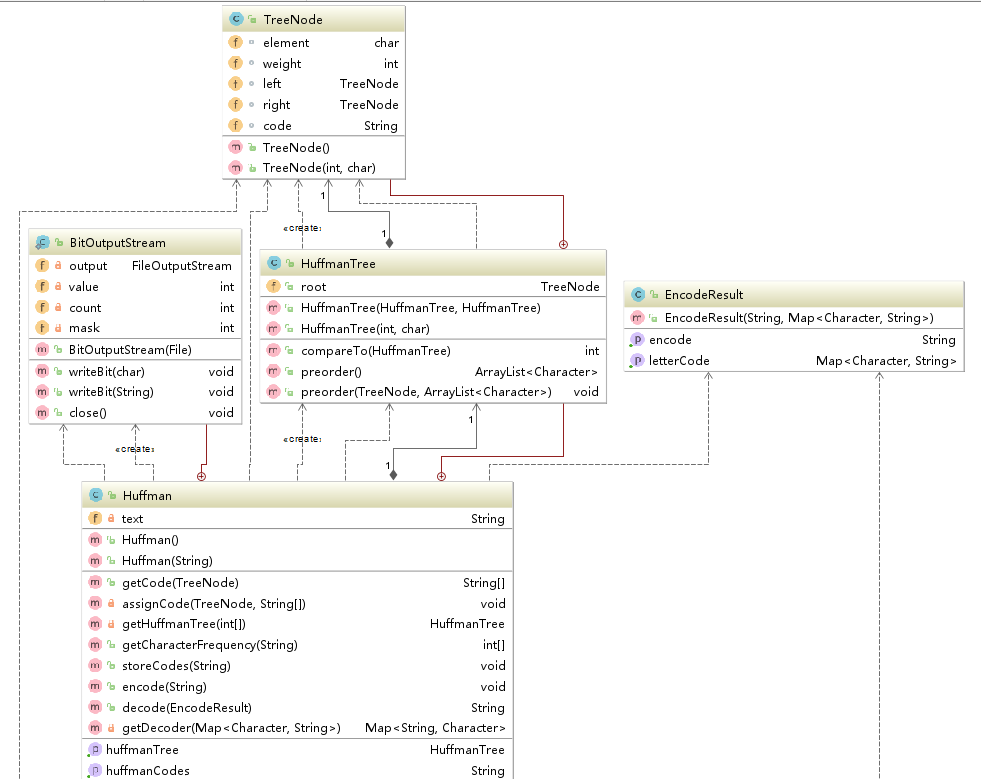
②异质性。由于区位条件、建筑材料、……

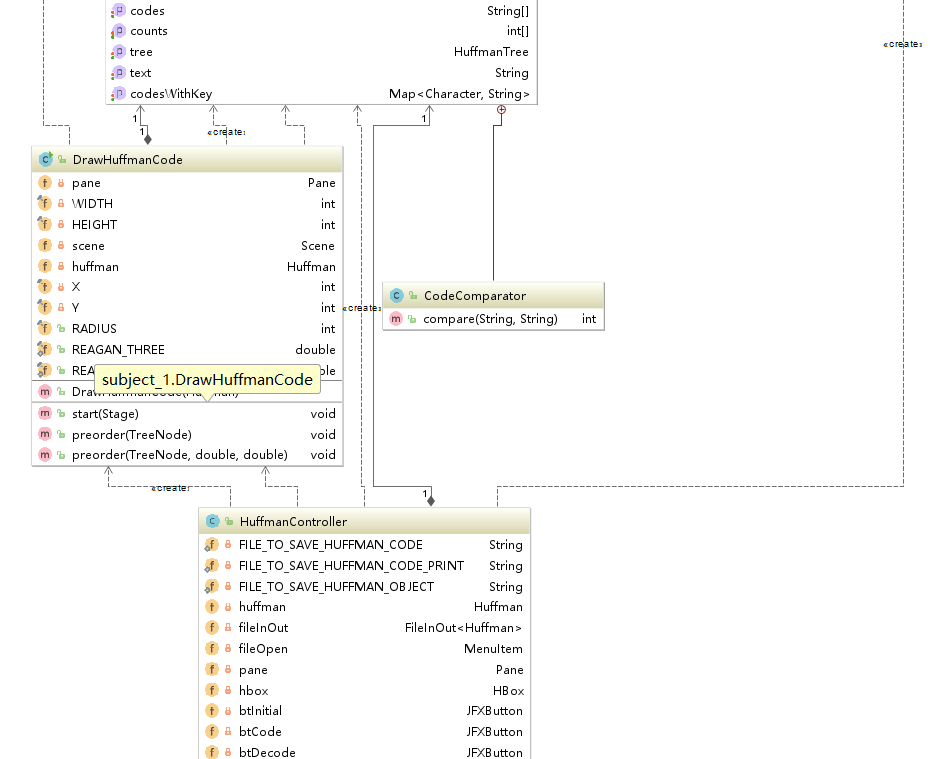
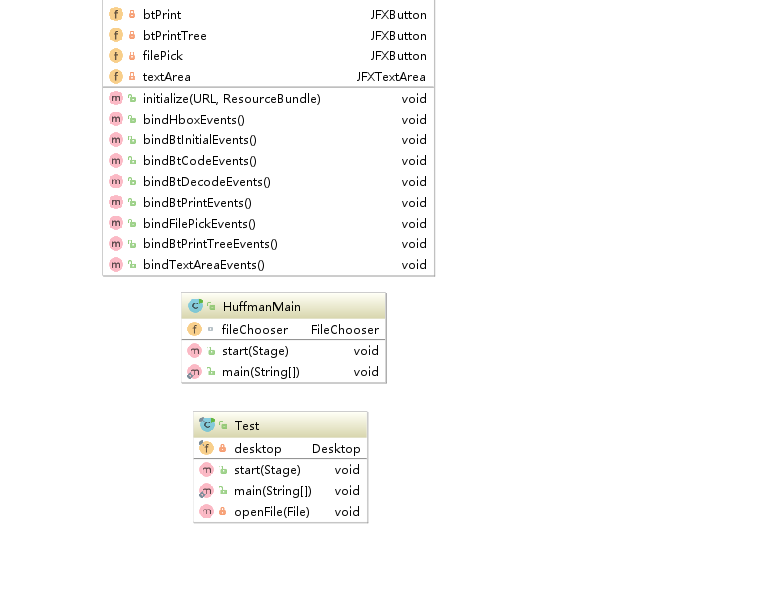
*小四号、宋体，首行缩进两字*

