

10. 优先级队列

(b1) 完全二叉堆：结构

逊问曰：“何人将乱石作堆？
如何乱石堆中有杀气冲起？”

邓俊辉

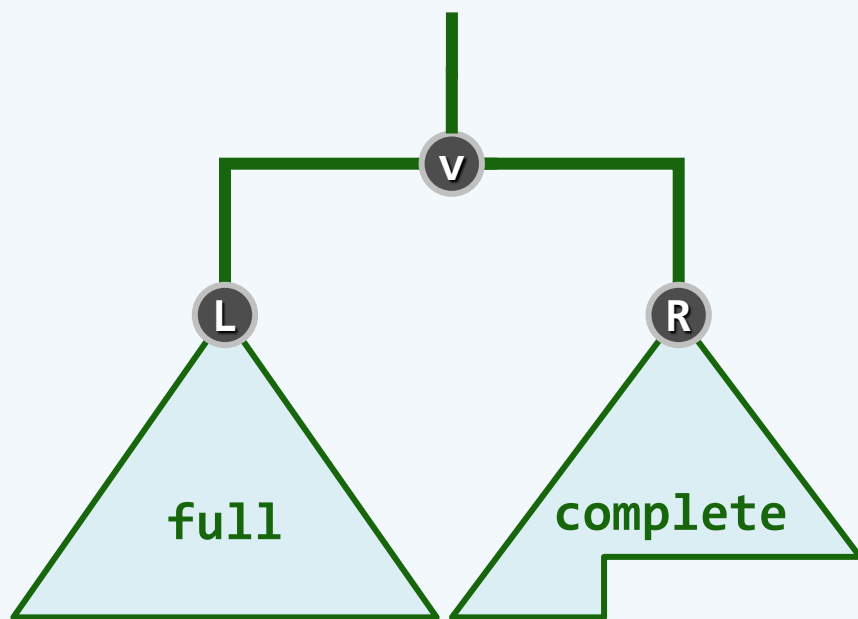
deng@tsinghua.edu.cn

完全二叉树

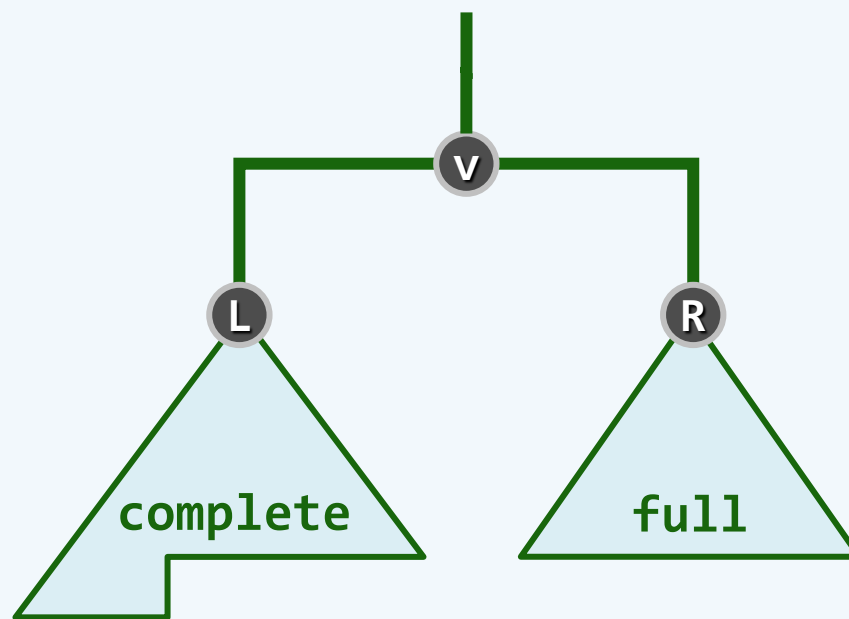
❖ Complete Binary Tree

平衡因子处处非负的AVL

而且...



