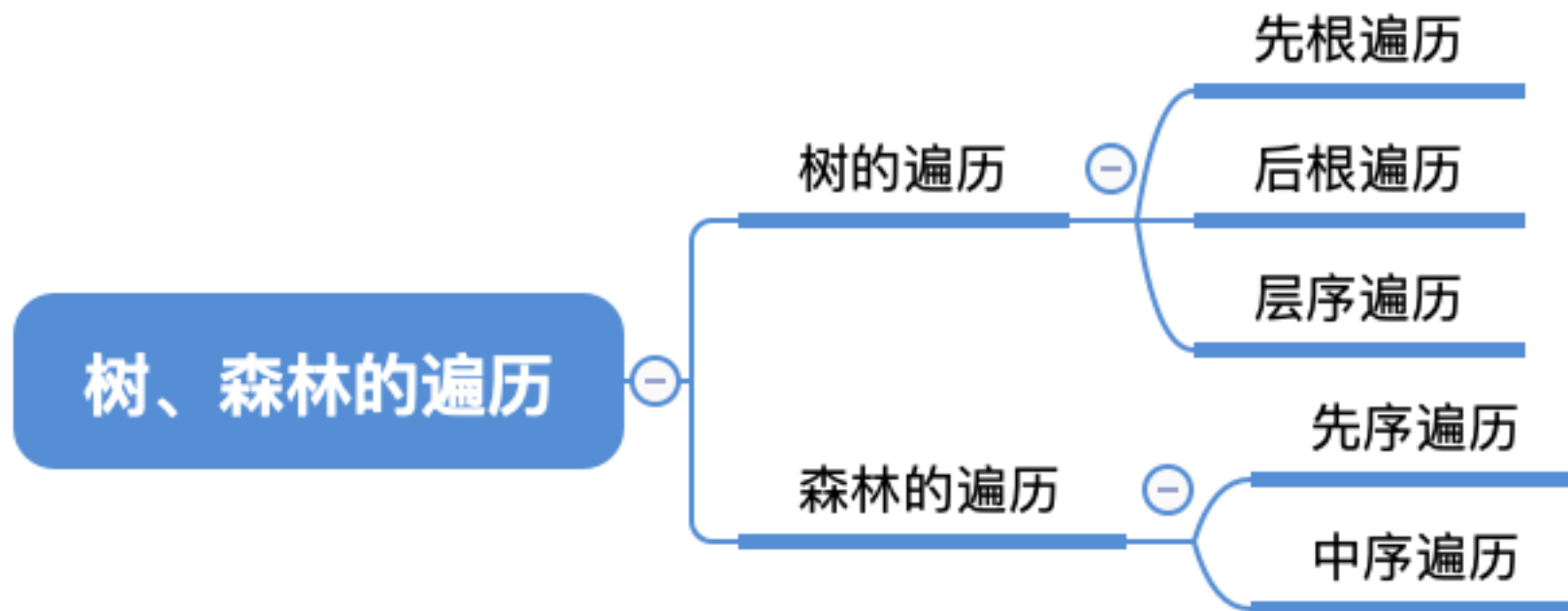


本节内容

# 树、森林 的遍历

