COMPX201 - Data Structures and Algorithms
&
COMPX241 - Software Engineering Development

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Heaps – intro and structure

COMPX201 & COMPX241

Overview

- What is a heap?
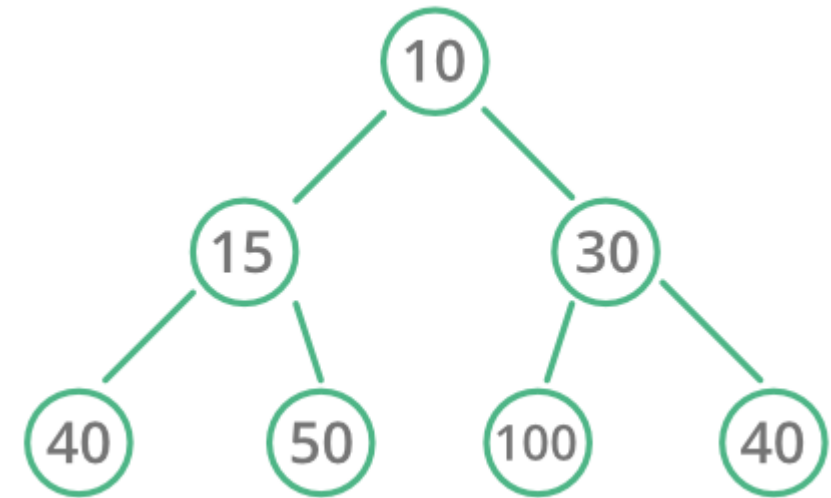- Heap structure
- Min heap
- Max heap

What is a heap?

- A special tree-based data structure
- Where the tree is a complete (or almost complete) binary tree

What is a heap?

- Also thought of as a 'minimum height' binary tree
- Where the deepest level is filled from left to right
- I.e. a 'leftist tree'

Heap structure

- A binary tree that maintains heap-order
  - Every node is the root of a heap (or sub-heap)
  - The value at the root has a higher <u>priority</u> than any of its descendants
- Heap-order is only concerned with the value of its descendants, the relative order of siblings is not important
- A heap can be used to store values in order, and can have duplicate values
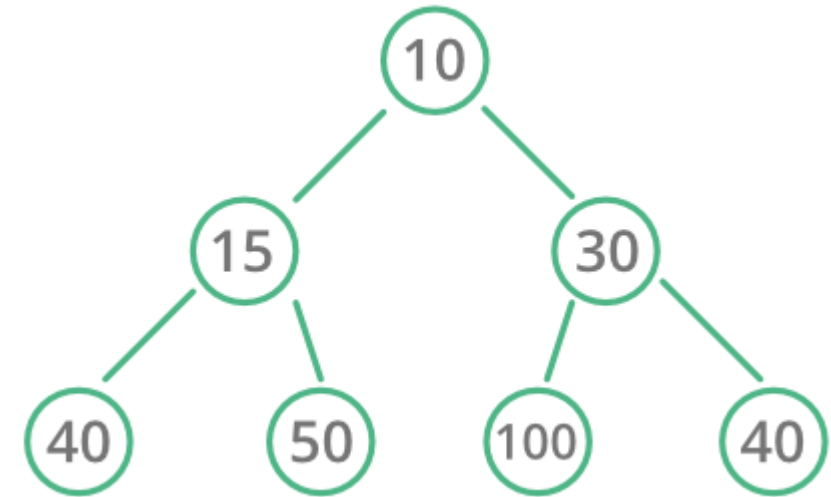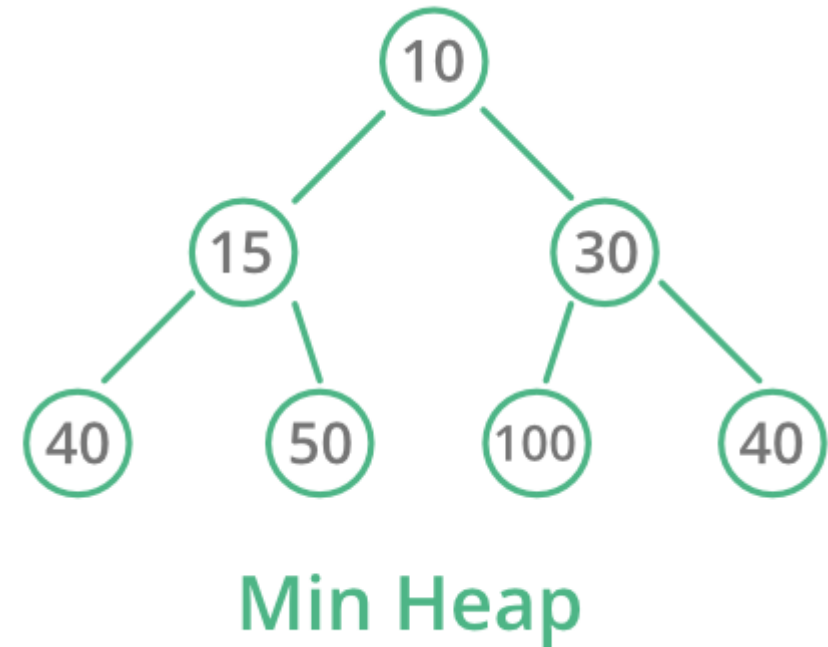
Heap structure

