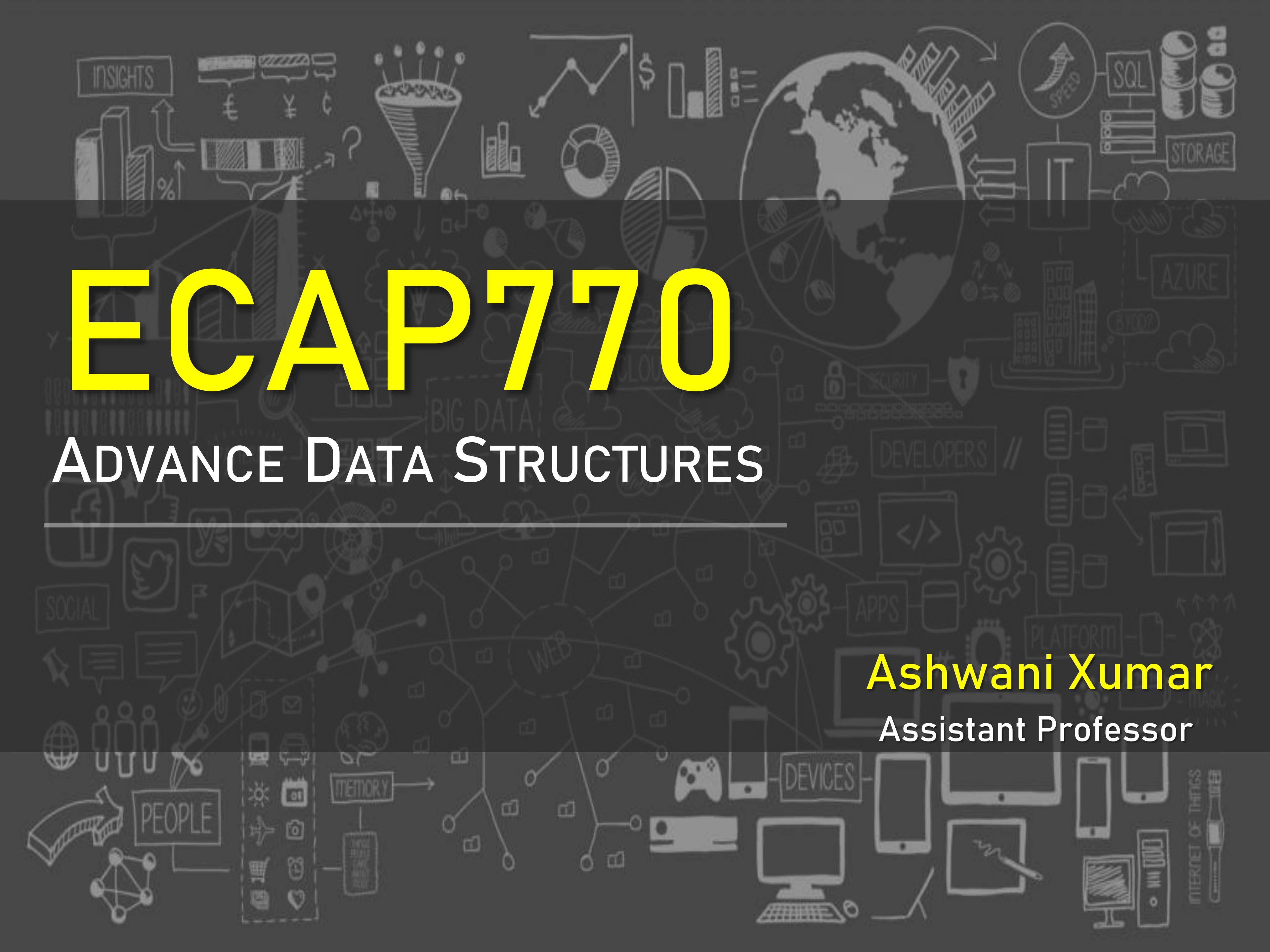


ECAP770

Advance Data Structures

Ashwani Xumar

Assistant Professor

Learning Outcomes

After this lecture, you will be able to

- understand heap sort.

Heap Sort

- Heap sort is sorting technique based upon Binary Heap data structure.
- It is comparison-based sorting technique.
- It processes the elements by creating the min heap or max heap using the elements of the array.

