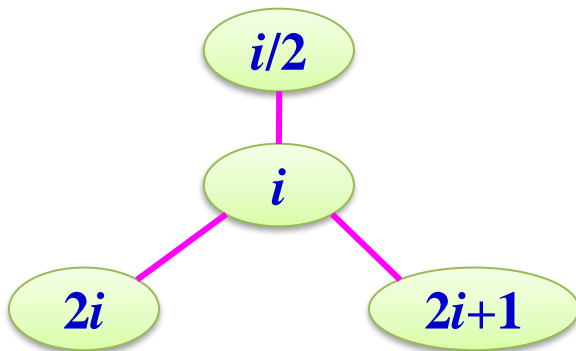


7.3 二叉树的存储结构

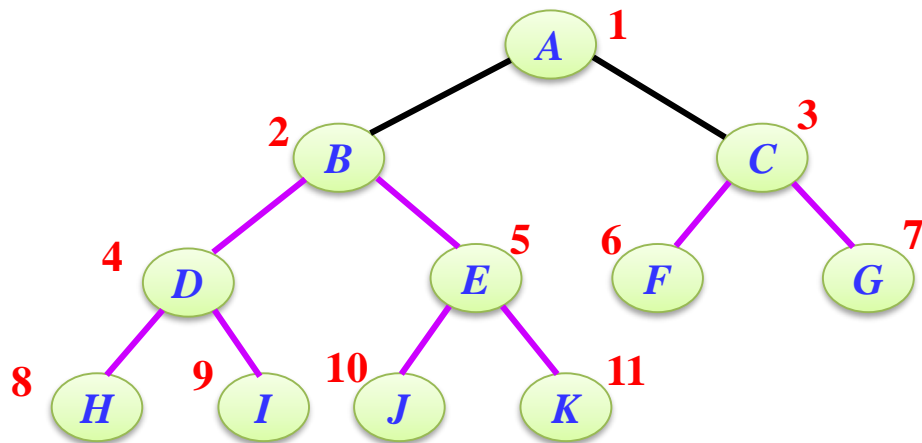
7.3.1 二叉树的顺序存储结构

回顾二叉树的性质4，完全二叉树节点按层序编号：





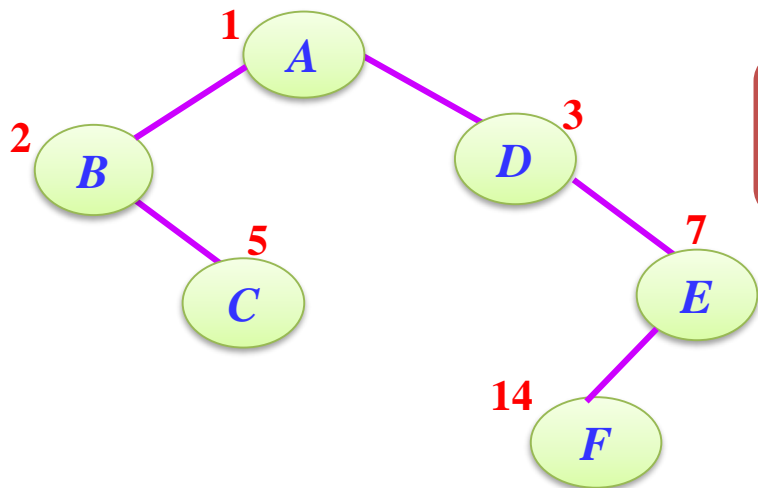
完全二叉树的顺序存储结构



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	B	C	D	E	F	G	H	I	J	K	#	#	#	#

顺序存储结构（不用下标为0的元素）

● 非完全二叉树的顺序存储结构



一般的二叉树先用空节点补全成为完全二叉树，然后对节点编号

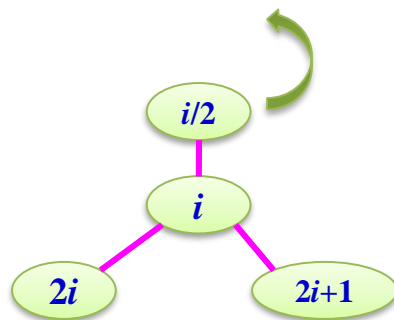
1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	B	D	#	C	#	E	#	#	#	#	#	#	F

```
typedef ElemType SqBTree[MaxSize];  
SqBTree bt="#ABD#C#E#####F";
```

} 用一个数组存储

二叉树顺序存储结构的特点：

- 对于完全二叉树来说，其顺序存储是十分合适的。
- 对于一般的二叉树，特别是对于那些单分支节点较多的二叉树来说是很不合适的，因为可能只有少数存储单元被利用，特别是对退化的二叉树（即每个分支节点都是单分支的），空间浪费更是惊人。
- 在顺序存储结构中，找一个节点的双亲和孩子都很容易。




