

本节内容

线索二叉树
概念

王道考研/CSKAOYAN.COM

1

知识总览

线索二叉树

线索二叉树的作用

线索二叉树的存储结构

三种线索二叉树

王道考研/CSKAOYAN.COM

2

二叉树的中序遍历序列

pre → NULL

pre q → D

中序遍历序列: D G B E A F C

能否从一个指定结点开始中序遍历?

```
//中序遍历
void InOrder(BiTree T){
    if(T!=NULL){
        InOrder(T->lchild); //递归遍历左子树
        visit(T);           //访问根结点
        InOrder(T->rchild); //递归遍历右子树
    }
}
```

①如何找到指定结点p在中序遍历序列中的前驱?
 ②如何找到p的中序后继?

思路:

从根节点出发, 重新进行一次中序遍历, 指针q记录当前访问的结点, 指针pre记录上一个被访问的结点

①当q==p时, pre为前驱

②当pre==p时, q为后继

缺点: 找前驱、后继很不方便; 遍历操作必须从根开始

王道考研/CSKAOYAN.COM

3

中序线索二叉树

中序遍历序列: D G B E A F C

n个结点的二叉树, 有n+1个空链域! 可用来记录前驱、后继的信息

线索化

问题: 如何找到G的后继?

图示说明

前驱线索 (由左孩子指针充当): ·····→

后继线索 (由右孩子指针充当): ·····→

指向前驱、后继的指针称为“线索”

王道考研/CSKAOYAN.COM

4

线索二叉树的存储结构

图示说明
前驱线索（由左孩子指针充当）：
后继线索（由右孩子指针充当）：

```
//二叉树的结点（链式存储）
typedef struct BiTNode{
    ElemType data;
    struct BiTNode *lchild,*rchild;
}BiTNode,*BiTree;

//线索二叉树结点
typedef struct ThreadNode{
    ElemType data;
    struct ThreadNode *lchild,*rchild;
    int ltag,rtag; //左、右线索标志
}ThreadNode,*ThreadTree;
```