Heap Sort

- Max Heap is used to arrange list of elements in Descending order.
- Min Heap is used to arrange list elements in Ascending order.

Steps for Heap Sort

- Construct a Binary Tree from the list of Elements.
- Transform the Binary Tree into Max Heap / Min Heap.
- Delete the root element from Max Heap / Min Heap
- Reducing the size of heap by 1

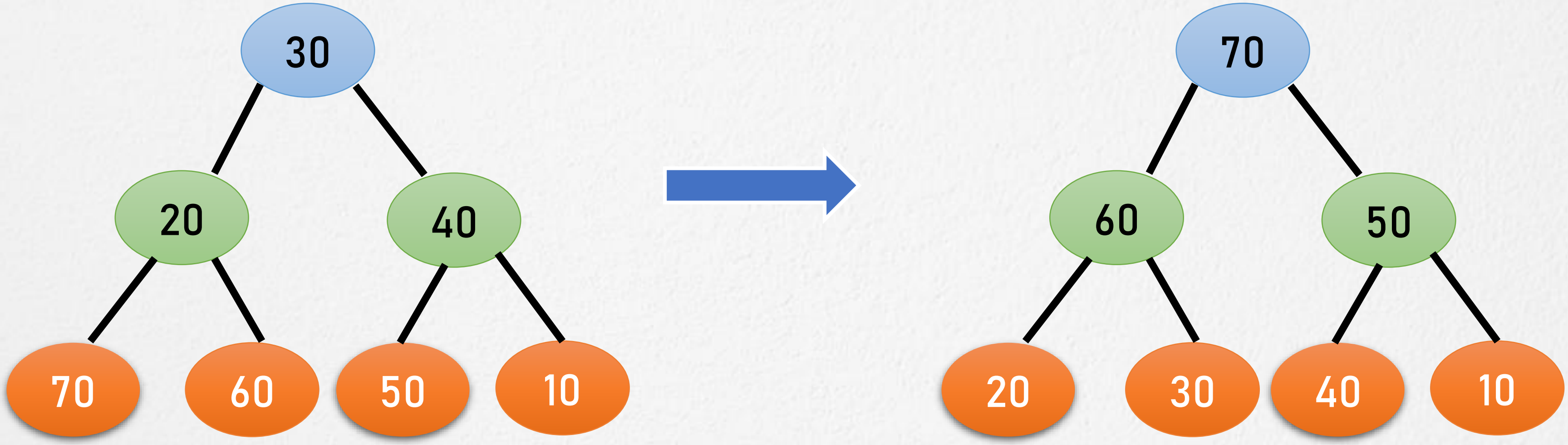
Steps for Heap Sort

- Heapify the root of the tree.
- Put the deleted element into the Sorted list.
- Repeat the same until Min Heap becomes empty.

Example: Heap Sort (Max Heap)