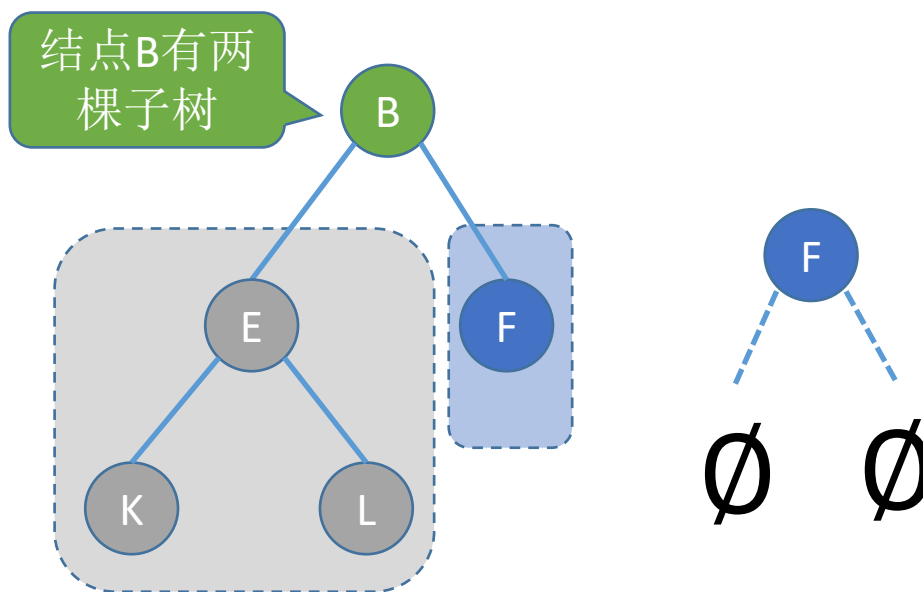
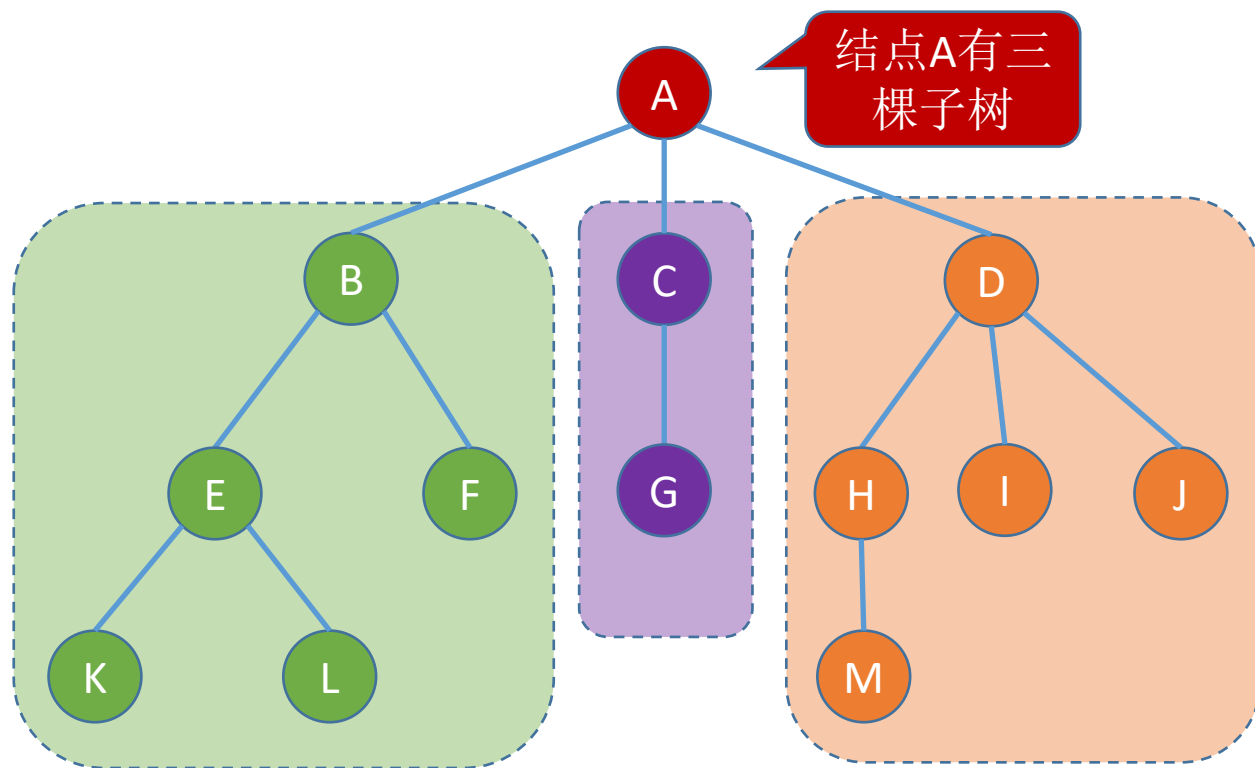
# 知识总览



