

实验三 二叉树的应用

崔士强 PB22151743

1 问题描述

程序需要完成的操作包括：

1. 输入电文字符串
2. 统计电文字符集和每种字符在电文中出现的次数
3. 构建 huffman 树
4. 产生每种字符的 huffman 编码
5. 将电文串翻译成比特流
6. 对电文比特流进行解码

输出信息如下所示：

原始字符	编码后字符
aaabbbcccd...	一串乱码
频率信息	编码信息
a 5000	a 1100
b 20000	b 111
c 8000	c 1101
d 4000	d 0
e 27000	e 10
.....

2 算法描述

2.1 数据结构描述

本程序所用到的 Huffman 树存储结构如下所示：

```
1  typedef struct {
2      int weight;
3      int parent, lchild, rchild;
4  } HTNode, *HuffmanTree;
5
6  typedef char **HuffmanCode;
```

2.2 程序结构描述

程序中的函数声明如下:

```
1 void Select(HuffmanTree HT, int n, int &s1, int &s2);
2 // 在HT[1..n]中选择parent为0且weight最小的两个结点, 其序号分别为s1和s2
3
4 void HuffmanCoding(HuffmanTree &HT, HuffmanCode &HC, int *weight, int n);
5 // 用n个字符的权值weight构造Huffman树HT,并求出n个字符的Huffman编码HC
6
7 int ReadFile(char *filename, char *chars);
8 // 从文件中逐个字节读取信息
9
10 int GetFrequency(char *chars, int *freq, char *charList, int &n);
11 // 获取待处理文本字符的频率信息
12
13 void WriteCodedString(char *CodedChar, char *chars, HuffmanCode HC, char *CharList, int
    FLength, int CharNum);
14 // 将比特流写入文件
15
16 void Decode(char *CodedChar, HuffmanCode HC, char *CharList, char *DecodedChar);
17 // 根据比特流还原文件
```

3 调试分析

3.1 测试数据

选取纯文本, .mp4 文件, .bmp 文件, .exe 文件进行测试

3.2 问题及解决方法

测试中发现程序无法处理除纯文本文件之外的文件类型, 经调试发现数组最大值过小导致读取文件时字符频率的获取不正确。

解决方法: 数组长度改为 1000000

4 算法的时空分析

我们可以从以下几个方面分析复杂度:

1. **Select** 函数: 这个函数在 Huffman 树数组 HT 中寻找两个最小的未被选中的节点。它通过两次遍历整个数组来实现, 因此其时间复杂度为 $O(n)$ 。
2. **HuffmanCoding** 函数: 这个函数构建 Huffman 树。它首先初始化一个大小为 $2n - 1$ 的数组, 然后通过 $n - 1$ 次迭代来构建树。每次迭代中, 它调用 **Select** 函数 ($O(n)$), 然后执行常数时间的操作。因此, 这个函数的总体时间复杂度为 $O(n^2)$ 。
3. **ReadFile** 函数: 这个函数读取文件内容。它的时间复杂度取决于文件的大小, 假设文件大小为 f , 则时间复杂度为 $O(f)$ 。

4. **GetFrequency** 函数: 这个函数计算每个字符的频率。它的时间复杂度为 $O(n^2)$, 其中 n 是文件内容的长度。
5. **WriteCodedString** 和 **Decode** 函数: 这两个函数的时间复杂度取决于输入字符串的长度和字符集的大小。在最坏情况下, 它们的时间复杂度可以达到 $O(nm)$, 其中 n 是输入字符串的长度, m 是字符集的大小。

这个程序的主要瓶颈在于 `HuffmanCoding` 函数和 `GetFrequency` 函数，它们都有 $O(n^2)$ 的时间复杂度。这意味着对于大型输入，这个程序的性能可能会受到显著影响。

