安 徽 新 华 学 院 实 验 报 告

姓 名 张卢 学号 1632104249 专业班级 16软件2班

课程名称 数据结构 实验名称 哈夫曼树、哈夫曼编码和解码

实验日期 2017.11.20 同组人员 指导老师 汪红霞 得分

**实 验 项 目 五**

**【实验名称】**哈夫曼树、哈夫曼编码和解码

**【实验类型】**基础性实验

**【实验目的与要求】**

* 掌握哈夫曼树的含义；
* 掌握哈夫曼树的建立，哈夫曼编码，哈夫曼树解码。

**【实验主要仪器设备及台套数】**

一人一台安装Visual C++的电脑

**【实验学时】** 2学时

**【实验内容】**

1. 实现哈夫曼数的建立、遍历的实现。

#include <iostream.h>

const int n=8; //n表示叶子数目

const int m= 2\*n-1 ;

//m为森林中树的棵数

class tree

{ public:

float weight; //权值

int parent; //双亲

int lch,rch; //左、右孩子

void creathuffmantree( );

};

tree hftree[m+1];

void tree::creathuffmantree( )

{ int i,j ,p1,p2;

float s1,s2;

for(i=1;i<=m;i++)

{hftree[i].parent=0;

hftree[i].lch=0;

hftree[i].rch=0;

hftree[i].weight-=0;

}

cout<<"输入权值: "<<endl;

for(i=1;i<=n;i++)

cin>>hftree[i].weight; //输入权值

for(i=n+1;i<=m;i++) //进行n-1次合并

{

p1=p2=0; // p1,p2分别指向两个权最小的值的位置

s1=s2=32767; // s1,s2 代表两个最小权值

for(j=1;j<=i-1;j++) //选两个最小值

if(hftree[j].parent==0) //该权值还没有被选中

if(hftree[j].weight<s1)

{s2=s1;

s1=hftree[j].weight;

p2=p1;

p1=j;

}

else

if(hftree[j].weight<s2)

{s2=hftree[j].weight;

p2=j;

}

hftree[p1].parent=i;

hftree[p2].parent=i;

hftree[i].lch=p1;

hftree[i].rch=p2;

hftree[i].weight=hftree[p1].weight+hftree[p2].weight;

}

cout<<"输出合并后的结果: "<<endl;

for(i=1;i<=m;i++){

if(i==m)

{

cout<<"哈夫曼数为: "<<" "<<hftree[i].weight<<" "<<hftree[i].parent<<" "<<hftree[i].lch<<" "<<hftree[i].rch<<endl;

break;

} //输出合并后的结果

cout<<"第"<<i<<"次: "<<" "<<hftree[i].weight<<" "<<hftree[i].parent<<" "<<hftree[i].lch<<" "<<hftree[i].rch<<endl;

}

}

void main()

{ tree t;

cout<<" 欢迎测试！！！！ "<<endl;

cout<<"----------测试开始-----------"<<endl;

cout<<"首先调用哈夫曼树构造的函数：huffman"<<endl;

cout<<"按下标顺序输入叶子的权重:"<<endl;

cout<<" 0 1 2 3 4 5 6 7 "<<endl;

t.creathuffmantree();

}

**【实验心得】**

理解了哈夫曼树的含义以及他的算法过程，明白哈夫曼树的建立，哈夫曼编码，哈夫曼树解码，运用cout提示以及for循环，if等基本语句，不仅展现哈夫曼树的遍历过程，也使得程序更具有健壮性