

Heap

⇒ 完全二元樹，滿足堆序順序性質

⇒ 形狀性質：
1. 每層都是滿的，除了最後一層
2. 最後一層節點序由左至右填滿不能有空缺 ⇒ 优点：heap 可以有效率的放在 array 中子造成記憶體浪費

⇒ 應用：“优先併列 priority queue”“排程 scheduling”“堆序排序 heap sort”

⇒ “key”代表优先权 (key 大, priority 高)

堆序順序性質 Heap-Order Property {
1. Max heap 最大堆序 { 父節點的值 \geq 子節點的值
根有最大值
2. Min heap 最小堆序 { 父節點的值 \leq 子節點的值
根有最小值
3. 應用：實作优先併列 priority queues / 堆序排序 heap sort

時間複雜度 {
插入 insert(x) \Rightarrow sift-up 向上調整, 改變層最左邊, 父子比 swap $\rightarrow O(\log n)$
取出 extract root() \Rightarrow sift-down 向下調整, 改變層最左邊, 父子比 swap $\rightarrow O(\log n)$
⇒ 樹的高度 $\log n$
查看 peek_root() \Rightarrow 讀取最大/小 $\rightarrow O(1)$ (直接讀取 index)
建堆序 build heap (array) \Rightarrow 陣列轉換成堆序 $\rightarrow O(n)$
堆序化 heapify(i) \Rightarrow 修復特定節點 i 的堆序性質 $\rightarrow O(\log n)$