Elements: 30 20 40 70 60 50 10

Elements after sorting: 70 60 50 40 30 20 10

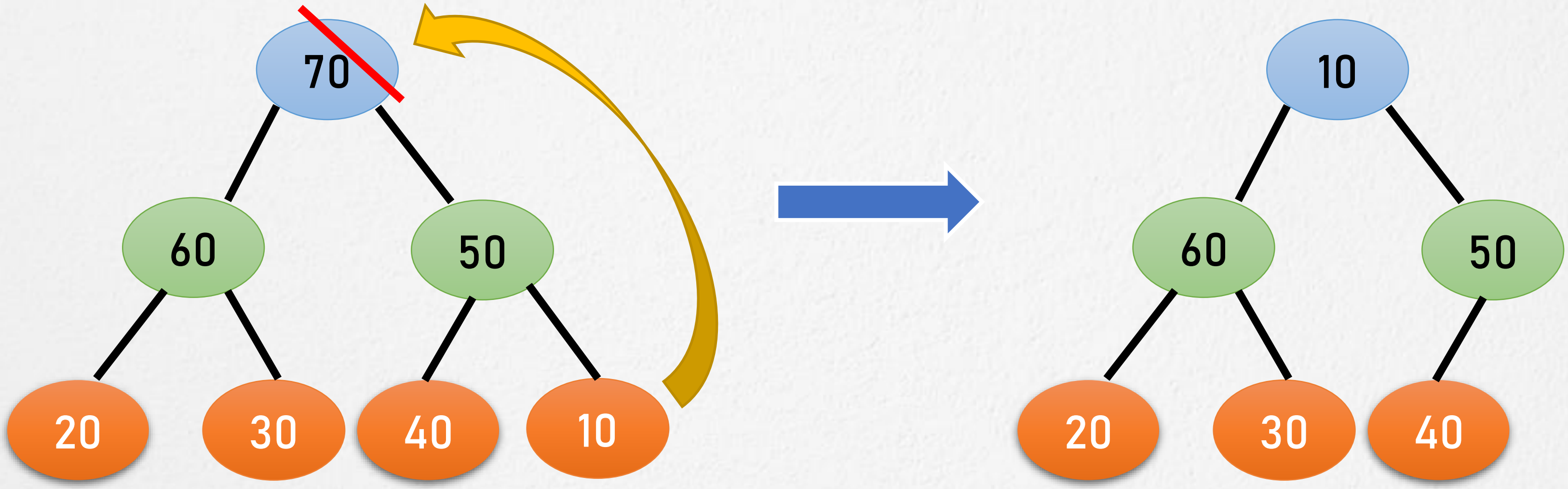


Convert tree into Max Heap

Example: Heap Sort

Elements: 30 20 40 70 60 50 10

Elements: 70

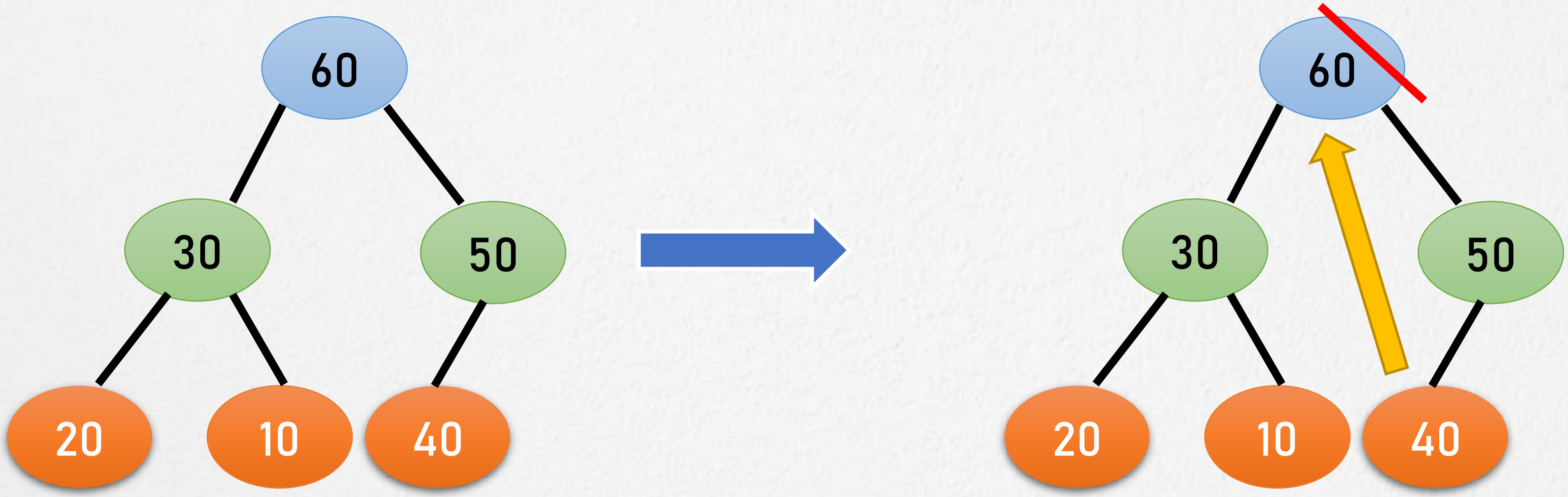


Heapify the root element

Example: Heap Sort

Elements: 30 20 40 70 60 50 10

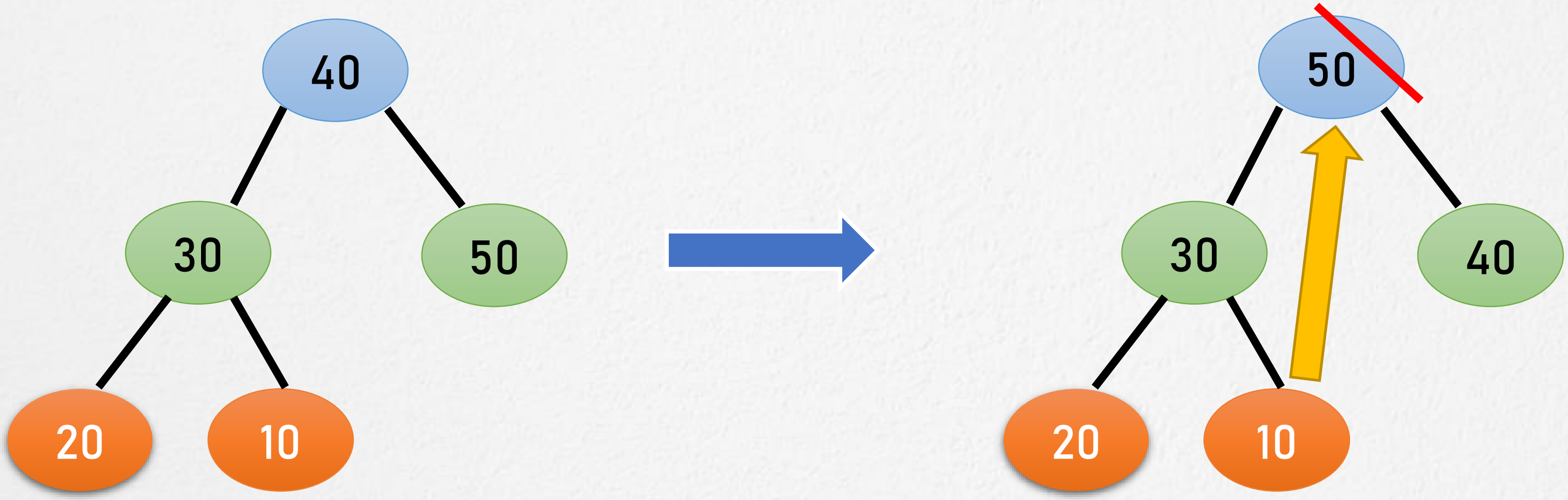
