

最优 2 元树在通信编码中的应用

2153726 罗宇翔

(经济与管理学院, 信息管理与信息系统)

目录

1	实验目的	1
2	实验内容	1
3	实验环境	1
3.1	Visual Studio Code	1
3.2	g++	1
4	实验原理和方法	1
4.1	树的带权深度	1
4.2	构造步骤	2
5	实验代码	2
6	实验数据及结果分析	6

1 实验目的

使用 $C++$ 生成最优二元树。

2 实验内容

输入字符频率，输出其对应的哈夫曼编码。

3 实验环境

3.1 Visual Studio Code

Version: 1.89.1

Browser:

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)

AppleWebKit/537.36 (KHTML, like Gecko) Code/1.89.1

Chrome/120.0.6099.291 Electron/28.2.8 Safari/537.36

3.2 g++

Apple clang version 14.0.0 (clang-1400.0.29.202)

Target: x86_64-apple-darwin21.6.0

Thread model: posix

4 实验原理和方法

4.1 树的带权深度

设二叉树具有 n 个带权叶结点，从根结点到各叶结点的路径长度与相应叶节点权值的乘积之和称为树的带权路径长度 (*Weighted Path Length of Tree*, WPL)。

设 w_i 为二叉树第 i 个叶结点的权值， l_i 为从根结点到第 i 个叶结点的路径长度，则 WPL 计算公式如下：

$$WPL = \sum_{i=1}^n w_i l_i$$

4.2 构造步骤

1. 初始化：由给定的 n 个权值构造 n 棵只有一个根节点的二叉树，得到一个二叉树集合 F 。
2. 选取与合并：从二叉树集合 F 中选取根节点权值最小的两棵二叉树分别作为左右子树构造一棵新的二叉树，这棵新二叉树的根节点的权值为其左、右子树根结点的权值和。
3. 删除与加入：从 F 中删除作为左、右子树的两棵二叉树，并将新建立的二叉树加入到 F 中。
4. 重复 2、3 步，当集合中只剩下一棵二叉树时，这棵二叉树就是霍夫曼树。