术语：二叉链表

术语：线索链表

王道考研/CSKAOYAN.COM

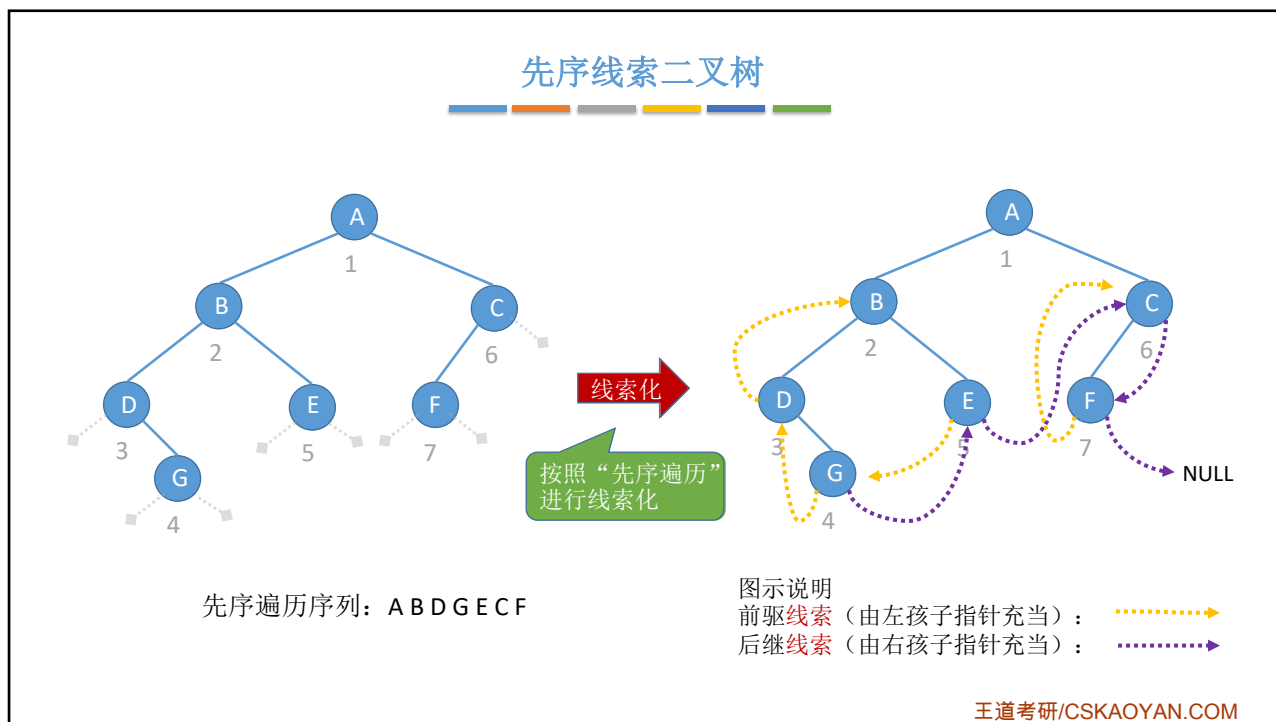
5

中序线索二叉树的存储

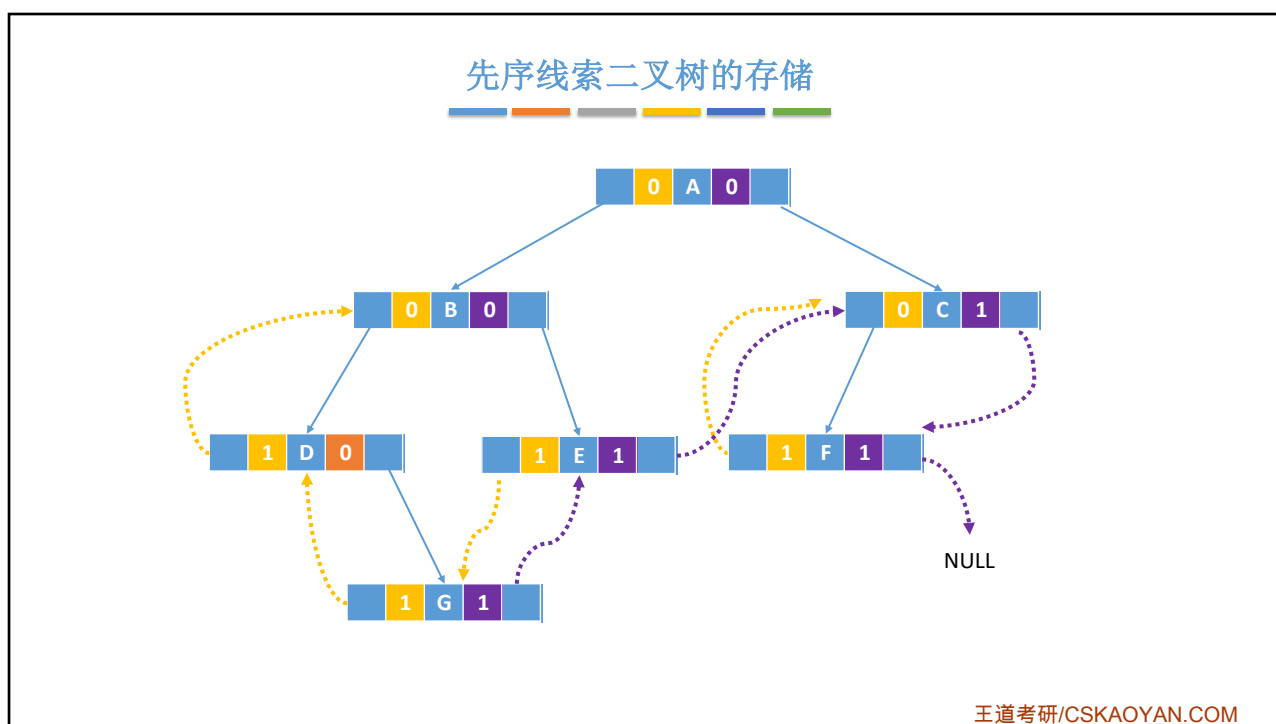
对应tag位为0时，表示指针指向其孩子
对应tag位为1时，表示指针是“线索”