# 树的逻辑结构

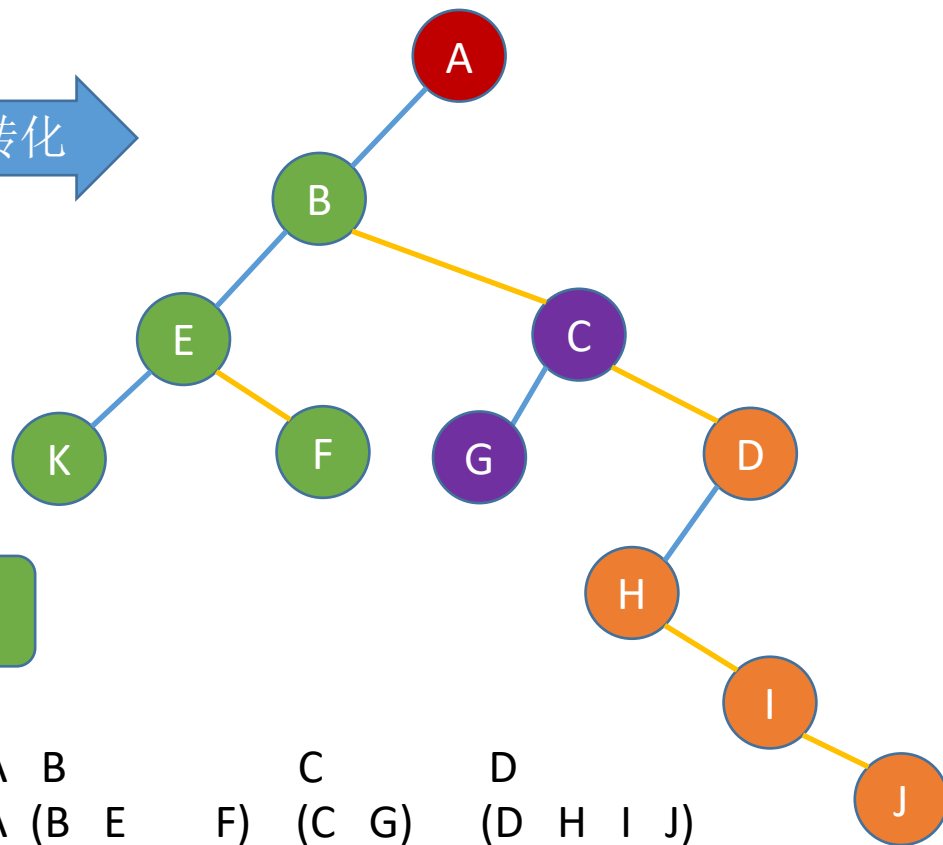
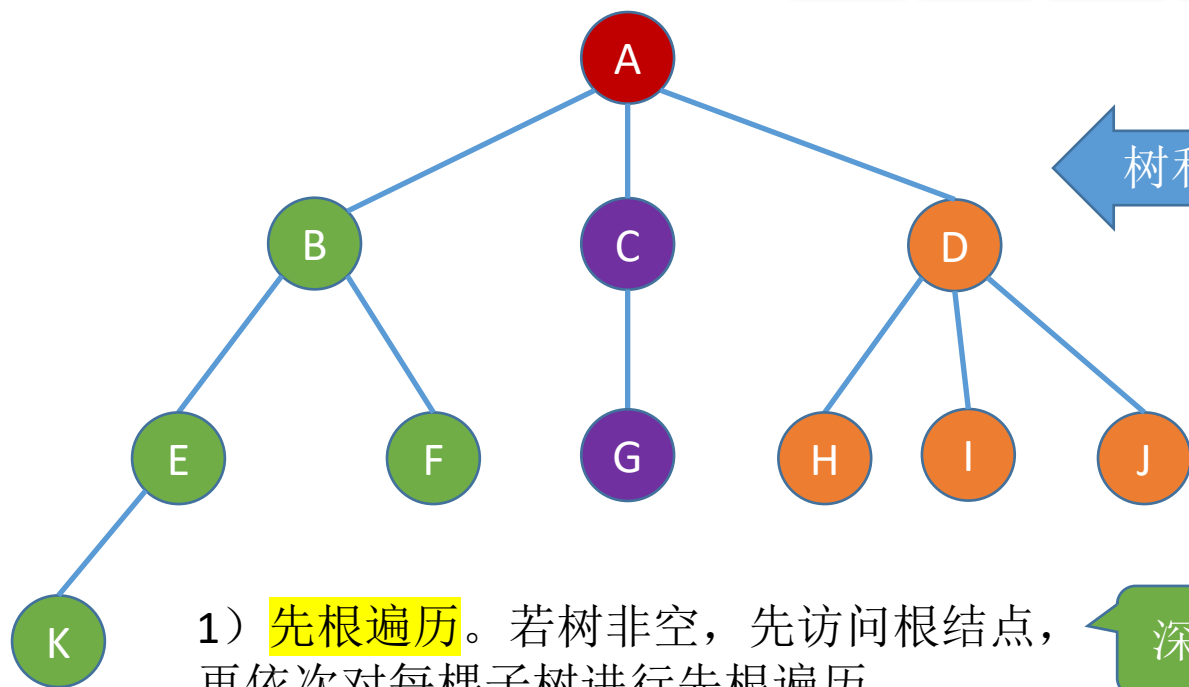
树是 $n$  ( $n \geq 0$ ) 个结点的有限集合,  $n = 0$ 时, 称为空树, 这是一种特殊情况。在任意一棵非空树中应满足:

- 1) 有且仅有一个特定的称为根的结点。
- 2) 当 $n > 1$ 时, 其余结点可分为 $m$  ( $m > 0$ ) 个互不相交的有限集合 $T_1, T_2, \dots, T_m$ , 其中每个集合本身又是一棵树, 并且称为根结点的子树。