5 实验代码

```
#include <iostream>
#include <queue>
#include <sstream>
#include <unordered_map>
#include <vector>
using namespace std;

struct Node { // define node structure
    int weight;
    Node *l;
    Node *r;

    Node(int w) : weight(w), l(nullptr), r(nullptr) {}
};

struct Compare { // define comparison structure
    bool operator()(Node *a, Node *b) { return a->weight > b->weight; }
```

```
};
```

```
Node *build_HFM_tree(vector< int > &freq) { // build tree
    priority_queue< Node *, vector< Node * >, Compare > pq;
    for (int i = 0; i < freq.size(); i++) { // build priority
queue
        Node *node = new Node(freq[i]);
        pq.push(node);
    }

    while (pq.size() > 1) { // build HFM tree
        Node *left = pq.top();
        pq.pop();
        Node *right = pq.top();
        pq.pop();

        Node *parent = new Node(left->weight + right->weight);
        parent->l = left;
        parent->r = right;
        pq.push(parent);
    }
    return pq.top();
}
```

```
void gen_code(Node *root, unordered_map< int, string > &codes,
string code = "0") {
    if (root == nullptr)
        return;
    if (root->l == nullptr && root->r == nullptr)
        codes[root->weight] = code;
    gen_code(root->l, codes, code + "0");
    gen_code(root->r, codes, code + "1");
}
```

```
}
```

```
int main() {
    while (true) {
        vector< int > freq;
        vector< string > invalid;
        vector< int > invalidIdx;
        cout << "Input frequency(Qq to quit)\n";

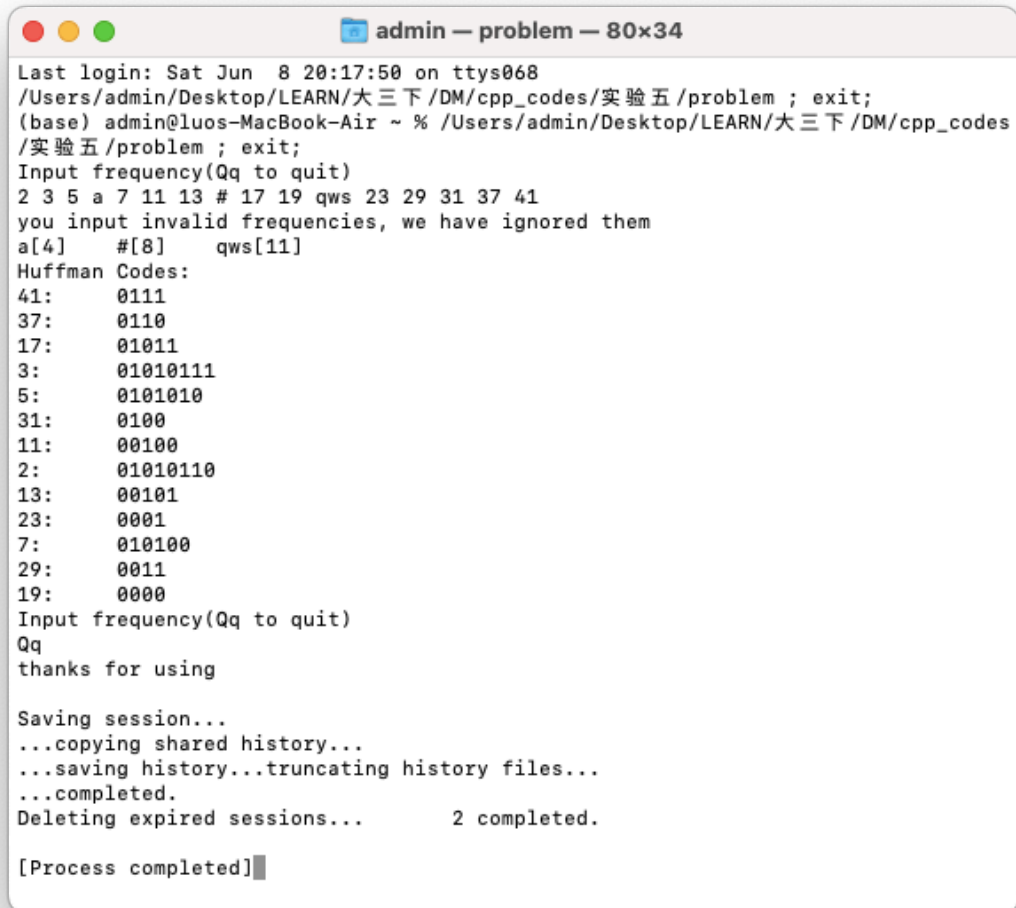
        string input;
        getline(cin, input);
        if (input == "Qq") {
            cout << "thanks for using\n";
            break;
        }
        istringstream ss(input);
        string token;
        int idx = 0;
        while (ss >> token) {
            bool isInteger = true;
            for (int i = 0; i < token.size(); i++) {
                if (!isdigit(token[i])) {
                    invalid.push_back(token);
                    invalidIdx.push_back(idx);
                    isInteger = false;
                    break;
                }
            }
            if (isInteger) {
                int number = stoi(token);
                if (number > 0) {
                    freq.push_back(number);
                }
            }
        }
    }
}
```

```

        }
    }
    idx++;
}
if (freq.size() == 0) {
    cout << "none of the inputs is valid, please retry\n";
    continue;
}
Node *tree = build_HFM_tree(freq);
unordered_map< int, string > codes;
gen_code(tree, codes);
cout << "you input invalid frequencies, we have ignored them\n";
for (int i = 0; i < invalid.size(); i++) {
    printf("%s[%d]\t", invalid[i].c_str(), invalidIdx[i] + 1);
}
cout << "\nHuffman Codes:\n";
for (unordered_map< int, string >::iterator it = codes.begin(); it != codes.end(); ++it) {
    pair< int, string > it_pair = *it;
    cout << it_pair.first << ":\t" << it_pair.second << endl;
}
}
return 0;
}

```

6 实验数据及结果分析

A terminal window titled "admin — problem — 80x34" showing the execution of a program. The window has a light gray title bar with three colored window control buttons (red, yellow, green) on the left. The terminal text is as follows:

```
Last login: Sat Jun 8 20:17:50 on ttys068
/Users/admin/Desktop/LEARN/大三下/DM/cpp_codes/实验五/problem ; exit;
(base) admin@luos-MacBook-Air ~ % /Users/admin/Desktop/LEARN/大三下/DM/cpp_codes
/实验五/problem ; exit;
Input frequency(Qq to quit)
2 3 5 a 7 11 13 # 17 19 qws 23 29 31 37 41
you input invalid frequencies, we have ignored them
a[4]    #[8]    qws[11]
Huffman Codes:
41:     0111
37:     0110
17:     01011
3:       01010111
5:       0101010
31:      0100
11:      00100
2:       01010110
13:      00101
23:      0001
7:       010100
29:      0011
19:      0000
Input frequency(Qq to quit)
Qq
thanks for using

Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.
Deleting expired sessions...      2 completed.

[Process completed]
```

Figure 1: 程序运行结果

见上图，程序忽略非法输入并输出正确结果。