

# Heaps - I

A (binary) heap data structure is an array object that can be viewed as a complete binary tree.

completely filled at all levels except possibly at the lowest, which is filled from left.

→ An array  $A[1:n]$  represents a heap object.

↳  $A.\text{heap-size}$  represents the number of elements in the heap

$$0 \leq A.\text{heap-size} \leq n$$

and  $A[1: A.\text{heap-size}]$  contains valid elements.

— root is located at  $A[1]$

— if  $A.\text{heap-size}$  is zero  $\Rightarrow$  the heap is empty.

— Given an index  $i$  of a node :

$$\text{Parent}(i) = \lfloor i/2 \rfloor$$

$$\text{Left}(i) = 2i$$

$$\text{Right}(i) = 2i + 1$$

— There are two kinds of heaps:

- Max-heap , and
- Min-heap

Both satisfy the heap property

— In a max-heap, the max heap property is satisfied at every node  $i$  (other than the root)

$$A[\text{parent}(i)] \geq A[i]$$

→ The largest element in a max-heap is located at the root.

— A min-heap is organized in the opposite way:

min-heap property

$$A[\text{parent}(i)] \leq A[i] , \text{ for any node } i \text{ other than the root}$$

→ the minimum element is at the root