🌲 树是一种递归定义的数据结构

# 树的先根遍历



1) **先根遍历**。若树非空，先访问根结点，再依次对每棵子树进行先根遍历。

深度优先遍历

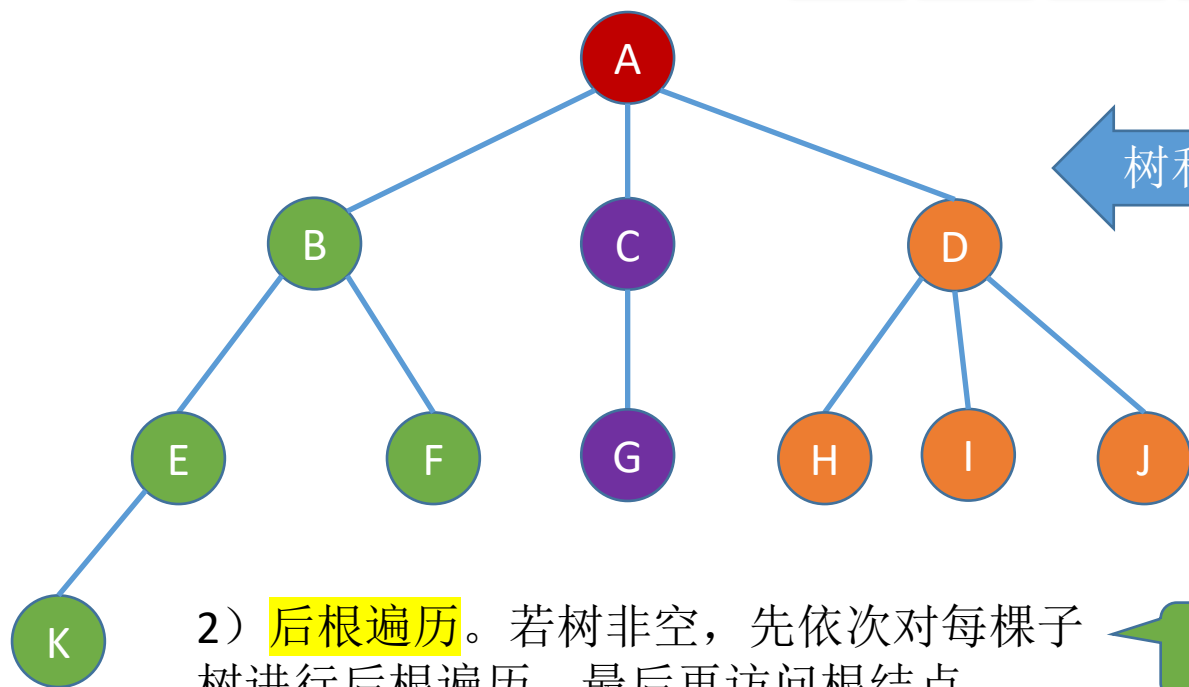
//树的先根遍历

```
void PreOrder(TreeNode *R){  
    if (R!=NULL){  
        visit(R);    //访问根节点  
        while(R还有下一个子树T)  
            PreOrder(T);    //先根遍历下一棵子树  
    }  
}
```

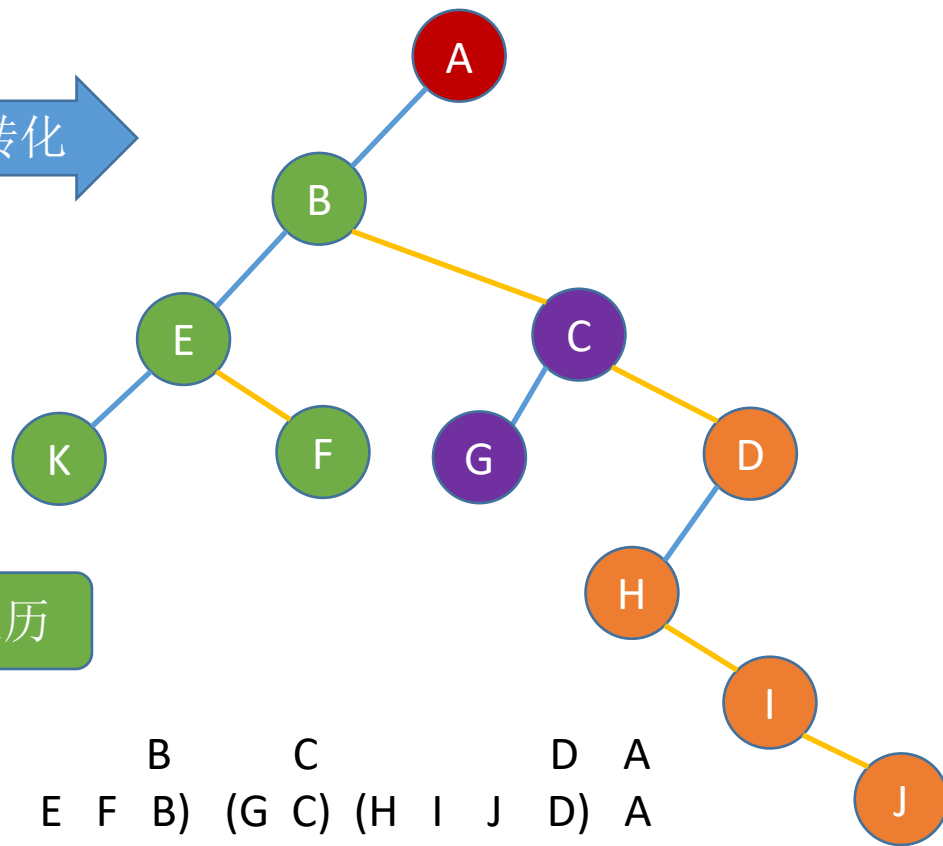
|               |       |           |   |
|---------------|-------|-----------|---|
| A             | B     | C         | D |
| A (B E F)     | (C G) | (D H I J) |   |
| A (B (E K) F) | (C G) | (D H I J) |   |

