实验环境:

```
OS:ubuntu 18.1.04 LTS
compiler:c++11,g++
MakeTool:CMake
```

实验过程

```
#include <bits/stdc++.h>

using namespace std;

class Node
{
public:
    int value;
    Node *parent;
    Node *left;
    Node *right;
    char key;

    Node()
    {
        key = '0';
        value = 0;
        parent = nullptr;
        left = nullptr;
        right = nullptr;
    }

};

struct cmp
{
    bool operator()(Node* a, Node* y)
    {
        return a->value > y->value;
    }
};
Node* Huffman(Node nodes[], const char *ch, const int *f, int n)
{
    priority_queue<Node*, vector<Node*>, cmp> q;
    for(int i = 0; i < n; ++i)
    {
        nodes[i].key = ch[i];
        nodes[i].value = f[i];
    }
    for(int i = 0; i < n; ++i)
        q.push(&nodes[i]);
    int k = n;
    Node* node1;
    Node* node2;
    while(!q.empty())
    {
        node1 = q.top();
        q.pop();
        if (q.empty())
            break;
        node2 = q.top();
        q.pop();
        nodes[k].value = node1->value + node2->value;
        node1->parent = &nodes[k];
        node2->parent = &nodes[k];
        nodes[k].left = node1;
        nodes[k].right = node2;
        nodes[k].key = k + '0';
        q.push(&nodes[k]);
        k++;
    }
    return node1;
}

void print(Node *node, string id)
{
    if (node->right == nullptr && node->left == nullptr)
    {
        cout << node->key << " : " << id << endl;
        return ;
    }
    if(node->left != nullptr)
    {
        print(node->left, id+"0");
    }
    if(node->right != nullptr)
    {
        print(node->right, id+"1");
    }
}
int main()
{
    int n;
    cin >> n;
    int *f = new int[n+1];
    char *ch = new char[n+5];
    Node nodes[2*n + 10];
    for (int i = 0; i < n; ++i)
    {
        cin >> ch[i] >> f[i];
    }
    Node *root = Huffman(nodes, ch, f, n);
    printf("huffman编码 : \n");
    print(root, "");
    return 0;
}

/*
5
b 5
c 6
j 2
```

```
m 9
p 7
 */
```

运行结果

input:

```
5
b 5
c 6
j 2
m 9
p 7
```

output:

```
huffman编码 :
c : 00
p : 01
j : 100
b : 101
m : 11
```