Elements: 70 60



Example: Heap Sort

Elements: 30 20 40 70 60 50 10

Elements: 70 60 50

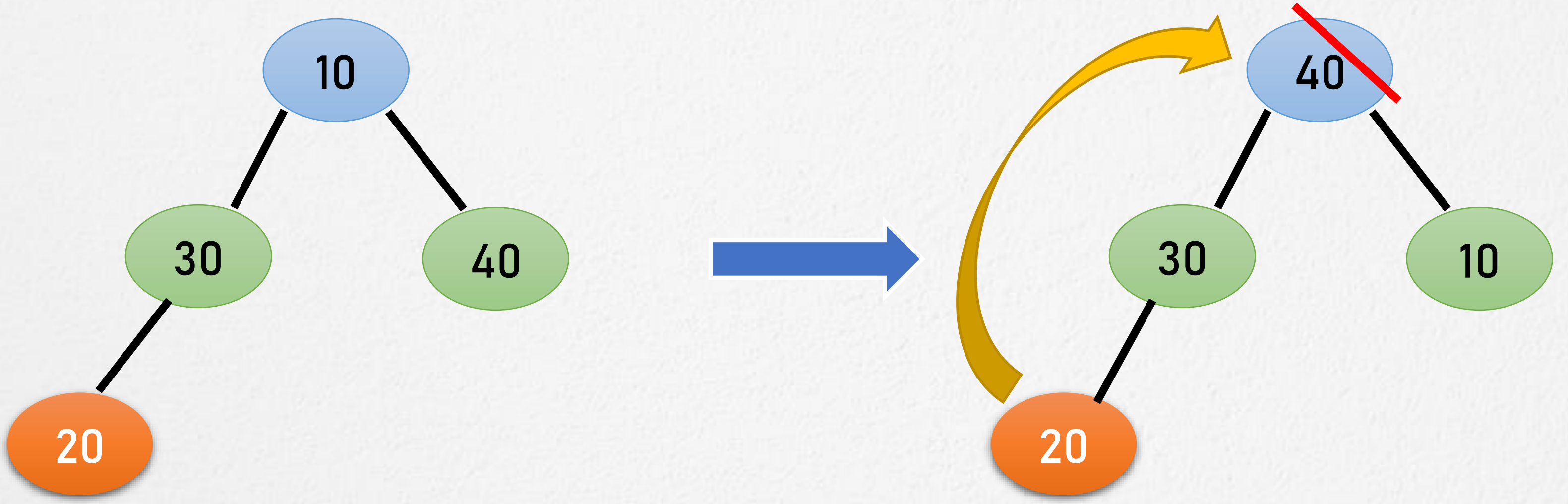


Heapify the root element

Example: Heap Sort

Elements: 30 20 40 70 60 50 10

Elements: 70 60 50 40

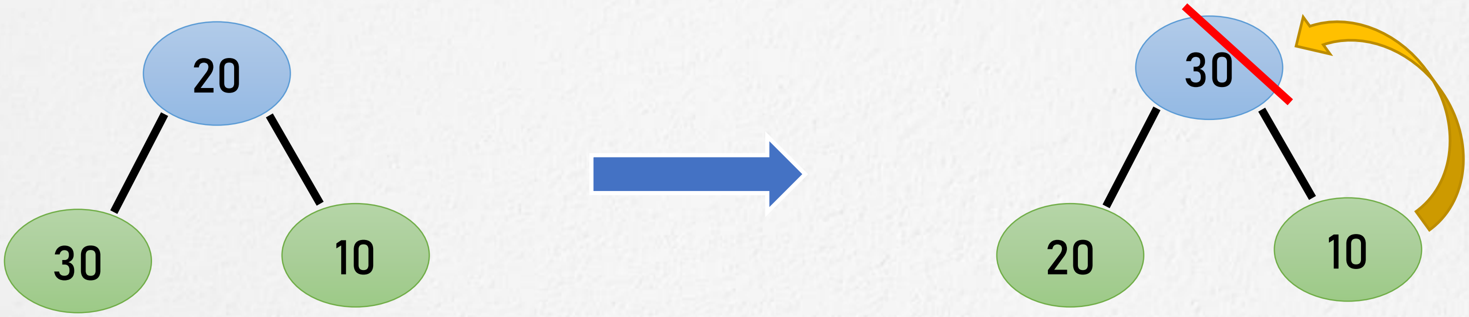


Heapify the root element

Example: Heap Sort

Elements: 30 20 40 70 60 50 10

Elements: 70 60 50 40 30