树的先根遍历序列与这棵树相应二叉树的先序序列相同。

# 树的后根遍历



树和二叉树的转化



2) 后根遍历。若树非空，先依次对每棵子树进行后根遍历，最后再访问根结点。

深度优先遍历

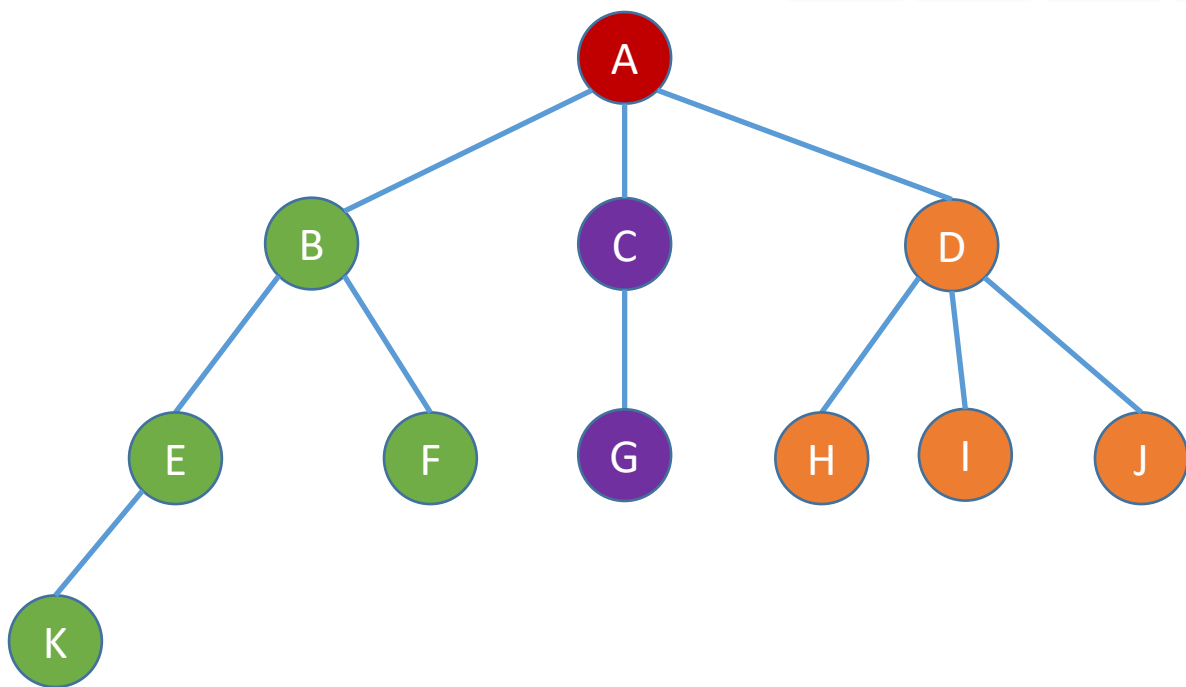
//树的后根遍历

```
void PostOrder(TreeNode *R){  
    if (R!=NULL){  
        while(R还有下一个子树T)  
            PostOrder(T); //后根遍历下一棵子树  
        visit(R); //访问根节点  
    }  
}
```