王道考研/CSKAOYAN.COM

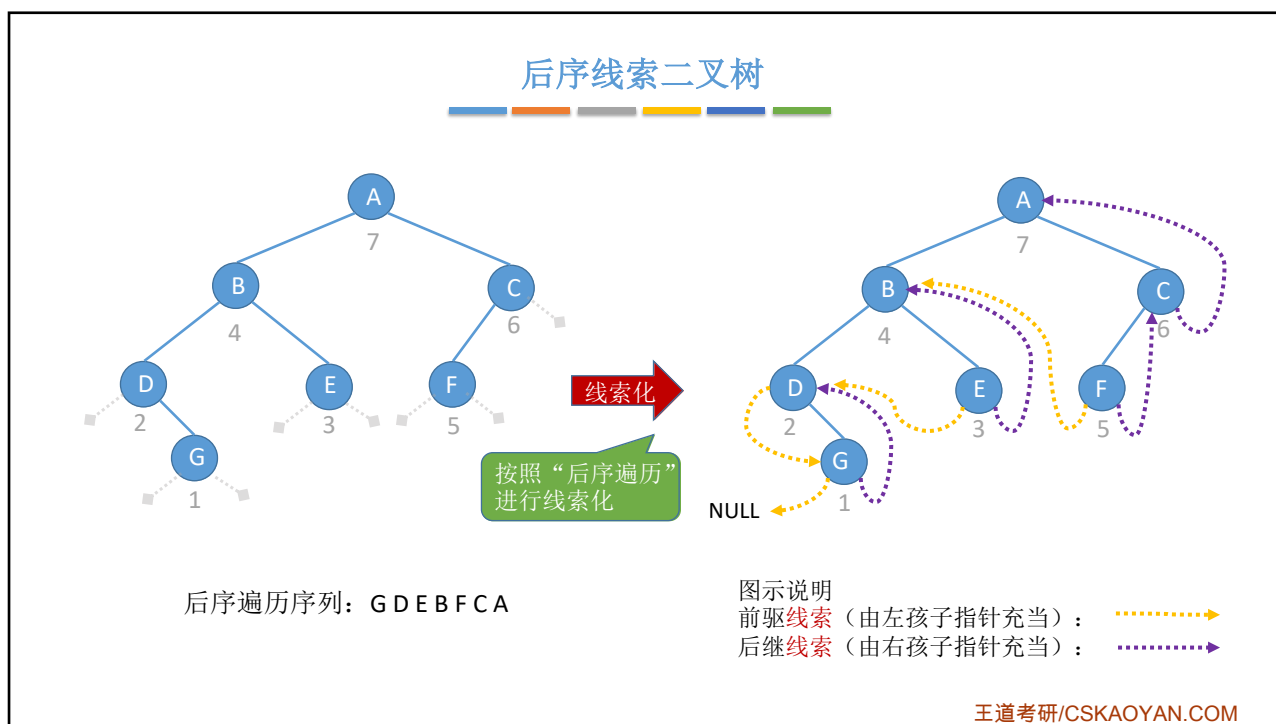
6



