

## Heap

⇒ 完全二元樹，滿足堆序性質

⇒ 形狀性質：  
“每層都是滿的，除了最後一層”  
“最後一層即為由左至右填滿，不能有空缺”

⇒ 優點：heap 可以高效率的放在 array 中，造成記憶浪費

⇒ 應用：“優先佇列 priority queue” “排程 scheduling” “堆積排序 heap sort

⇒ “key” 代表優先權 (key 大, priority 高)

堆序性質 Heap-Order Property

1. Max heap 最大堆積 { 父節點的值  $\geq$  子節點的值

根有最大值

2. Min heap 最小堆積 { 父節點的值  $\leq$  子節點的值

根有最小值

3. 應用：實現優先佇列 priority queues / 堆積排序 heap sort

時間複雜度

插入 insert(x) ⇒ sift-up 向上調整，故本層最左邊，父子比、swap ⇒  $O(\log n)$

取出 extract root() ⇒ sift-down 向下調整，故本層最左邊，父子比、swap ⇒  $O(\log n)$

⇒ 樹的高度  $\log n$

查看 peek\_root() ⇒ 讀取最大/小 ⇒  $O(1)$  (直接讀取 index)

建堆積 build heap(array) ⇒ 陣列轉換成堆積 ⇒  $O(n)$

堆積化 heapify(i) ⇒ 修復特定節點 i 的堆積性質 ⇒  $O(\log n)$