5 测试结果及分析

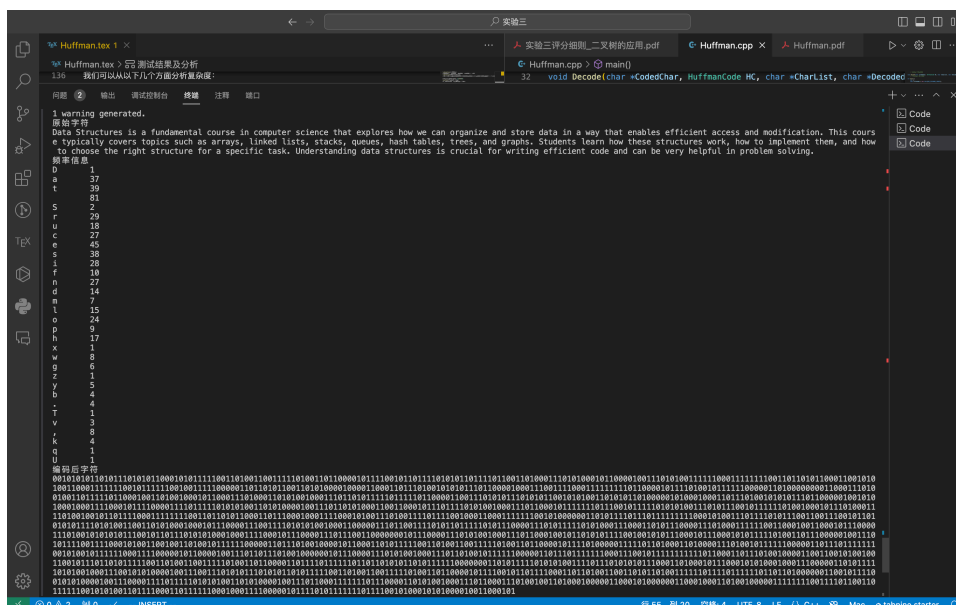


图 1: 测试样例 1

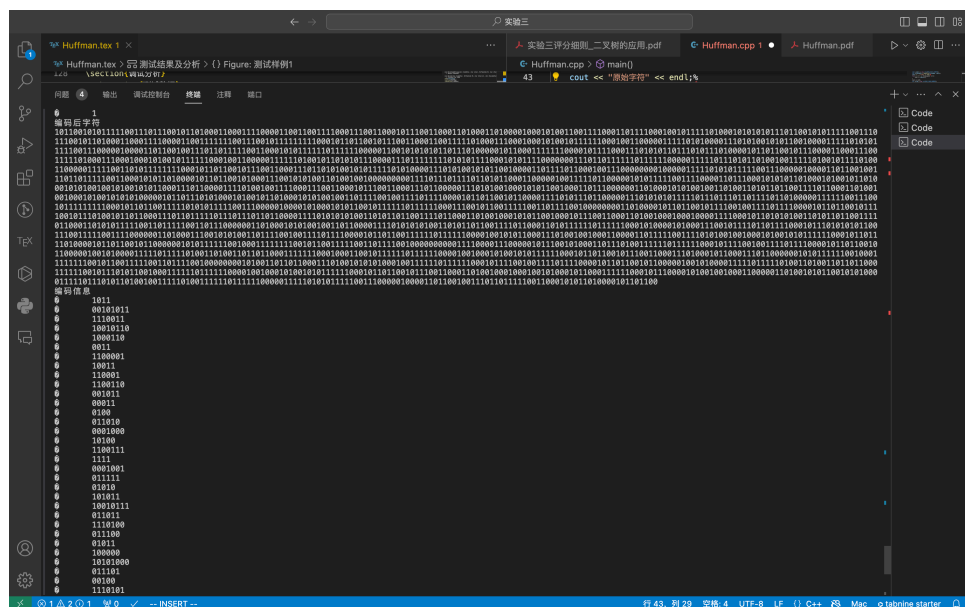
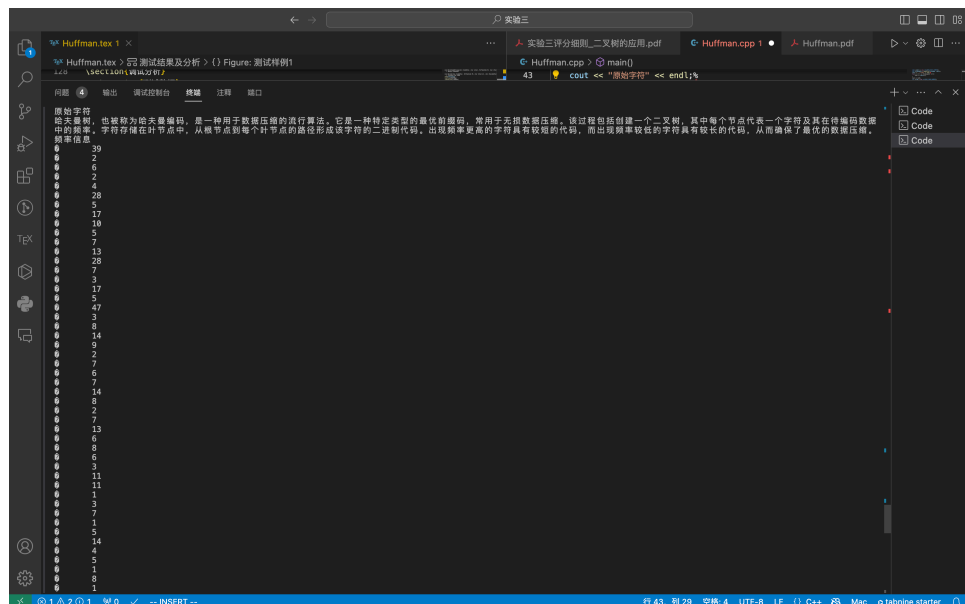


图 2: 测试样例 2

6 实验体会和收获

熟练掌握了二叉树的相关结构、性质，以及 Huffman 算法的原理。