B C D A  
( E F B) (G C) (H I J D) A  
((K E) F B) (G C) (H I J D) A

树的后根遍历序列与这棵树相应二叉树的中序序列相同。

# 树的层次遍历



广度优先遍历

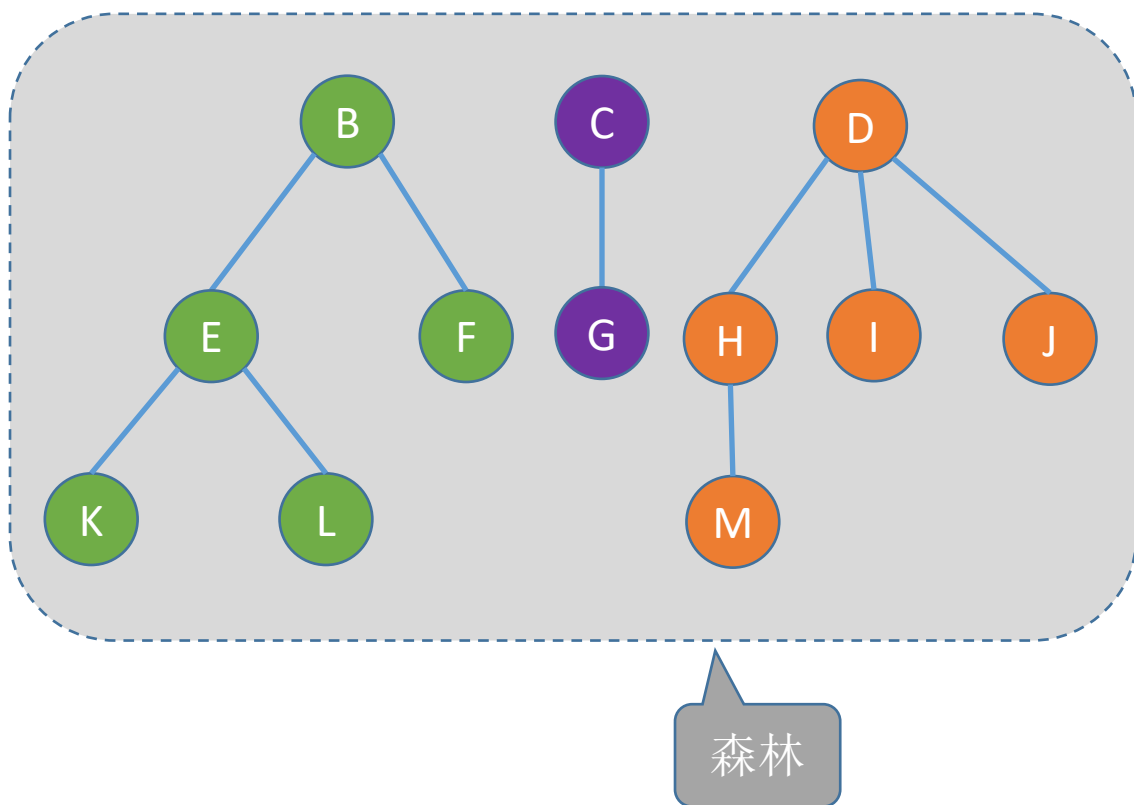
## 3) 层次遍历 (用队列实现)