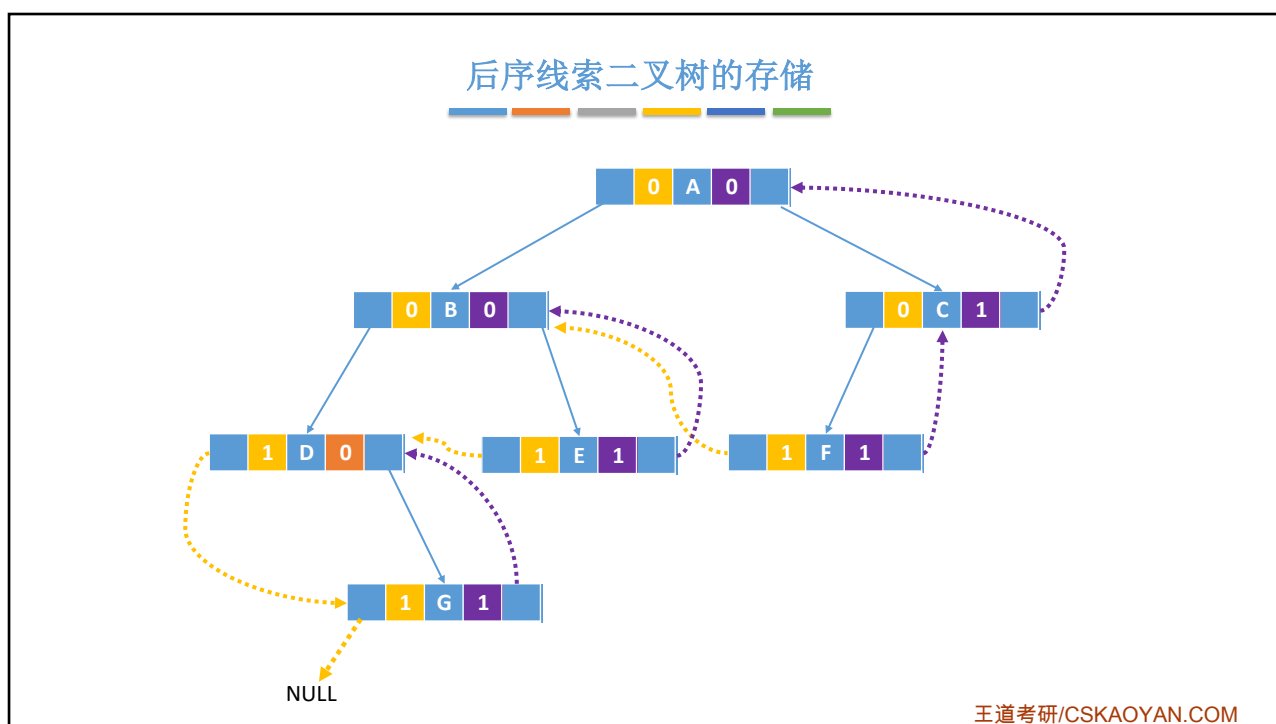
7



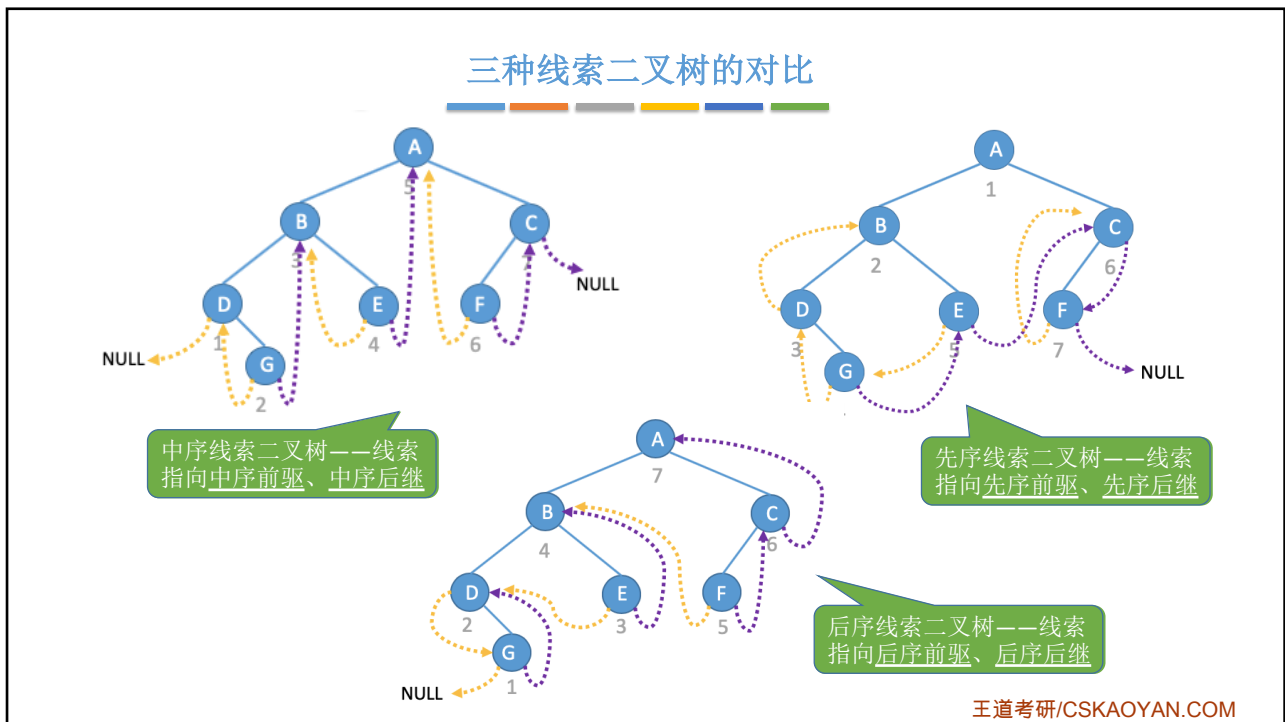
