

12-B1

优先级队列

完全二叉堆：结构

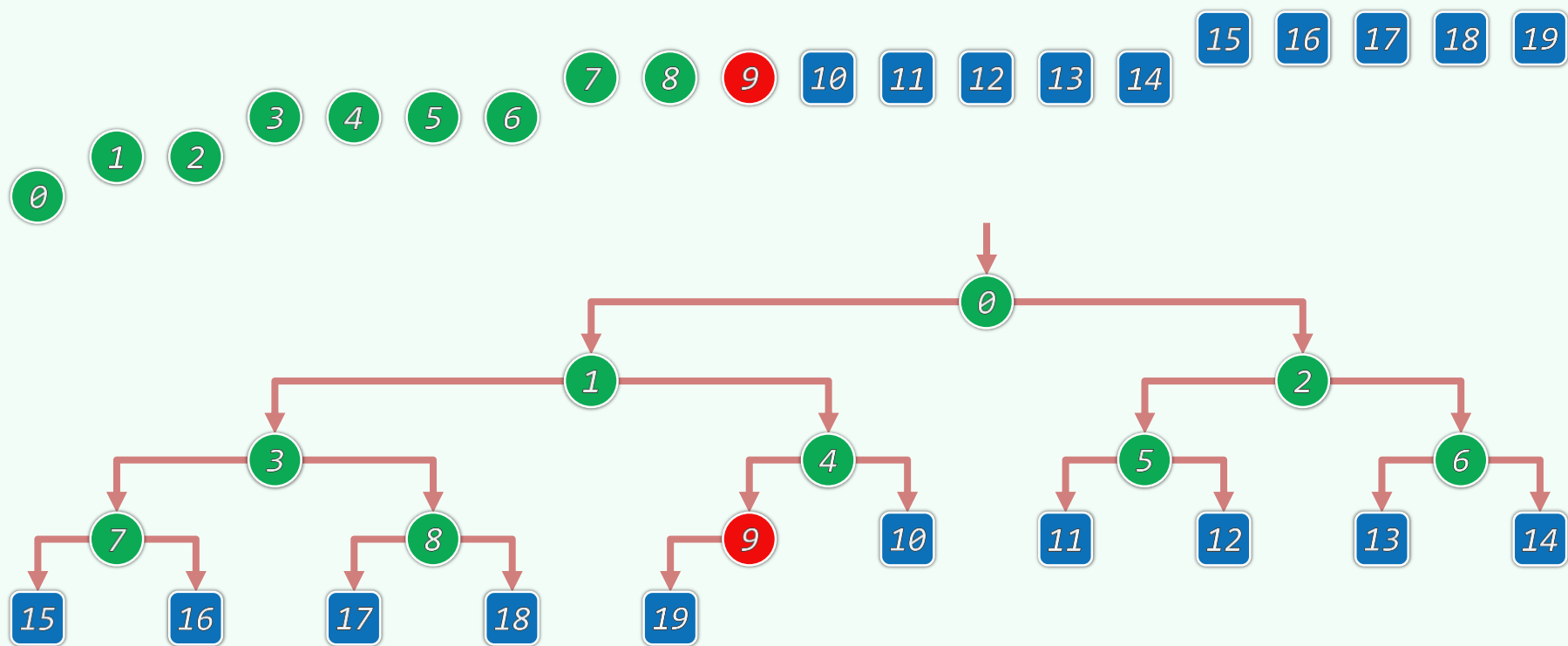
邓俊辉

deng@tsinghua.edu.cn

何人将乱石作堆？如何乱石堆中有杀气冲起？

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

```
#define Parent(i)  ( ((i) - 1) >> 1 )    ❖ 内部节点的最大秩 =  $\lfloor \frac{n-2}{2} \rfloor = \lceil \frac{n-3}{2} \rceil$   
#define LChild(i)  ( 1 + ((i) << 1) )  
#define RChild(i)  ( (1 + (i)) << 1 )
```



PQ_ComplHeap = PQ + Vector

- ❖

```
template <typename T> struct PQ_ComplHeap : public PQ<T>, public Vector<T> {  
    PQ_ComplHeap( T* A, Rank n ) { copyFrom( A, 0, n ); heapify( _elem, n ); }  
    void insert( T ); T delMax(); T & getMax();  
};
```
- ❖

```
template <typename T> Rank percolateDown( T* A, Rank n, Rank i ); //下滤
```

```
template <typename T> Rank percolateUp( T* A, Rank i ); //上滤
```
- ❖

```
template <typename T> void heapify( T* A, Rank n ); //Floyd建堆算法
```

堆序性：处处满足 $H[i] \leq H[\text{Parent}(i)]$

❖ 故 $H[0]$ 即是全局最大者

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

