

补充内容

1. K叉树
2. 堆与优先队列

K叉树

二叉树的扩展

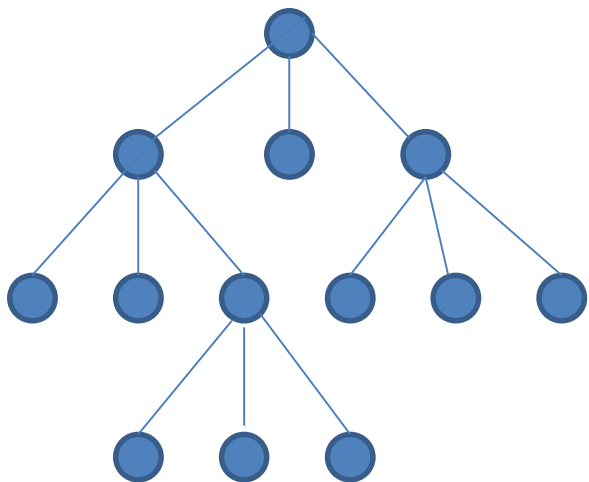
定义: 满足下列性质的有限结点集

- (1) 可以是空 (空的K叉树)
- (2) 由特殊称谓root的结点, 以及K颗互不相交的K叉树构成。

除root外, 剩余结点分成 T_0 、 T_1 、.....、 T_{k-1}
($k \geq 1$) 个子集, 每个子集都是K叉树。

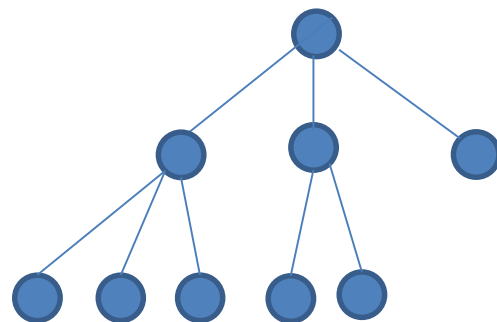
二叉树的若干性质可以用于K叉树

深度 (高度) 与结点数目关系;
满K叉树、完全K叉树;



满3叉树

只有度为 k 和度为0的结点



完全3叉树

结点编号，顺序存储

结点之间的逻辑关系隐含在编号中

给定编号 i ，其子女编号

给定编号 i ，其父结点编号

最常用的：4叉树，8叉树

堆与优先队列

1. 堆的定义:

- (1) 小根堆
- (2) 大根堆

逻辑上：堆是完全二叉树（结点编号关系）

物理上：堆是满足某种性质的元素序列

小根堆，最小值在root

大根堆，最大值在root

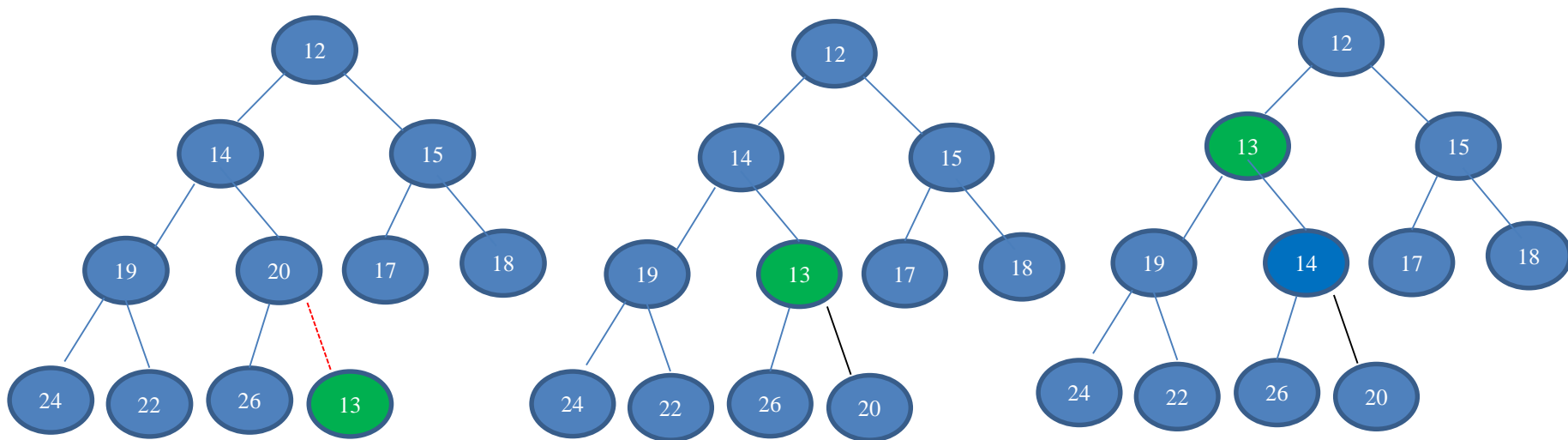
2. 运算：插入、删除、构建

3. 堆的插入:

插入后不破坏堆的性质。

具体实现（小根堆）：

- (1) 先插入在最尾部；
- (2) 沿祖先路径调整，直到满足堆性质。



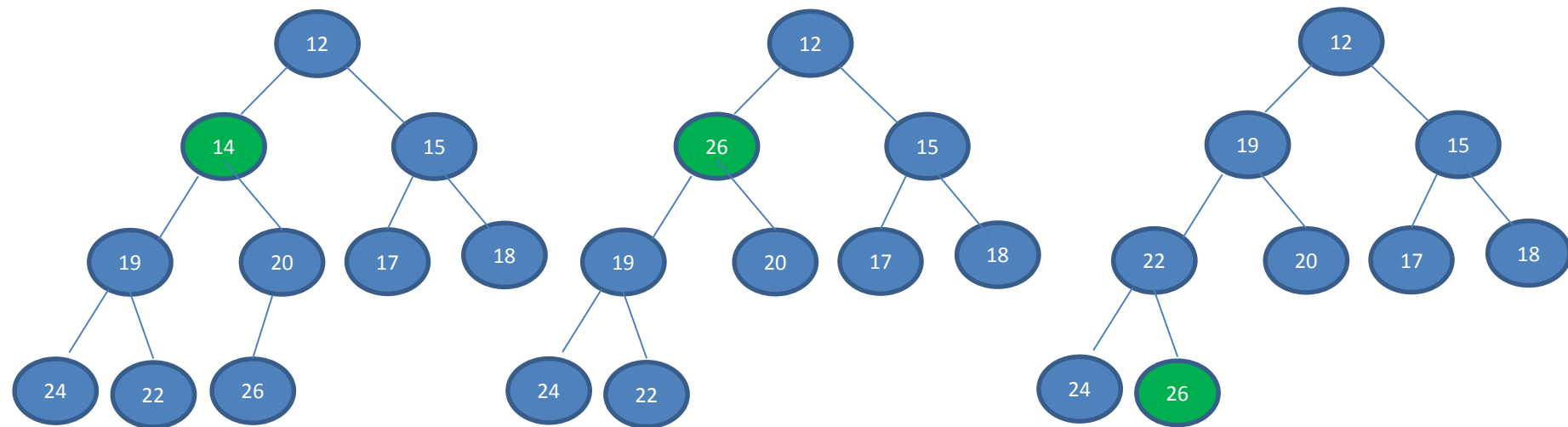
筛选法：祖先路径上向上调整，满足小根堆性质时结束，最多到根。

4. 堆的删除:

删除后不破坏堆的性质。

具体实现（小根堆）：

- (1) 先用最末尾元素替换被删元素;
- (2) 从删除位置进行筛选, 直到满足堆性质。



筛选法：子女路径上往下调整，满足小根堆性质时结束，最多到树叶。

5. 堆的构建:

- (1) 数据存储在一个顺序表（数组）中
- (2) 按照“筛选法”构造初始堆（堆排序中第一个过程）；

另外，可以使用堆的插入运算，一个一个插入完成。但麻烦，不如前面的初始堆构建简单！

注意：“ $n/2, \dots, n-1$ ”已经满足性质，不需要筛选。从“ $n/2-1, \dots, 0$ ”，需要筛选。

时间复杂度分析： $O(n)$ ，见堆排序的第一部分。
大根堆类似，性质不同，需要自己理解。

6. 优先队列:

一种数据结构，0或多个元素集合。

运算：查找，插入，删除

插入、删除后仍然满足堆性质。

与一般队列区别：非队头、队尾，而是“最小”、“最大”问题。

1) 查找、删除最小元素，插入运算，对应“小根堆”

2) 查找、删除最大元素，插入运算，对应“大根堆”

其它问题

1. 编号问题

0开始或1开始，题目将明确；

2. 树、二叉树的深度（高度）问题

如题目没有明确，按照我们课堂讲的写。

仔细审题，看清题义！！！！

（考试：就是题目看似简单，但需要琢磨）