

# 二叉树的继承遍历算法

李 军

(广西大学梧州分校 计算机科学系,广西 梧州 543002)

[摘要] 二叉树的继承遍历方法,是根据皇位继承的方法和宗族继承的习俗而提出的一种新的二叉树遍历算法,文章对其进行了讨论和分析。

[关键字] 二叉树;遍历;算法

[中图分类号]N39

[文献标识码]A

[文章编号]1009-2633(2003)02-0044-02

## 一、二叉树遍历算法的回顾

树形结构是一种重要的非线性数据结构,而二叉树是规范化的树,一般树与二叉树可以互相转换,研究并解决二叉树的存储结构及其算法,从而也就解决了一般树的存储结构及其算法的复杂性。二叉树的遍历在二叉树的算法中占有重要地位,无论是二叉树的生成、插入、查找还是删除等操作都要使用到二叉树的遍历算法。二叉树(BinaryTree)是 $n(n \geq 0)$ 个有限元素的集合,这个集合或者为空,或者有一个称作根的元素及两个不相交的、被分别称为左右子树的二叉树组成。

二叉树的遍历是指按照某种顺序访问二叉树中的每一个结点,使每个结点被访问一次且只被访问一次。

二叉树的遍历主要有:

### 1、先序遍历

先序遍历的递归算法为:若二叉树为空,遍历结束。否则,

- (1) 访问根结点;
- (2) 先序遍历根结点的左子树;
- (3) 先序遍历根结点的右子树;

### 2、中序遍历

中序遍历的递归算法为:若二叉树为空,遍历结束。否则,

- (1) 中序遍历根结点的左子树;
- (2) 访问根结点;

(3) 中序遍历根结点的右子树;

### 3、后序遍历

后序遍历的递归算法为:若二叉树为空,遍历结束。否则,

- (1) 后序遍历根结点的左子树;
- (2) 后序遍历根结点的右子树;
- (3) 访问根结点;

### 4、层序遍历

层序遍历算法,是指从二叉树的第一层(根结点)开始从上而下逐层遍历,在同一层中,则按从左到右的顺序对结点逐个访问。

## 二、二叉树的继承遍历算法

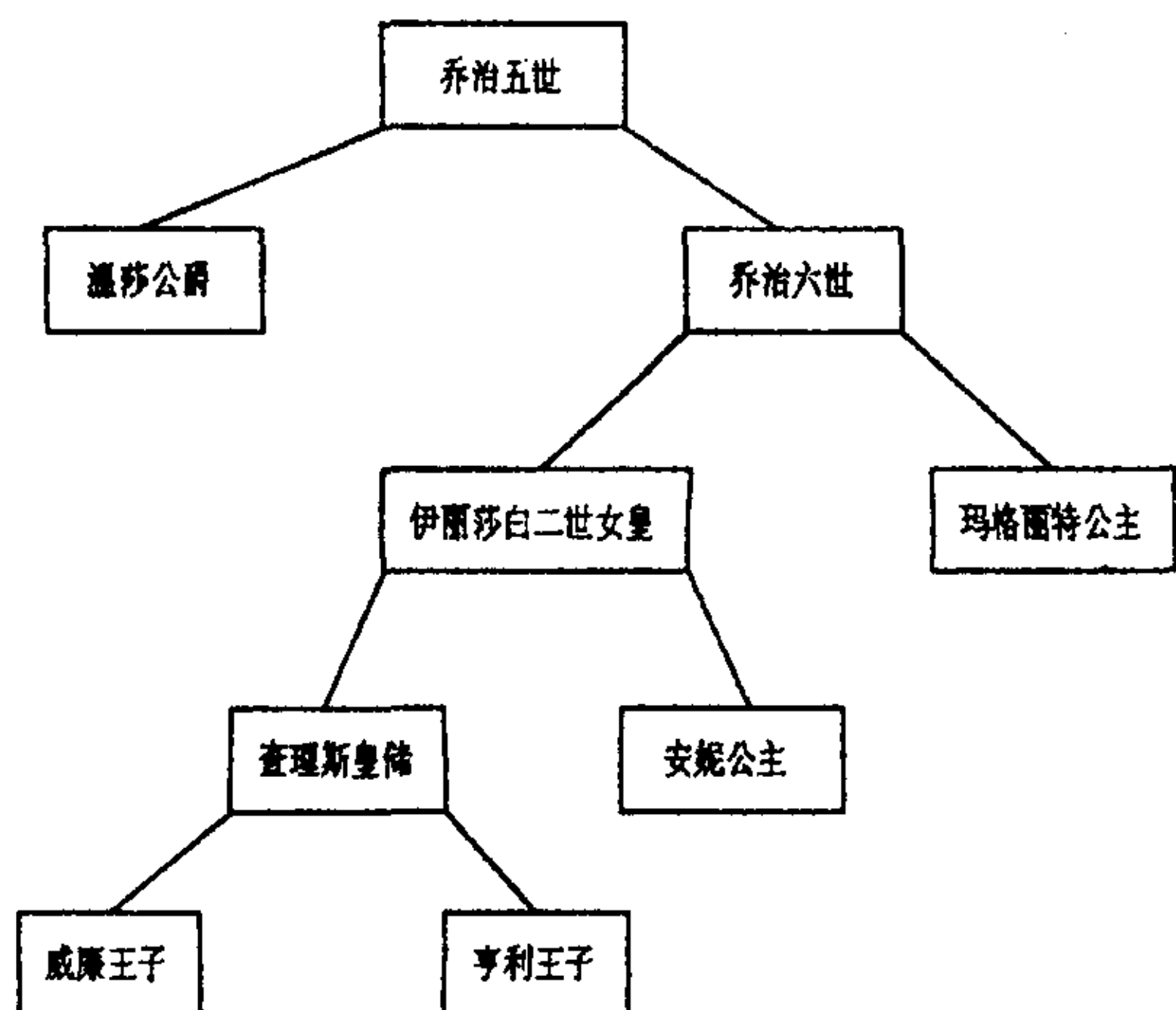
### 1、算法的提出:

二叉树的继承遍历方法,是根据家庭、宗族、王权继承的风俗、习惯、法理而提出的一种新的二叉树遍历算法。

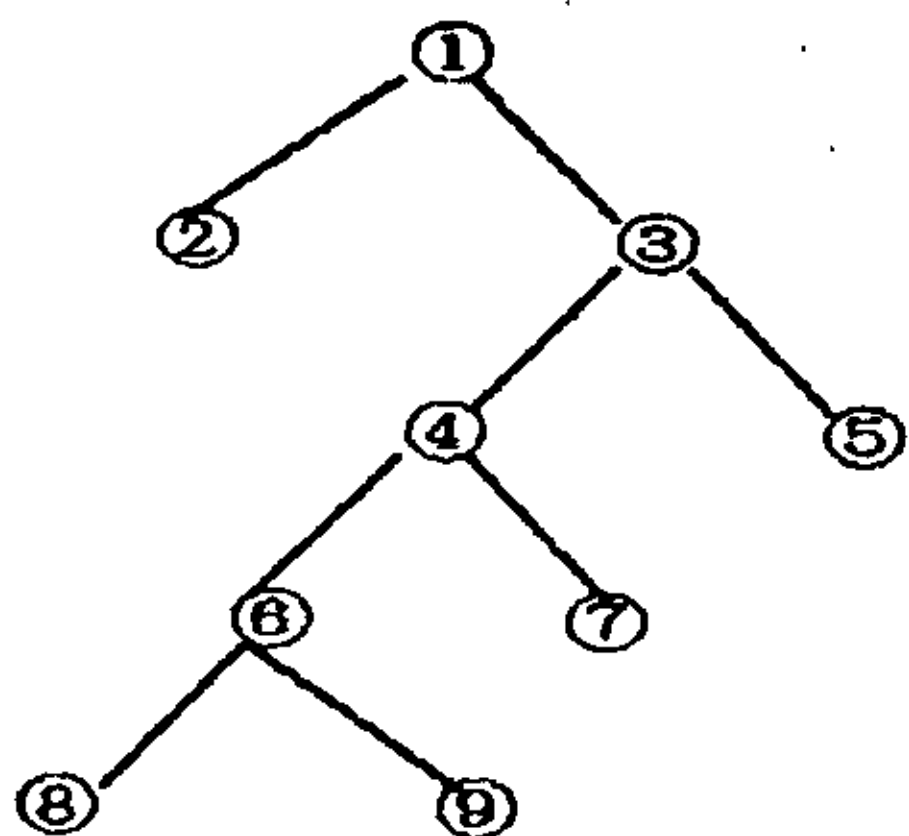
我们知道,宗族的族长、氏族的首领一般是由其族长、首领的长子继承的,只有当长子不能继承时才由次子等其他继承人继承,也就是说其继承是有一定顺序的,宗族成员的结构刚好就是树型结构,我们分析这个继承序列,就可以得出一种树的访问序列,如果这棵树恰好是二叉树,那即是二叉树的遍历序列了。