- ①若树非空，则根节点入队
- ②若队列非空，队头元素出队并访问，同时将该元素的孩子依次入队
- ③重复②直到队列为空



# 森林的先序遍历

森林。森林是 $m$  ( $m \geq 0$ ) 棵互不相交的树的集合。每棵树去掉根节点后，其各个子树又组成森林。



## 1) 先序遍历森林。

若森林为非空，则按如下规则进行遍历：

访问森林中第一棵树的根结点。

先序遍历第一棵树中根结点的子树森林。

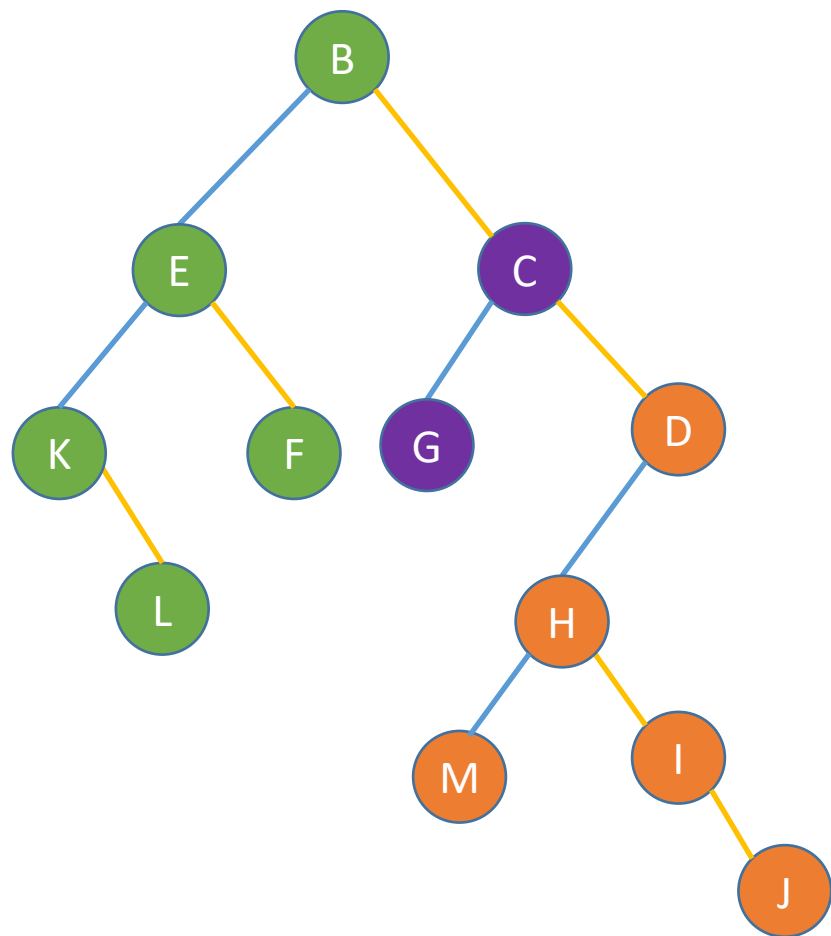
先序遍历除去第一棵树之后剩余的树构成的森林。

| B  |    |   |    |    | C  |    | D  |    |    |      |
|----|----|---|----|----|----|----|----|----|----|------|
| (B | E  |   |    | F) | (C | G) | (D | H  |    | I J) |
| (B | (E | K | L) | F) | (C | G) | (D | (H | M) | I J) |

效果等同于依次对各个树进行先根遍历

# 森林的先序遍历

森林。森林是 $m$  ( $m \geq 0$ ) 棵互不相交的树的集合。每棵树去掉根节点后，其各个子树又组成森林。



## 1) 先序遍历森林。

若森林为非空，则按如下规则进行遍历：

访问森林中第一棵树的根结点。

先序遍历第一棵树中根结点的子树森林。

先序遍历除去第一棵树之后剩余的树构成的森林。

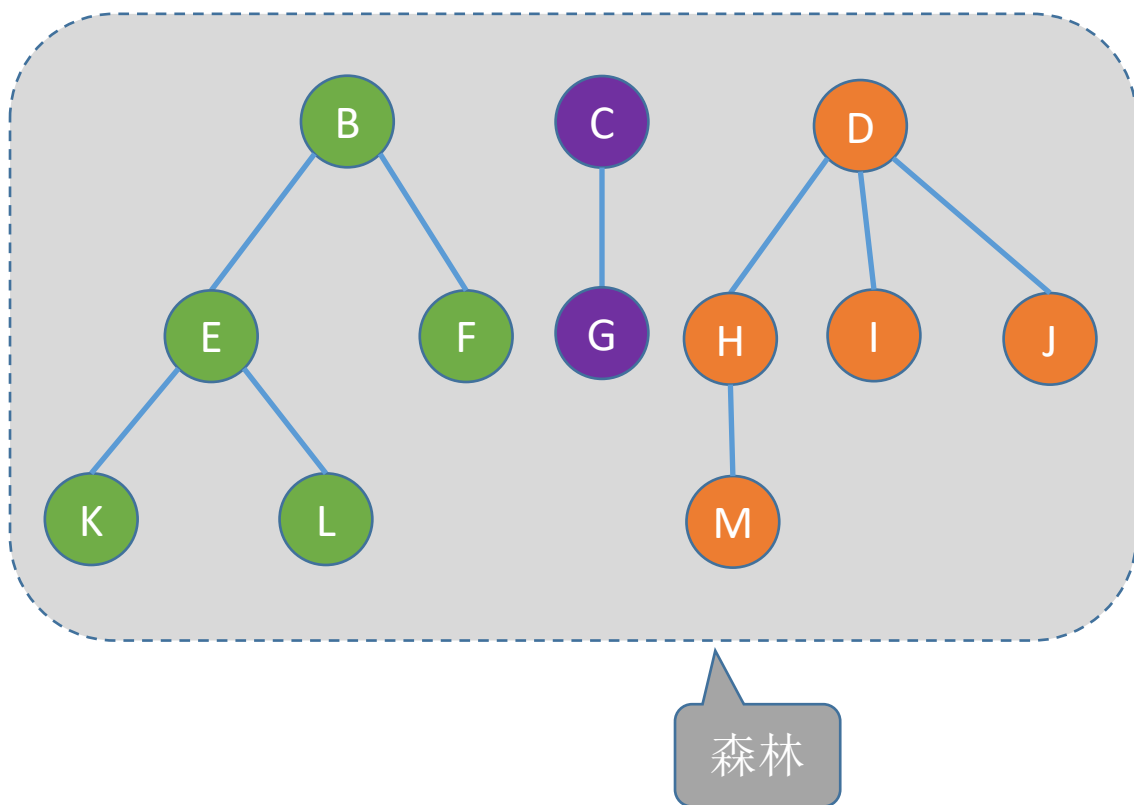
| B  |    |   |    |    |  | C  |    |    | D  |    |   |    |  |  |
|----|----|---|----|----|--|----|----|----|----|----|---|----|--|--|
| (B | E  |   |    | F) |  | (C | G) | (D | H  |    | I | J) |  |  |
| (B | (E | K | L) | F) |  | (C | G) | (D | (H | M) | I | J) |  |  |

