广度优先搜索算法：

代码：

#include<iostream>

#include<vector>

#include<queue>

#include<memory.h>

using namespace std;

vector<vector<int> > tree;//声明一个二维向量

int flag[10];//用于搜索到了节点i的第几个节点

queue<int> M;//声明一个队列

int ar\_tree[8] = { 1,1,1,3,5,3,5,7 };

void BFS(int node) {

int temp;

cout << node << " ";

//从队列中取出第一个节点

int m\_first = M.front();

M.pop();

while(flag[node] < tree[node].size()) {

temp = tree[node][flag[node]];

flag[node]++;

//把temp加入队列中

M.push(temp);

}

if (!M.empty()) {

BFS(M.front());

}

}

int main() {

ios::sync\_with\_stdio(false);

memset(flag, 0, sizeof(flag));

int i,temp;

tree.resize(10);//图中的数共有九个节点

//生成树

for (i = 2; i <=9; i++) {

temp = ar\_tree[i - 2];

tree[temp].push\_back(i);//表示第i个节点为第temp个节点的子节点

}

//BFS

cout << "BFS过程：" << endl;

M.push(1);

BFS(1);

cout << endl;

return 0;

}