8



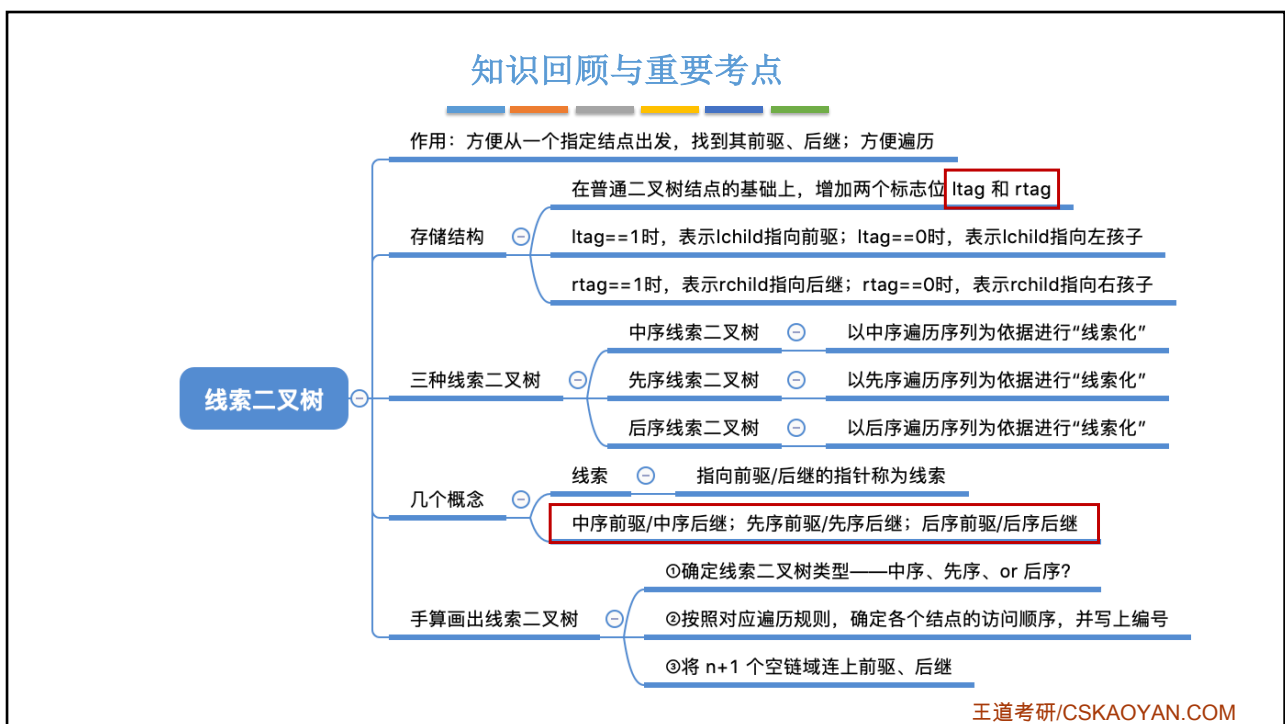
9



10



11



12