效果等同于依次对二叉树的先序遍历



# 森林的中序遍历

森林。森林是 $m$  ( $m \geq 0$ ) 棵互不相交的树的集合。每棵树去掉根节点后，其各个子树又组成森林。



## 2) 中序遍历森林。

若森林为非空，则按如下规则进行遍历：

中序遍历森林中第一棵树的根结点的子树森林。

访问第一棵树的根结点。

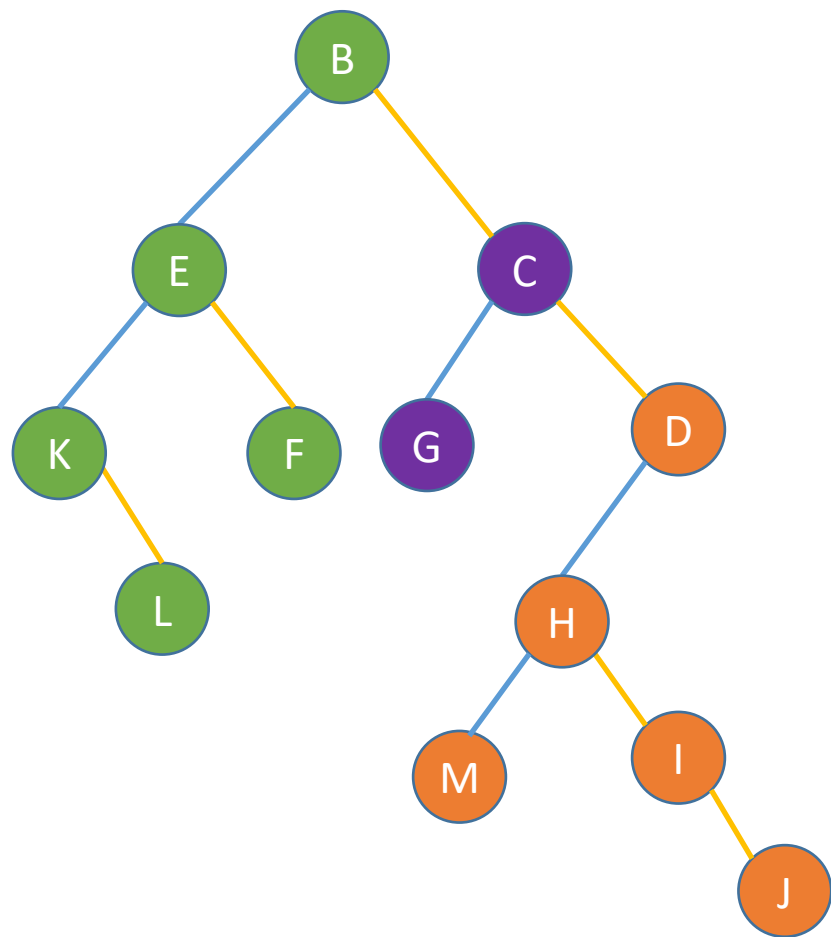
中序遍历除去第一棵树之后剩余的树构成的森林。

```
      B      C      D
(      E  F  B) (G  C) (      H  I  J  D)
(( K  L  E)  F  B) (G  C) (( M  H)  I  J  D)
```

效果等同于依次对各个树进行后根遍历

# 森林的中序遍历

森林。森林是 $m$  ( $m \geq 0$ ) 棵互不相交的树的集合。每棵树去掉根节点后，其各个子树又组成森林。



## 2) 中序遍历森林。

若森林为非空，则按如下规则进行遍历：

中序遍历森林中第一棵树的根结点的子树森林。

访问第一棵树的根结点。

中序遍历除去第一棵树之后剩余的树构成的森林。

```

          B      C      D
        (      E  F  B) (G  C) (      H  I  J  D)
      (( K  L  E)  F  B) (G  C) (( M  H)  I  J  D)
```

效果等同于依次对二叉树的中序遍历

# 知识回顾与重要考点



| 树    | 森林   | 二叉树  |
|------|------|------|
| 先根遍历 | 先序遍历 | 先序遍历 |
| 后根遍历 | 中序遍历 | 中序遍历 |