若 $bf(v) = 0$ ，则 $lc(v)$ 满



若 $bf(v) = 1$ ，则 $rc(v)$ 满

结构性

❖ 逻辑上，等同于完全二叉树

物理上，直接借助向量实现

❖ 逻辑节点与物理元素

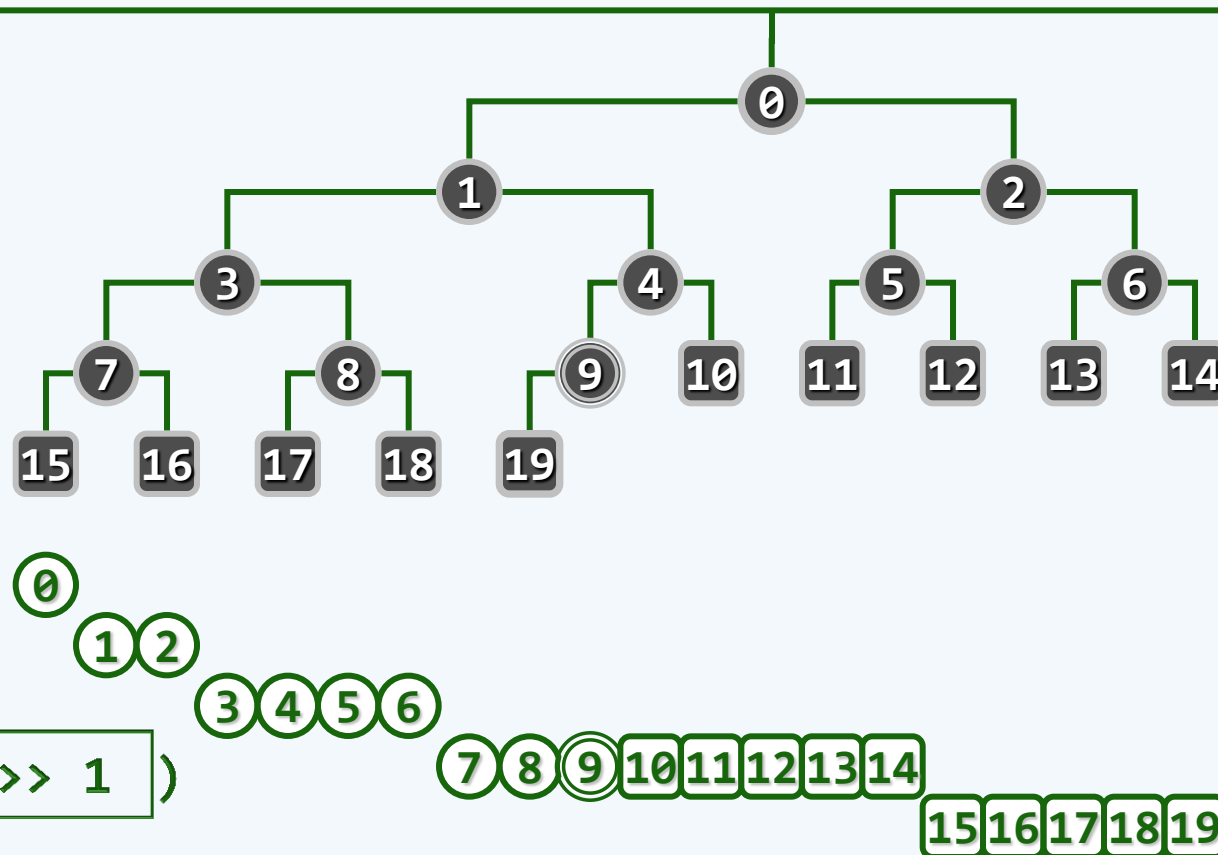
依层次遍历次序彼此对应

```
#define Parent( i ) ( ( i - 1 ) >> 1 )
```

```
#define LChild( i ) ( 1 + ( ( i ) << 1 ) ) //奇数
```

```
#define RChild( i ) ( ( 1 + ( i ) ) << 1 ) //偶数
```

❖ 共 n 个节点时，内部节点的最大秩 = $\lfloor (n - 2)/2 \rfloor = \lceil (n - 3)/2 \rceil$ //比如，这里的[9]