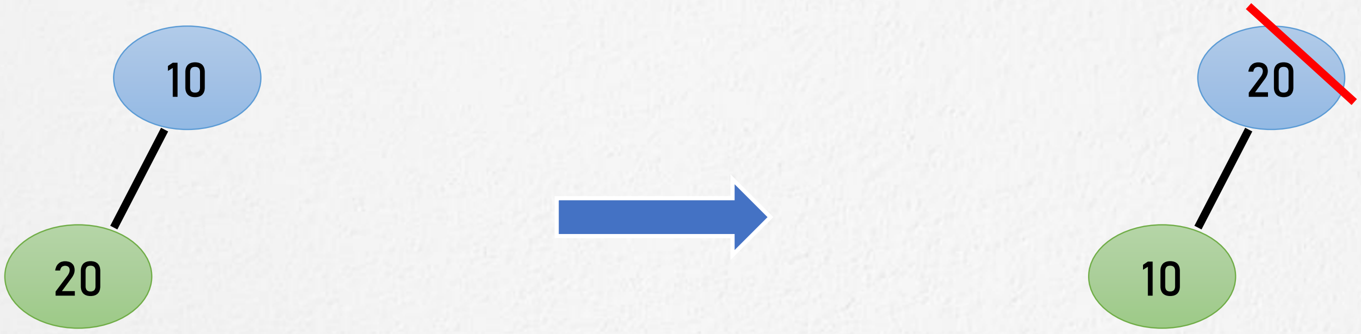


Heapify the root element

Example: Heap Sort

Elements: 30 20 40 70 60 50 10

Elements: 70 60 50 40 30 20 10



Heapify the root element

Complexity of the Heap Sort

- Worst Case : O(n log n)
- Best Case : O(n log n)
- Average Case : O(n log n)

Applications of Heap Sort

- K sorted array
- K largest or smallest elements in an array

That’s all for now…