

实验四 二叉树的实现与应用

实验目的

1. 掌握二叉树的存储结构，实现二叉树的基本操作
2. 合理设计运用数据结构，编写有效算法解决 Huffman 编译码问题

实验内容

题 1: Huffman 编码与译码问题

哈夫曼 (Huffman) 编码是 David Huffman 于 1952 年提出的一种无损压缩编码方法，能够依据字符在信源中的出现概率来构造可变长码字，将其转换为可以用 0、1 序列表达的比特流。用户 Alice 和 Bob 正在寻找一种最佳编码和译码方案，期望能够利用某种信息传输通道（简称信道）实现信息的无损传输。已知该信道仅允许使用 1，0 两种符号作为信道的输入和输出，并且具有非常完美的无差错传输能力。请帮助 Alice 和 Bob，分别设计 Huffman 编码器和译码器，并将其应用在发送端和接收端，实现信息通信过程。提醒注意，对于双工信道（即可以双向传输信息的信道），每个用户都需要一个完整的编、解码器。

实验要求

本次实验要求实现的一个完整的 huffman 编解码系统，应该具有以下功能：

1. 初始化 (Initialization)：从文件 **CharSet.txt** 读入字符集大小 n 、 n 个字符和对应的 n 个权值，建立 Huffman 树，并将它存入文件 **hfmtree.txt** 中。
2. 编码 (Encoding)：利用已经建好的 Huffman 树（如果不在内存，则应从文件 **hfmtree.txt** 中读取），对文件 **ToBeTran.txt** 中的正文进行编码，然后将结果存入文件 **CodeFile.txt** 中。
3. 解码 (Decoding)：利用已经建立好的 Huffman 树将文件 **CodeFile.txt** 中的代码

进行解码，结果存入 **TextFile.txt** 中。

4. 打印代码文件 (Print)：将文件 **CodeFile.txt** 以紧凑的格式显示在终端上，每行 50 个代码。同时将此字符形式的编码文件写入文件 **CodePrint.txt** 中。
5. 打印 Huffman 树 (TreePrinting)。将已经在内存中的 Huffman 树以直观的形式（树或者凹入的形式）显示在终端上，同时将此字符形式的 Huffman 树写入文件 **TreePrint.txt** 中。
6. 用户界面可以设计为“菜单”方式：显示上述功能符号，再加上一个退出功能“Q”，表示退出 (quit)。用户键入一个选择功能符，此功能执行完毕后再显示此菜单，直至某次用户选择了 Q 为止。

请认真阅读实验内容，根据实验要求做好问题分析与数据建模，在此基础上完成详细代码设计和上机调试。建议在 5 学时内完成本次实验的所有内容。

参考示例

用下表给出的字符集和频度的统计数据建立 Huffman 树，并对以下报文进行编码和译码：“THIS PROGRAM IS MY FAVORITE”。

表 4-1

字符		A	B	C	D	E	F	G	H	I	J	K	L	M
频度	186	64	13	22	32	103	21	15	47	57	1	5	32	20
字符	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
频度	57	63	15	1	48	51	80	23	8	18	1	16	1	

```
----哈夫曼编解码-----
----1.初始化(I)-----
----2.编码(E)-----
----3.译码(D)-----
----4.打印代码文件(P)-----
----5.打印Huffman树(T)-----
----6.退出(Q)-----
请输入指令：
I
初始化完成，Huffman树已存入hfmTree.txt中
E
编码完成，编码结果已存入CodeFile.txt中
D
解码完成，解码结果已存入TextFile.txt中
P
11010001011000111111000100010100110000100101010110
01011101100011111110010100011111110011101011000001
001001001101101010
编码结果打印完成，打印结果已存入CodePrint.txt中
```

图 4-1 参考示例的运行结果

以下给出了一种凹入法表示的树形结构示意图，可用于二叉树上所有结点的逐行屏幕打印或文件输出。其基本思想是利用凹入（缩进）程度表达结点之间的层次关系。例如，根结点是第一层，则在屏幕第一行靠左顶格输出根结点信息，从第二行开始依次输出根结点的左子树和右子树所辖结点信息。

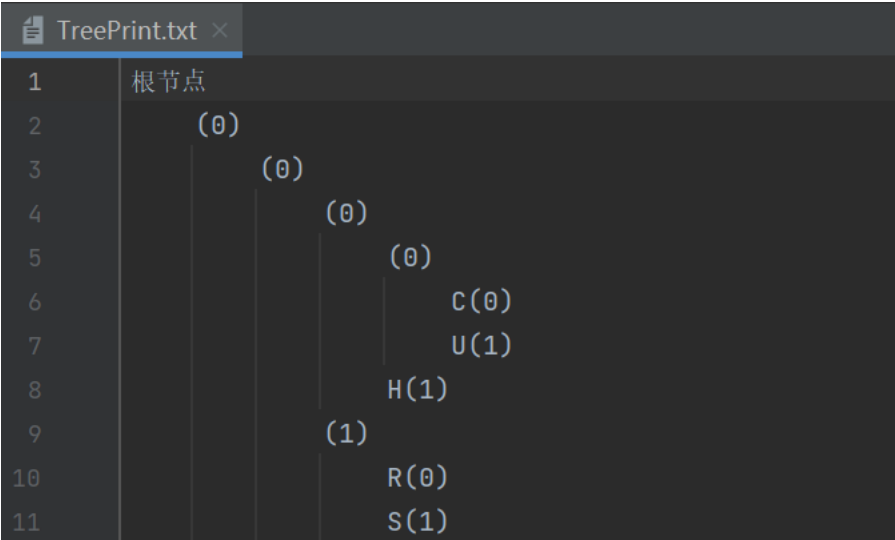


图 4-2 凹入法表示一棵树