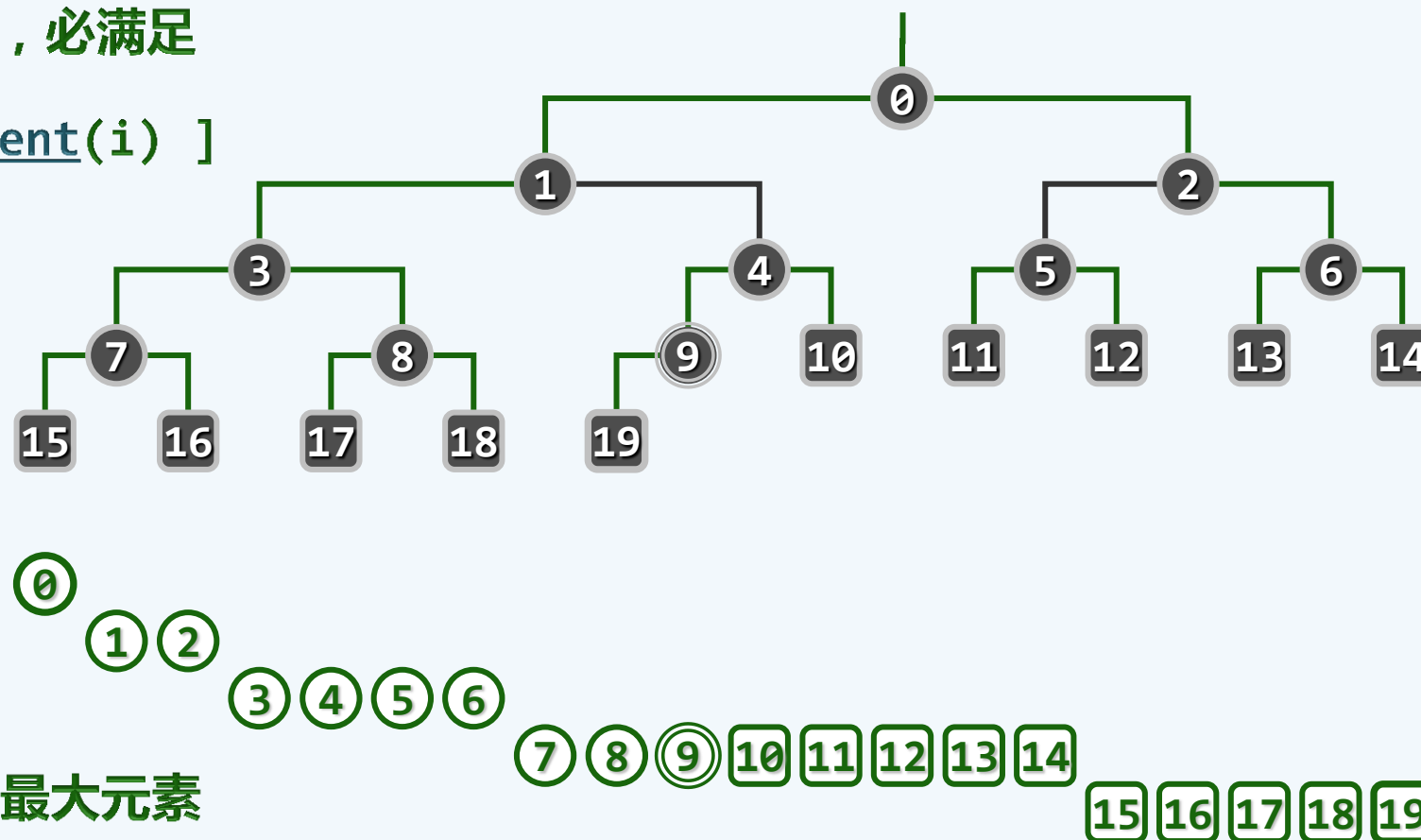
PQ_ComplHeap = PQ + Vector

```
❖ template <typename T> class PQ_ComplHeap : public PQ<T>, public Vector<T> {  
    protected: Rank percolateDown( Rank n, Rank i ); //下滤  
                Rank percolateUp( Rank i ); //上滤  
                void heapify( Rank n ); //Floyd建堆算法  
    public:     PQ_ComplHeap( T* A, Rank n ) //批量构造  
                { copyFrom( A, 0, n ); heapify( n ); }  
                void insert( T ); //按照比较器确定的优先级次序，插入词条  
                T getMax(); //读取优先级最高的词条  
                T delMax(); //删除优先级最高的词条  
};
```

堆序性

❖ 数值上, 只要 $0 < i$, 必满足

$$H[i] \leq H[\text{Parent}(i)]$$



❖ 故 $H[0]$ 即是全局最大元素

```
template <typename T> T
```

```
PQ_ComplHeap<T>::getMax() { return _elem[0]; }
```