为了更清楚地看清这个问题我们用下面英国部分王室成员结构图(注:为了使其成为二叉树,故省去了一些成员,而不是完整的英国王室成员结构图,敬请原谅)进行分析:(图一)





皇位的继承次序为: 乔治五世、温莎公爵、乔治六世、伊丽莎白二世女皇、查尔斯皇储、威廉王子、亨利王子、安妮公主、玛格丽特公主。继承的顺序并不是先序遍历、中序遍历、后序遍历、层序遍历所得到的序列, 而是另外一种序列, 那么它是怎样得到的一种序列, 我们用下面的二叉树来分析: (图二)



皇位的继承顺序对应为: 1、2、3、4、6、8、9、7、5。遍历的特点是: 先逐个访问左分支结点, 左分支结点访问完后, 访问最左的结点的右孩子及其左分支结点, 重复, 直至全部结点访问完。

## 2、叉树的继承遍历算法思想

(1) 若二叉树为空, 遍历结束: 否则, 访问根结点, 把根结点压入栈, 然后使当前 结点指针指向根结点的左孩子结点。

(2) 若当前结点指针域非空, 访问该结点。把该结点压入栈, 然后使当前结点指针 指向该结点的左孩子结点。

(3) 转(2)

(4) 若栈非空, 则栈顶元素弹出栈, 然后使当前结点指针指向该结点的右孩子结点,

转(2)。

(5) 算法结束。

## 3、存储结构与算法

(1) 采用二叉链表存储结构:

```
typedef struct BiTNode{
```

```
TElem Type data; //TElemType 为结点
```

元素数据类型

```
struct BiTNode *lchild, *rchild;
```

```
}BiTNode, *BiTree;
```

(2) 二叉树的继承遍历算法:

```
void BiTreeInhericTraverse(BiTree T)
```

```
{
```

```
p=T; // 二叉树 T
```

```
if(!P) return; // 二叉树为空
```

```
InikStack(); // 初始化栈
```

```
Push(p); // p 进栈
```

```
p=p->lchild;
```

```
Loop_lo: while(1p)
```

```
{
```

```
visit(p->data); // 访问 p 进栈
```

```
push(p); // p 进栈
```

```
p = p->lchild;
```

```
}
```

```
if([StackEmpty()]) // 若栈非空。
```

```
{
```

```
p=Pop();
```

```
p=p->Loop-10
```

```
goto Loop_10;
```

```
}
```

```
return; // 遍历完成
```

```
// 算法结束
```

## 三、算法复杂度分析和结论

### 1、算法复杂度分析:

#### (1) 时间复杂度

假设二叉树有  $n$  个结点, 算法中每个结点都要入栈出栈各一次, 因此, 出入栈共要执行  $2n$  次, 时间复杂度为  $O(n)$ 。

每次沿结点左分支循环访问时, 循环次数最多为  $n$  次, 时间复杂度为  $O(n)$ 。

所以, 二叉树的继承遍历算法的时间复杂度是:  $O(n)$ 。

#### (2) 空间复杂度

最坏情况: 这时, 二叉树是左单链二叉树, 每个结点均要入栈, 且全部结点入栈后才开始出栈, 这时要求栈的长度为  $n+1$ , 所以空间复杂度为:  $O(n)$ 。

(下转第 60 页)



Garden of England(英格兰花园),用来指称 Kent(英国肯特郡);

Land of Lakes (湖泊之乡), 用来指称 “Minnesota” (美国明尼苏达州)。

2、以历史上或传说中的典故指称当今类似的人或事。

例如:

(1) His present is no more than Pandora's box

他的礼品是潘多拉的盒子, 只能给人带来祸害。(Pandora's box 潘多拉的盒子, 语出希腊神话。主神宙斯发现普罗米修斯盗取天火给人类后勃然大怒, 他命火神选一美无伦比的女子潘多拉(Pandora), 让她带着一个装满各种灾难和祸患的盒子, 降落人间, 潘多拉私自打开盒子, 里面的各种“疾病”、“悲伤”、“罪恶”、“嫉妒”等祸患全飞出来散布到世上。这里用来指称“给人带来祸害的东西”。)

(2) Every government should attend to cleaning its own Augean stables.

各国政府都应当注意扫除内部的腐败现象。(Augean stables 奥吉厄斯(Augeas)国王的牛舍, 相传他养有三千头牛, 牛舍三十年未扫, 脏臭无比。这里用来指称“极肮脏的地方”或“腐败现象”。)

换称和转喻这两种辞格十分相近, 但又有所不同。其共同点是不直接说出指代对象的名称, 而采用某种替代形式; 它们的区别在于: (1) 换称中的代体必须具有一些家喻户晓的特点或象征, 或能生动体现本体的特点, 从而唤起人们对本体的联想。例如, 当用 Uncle Tom 代换本体时, 人们就会联想到一个人性命是从、逆来顺受的特

点, 这种修辞手法就属于换称。而当用 Uncle Sam 代换本体时, 人们知道它只是代表美国, 而不会联想到其特征, 这种修辞手法就属于转喻。

(2) 换称中作为代体的专有名称通常来源于宗教、历史和文学作品以及影视文艺等, 而转喻中的专有名称则无须具有类似的起端。例如, 用 Solomion(所罗门)指称“聪明人”、Judas(犹大)指称“叛徒”、“出卖朋友的人”时, 属于换称, 而用 the Pentagon (五角大楼) 指称“美国国防部”、“Wall Street” (华尔街) 指称“美国金融界”时, 则属于转喻。(3) 换称中作为代体的专有名称可以作普通名称用。例如, 我们可以说 “Tom is a Hercules”, 意指“汤姆是个大力士”, 但我们不能说 “Tom is an Uncle Sam”。

恰当地使用换称, 能达到多种修辞效果: (1) 可以丰富语言表达; (2) 可以使语言形象具体, 幽默巧妙; (3) 可以使语言含蓄, 简洁; (4) 能表达作者的爱憎感情, 感染读者。因此, 换称常出现在文学作品、报刊杂志和日常交谈中。

西方的宗教、历史和文学是换称的重要渊源, 因此, 文化背景知识在换称的理解和应用上显得十分重要。如果不了解西方文化, 就不容易理解和应用英语中换称这一特殊修辞手段。

#### 参考文献:

[1] 冯翠华. 英语修辞大全[M]. 北京: 外语教学与研究出版社, 1995.

[2] 邹世诚. 英语修辞[M]. 台湾: 书林出版有限公司, 1993.

[3] 陆谷孙. 英汉大词[M]. 上海: 上海译文出版社, 1993.

[作者简介] 黄国都(1966-), 男, 现任教于华南师范大学增城康大学院外语系。

(责任编辑: 黄积达)

(上接第45页)

最好情况: 这时, 二叉树是右单链二叉树, 每个结点均要入栈, 但是是进栈一个结点后马上出栈, 这时要求栈的长度为 2, 所以空间复杂度为:  $O(1)$ 。

为了保证算法能够执行, 故要求栈的长度为  $n+1$ , 所以空间复杂度为:  $O(n)$ 。

#### (3) 结论

这里讨论的继承遍历算法虽然只是针对二

叉树的, 其算法思想同样适用于多叉树, 而多叉树才真正是皇位和宗族家谱继承所对应的。

#### 参考文献:

[1] 朱战立, 刘天时. 数据结构[M]. 西安: 西安交通大学出版社, 2000.

[2] 彭波. 数据结构[M]. 北京: 清华大学出版社, 2002.

[作者简介] 李军, 男, 广西大学梧州分校计算机科学系讲师, 研究方向: 数据库、网络。

(责任编辑: 仇仲谦)