- The value at the root has a higher
  <u>priority</u> than any of its descendants
  - If a smaller value has higher priority than a
    larger one, then we call this a <u>min-heap</u>
  - If a larger value has higher priority, then we
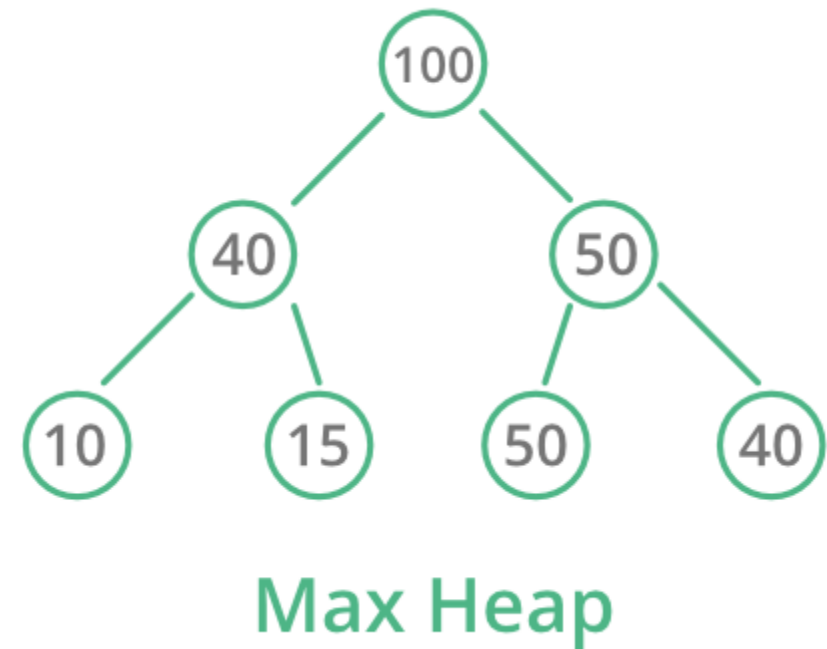   call this a <u>max-heap</u>
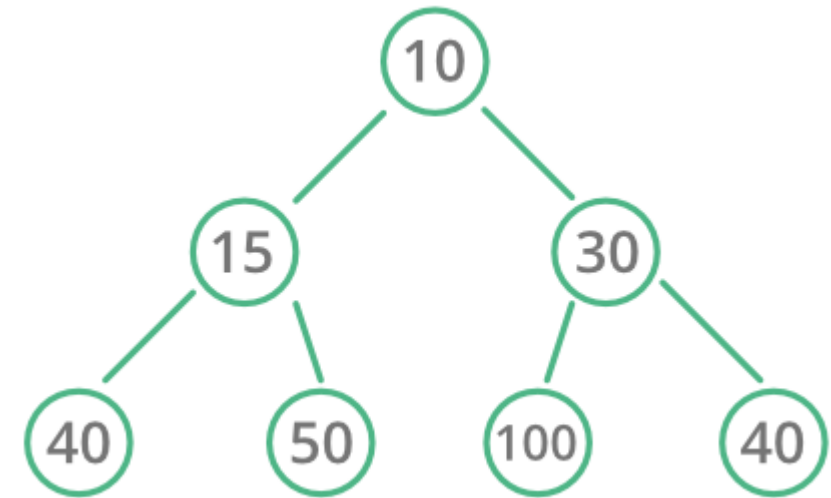
Min heap

- If a smaller value has higher priority than a larger one, then we call this a <u>min-heap</u>
- Smallest value in the collection is at the root and thus can be obtained in order-one O(1) time.
- This is a strength/utility of a heap

Min Heap

Max heap

- If a larger value has higher priority than a smaller one, then we call this a max-heap
- Largest value in the collection is at the root and thus can be obtained in order-one O(1) time.
- This is a strength/utility of a heap

Max Heap

COMPX201 - Data Structures and Algorithms
&
COMPX241 - Software Engineering Development