7.3.2 二叉树的链式存储结构

借鉴树的孩子链存储结构 \Rightarrow 二叉树的链式存储结构。

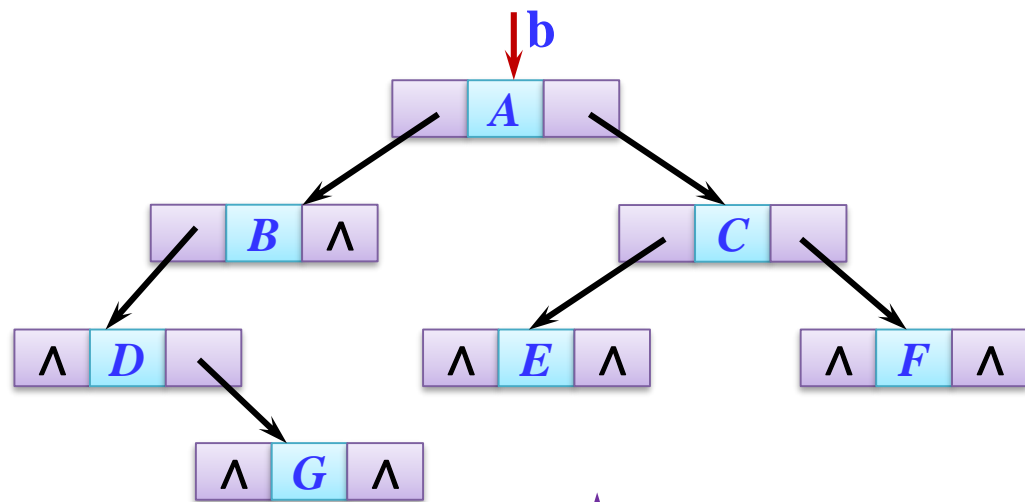
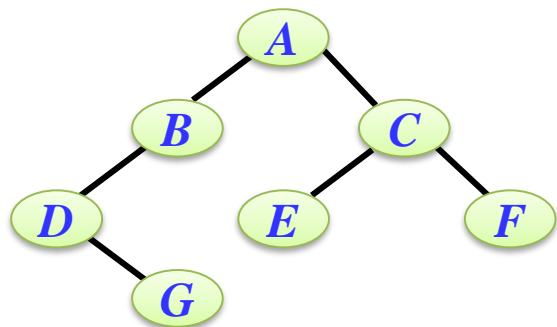
在二叉树的链式存储中，节点的类型定义如下：

```
typedef struct node
{
    ElemType data;
    struct node *lchild, *rchild;
} BTNode;
```



指向的都是二叉树：递归性

二叉链存储结构演示



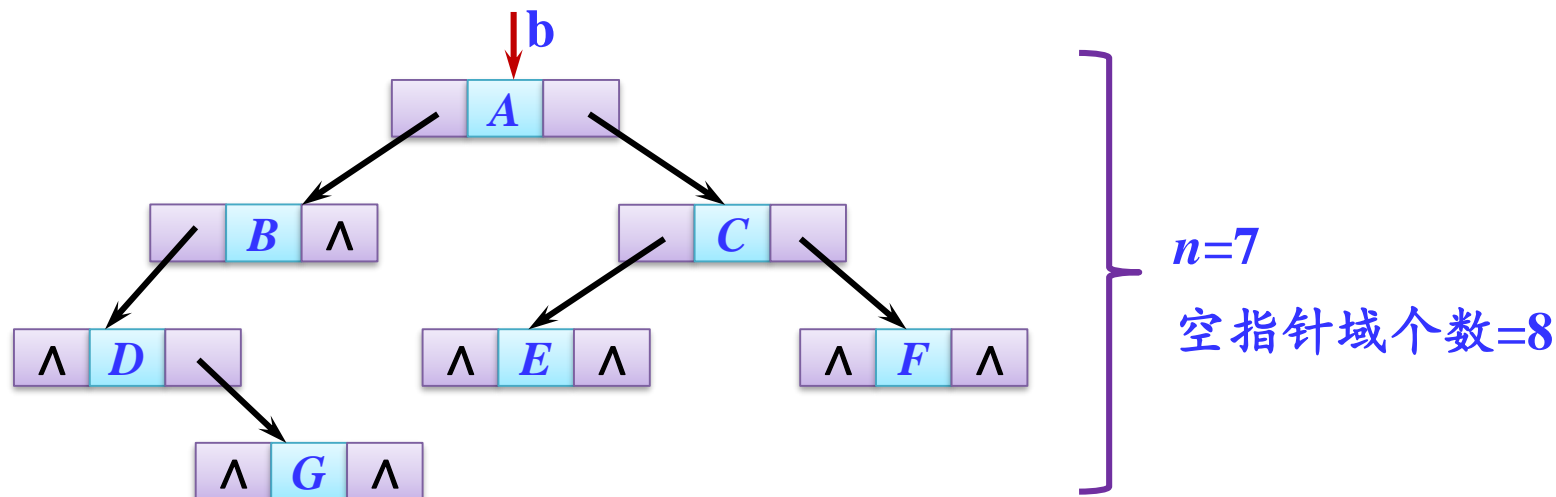
↑
二叉链

二叉链存储结构的特点：

- 除了指针外，二叉链比较节省存储空间。占用的存储空间与树形没有关系，只与树中节点个数有关。
- 在二叉链中，找一个节点的孩子很容易，但找其双亲不方便。

 一颗树采用孩子兄弟链存储结构表示 \Rightarrow 二叉链

在二叉链中，空指针的个数？



- n 个节点 $\Rightarrow 2n$ 个指针域
- 分支数为 $n-1 \Rightarrow$ 非空指针域有 $n-1$ 个
- 空指针域个数 $= 2n - (n-1) = n+1$

思考题

二叉树的顺序存储结构和二叉链存储结构各有什么优缺点？

——本讲完——