*上面为文档正文的格式*

### 1.2 题目设计

###### 1.2.1 系统静态模型



###### 1.2.2 存储结构设计

使用数组存储编码的的哈夫曼编码

###### 1.2.3 关键算法设计

1 编码

**public void** encode(String filename) **throws** IOException {  
 BufferedInputStream fileInput = **new** BufferedInputStream(  
 **new** FileInputStream(**new** File(filename)));  
 BitOutputStream output = **new** BitOutputStream(**new** File(filename + **".new"**));  
  
 **int** r;  
 **while** ((r = fileInput.read()) != -1 ) {  
 output.writeBit(**codes**[r]);  
 }  
}

2 解码

**public** String decode(EncodeResult decodeResult) {  
 *// 解码得到的字符串* StringBuffer decodeStr = **new** StringBuffer();  
 *// 获得解码器* Map<String, Character> decodeMap = getDecoder(decodeResult  
 .getLetterCode());  
 *// 解码器键集合* Set<String> keys = decodeMap.keySet();  
 *// 待解码的（被编码的）字符串* String encode = decodeResult.getEncode();  
 *// 从最短的开始匹配之所以能够成功，是因为哈夫曼编码的唯一前缀性质  
 // 临时的可能的键值* String temp = **""**;  
 *// 改变temp值大小的游标* **int** i = 1;  
 **while** (encode.length() > 0) {  
 temp = encode.substring(0, i);  
 **if** (keys.contains(temp)) {  
 Character character = decodeMap.get(temp);  
 decodeStr.append(character);  
 encode = encode.substring(i);  
 i = 1;  
 } **else** {  
 i++;  
 }  
 }  
 **return** decodeStr.toString();  
}

### 1.3 系统测试

（下面为测试案例模板，至于一个题目应设计多少个测试案例，根据题目情况来定。测试案例如何设计，请自行到网上查找资料。此说明最终打印时要删除）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试用例编号 | 1 | | 版本号 | | 1 |
| 测试环境 | Windows10 JDK1.8 IDEA2017.2 | | | | |
| 用例名称 |  | | | | |
| 前提条件 |  | | | | |
| 测试步骤 |  | | | | |
| 输入数据 |  | | | | |
| 预期输出 |  | | | | |
| 实际输出 |  | | | | |
| 问题描述 |  | | | | |
| 设计人 |  | 设计日期 | |  | |
| 测试人 |  | 测试日期 | | |  |
| 再测试人 |  | 再测试日期 | | |  |
| 问题修改摘要 |  | | | | |
| 修改人 |  | | 修改日期 | |  |

### 2. 题目1

### 2.1 题目说明

对住房的属性问题，……，具有五大属性：

①区位固定性。建筑物通常和土地一样，……

*小四号、宋体，首行缩进两字*

②异质性。由于区位条件、建筑材料、……

*小四号、宋体，首行缩进两字*

*上面为 文档正文的排版格式*

### 2.2 题目设计

###### 2.2.1 系统静态模型

###### 2.2.2 存储结构设计

###### 2.2.3 关键算法设计

### 2.3 系统测试

（下面为测试案例模板，至于一个题目应设计多少个测试案例，根据题目情况来定。测试案例如何设计，请自行到网上查找资料。此说明最终打印时要删除）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试用例编号 |  | | 版本号 | |  |
| 测试环境 |  | | | | |
| 用例名称 |  | | | | |
| 前提条件 |  | | | | |
| 测试步骤 |  | | | | |
| 输入数据 |  | | | | |
| 预期输出 |  | | | | |
| 实际输出 |  | | | | |
| 问题描述 |  | | | | |
| 设计人 |  | 设计日期 | |  | |
| 测试人 |  | 测试日期 | | |  |
| 再测试人 |  | 再测试日期 | | |  |
| 问题修改摘要 |  | | | | |
| 修改人 |  | | 修改日期 | |  |

说明：如果做的多于2道，请继续按照上面的格式处理。但要注意修改标题的序号及总结的标题序号。

### 3.课程设计总结

工作意识、工作方法、问题及解决过程、