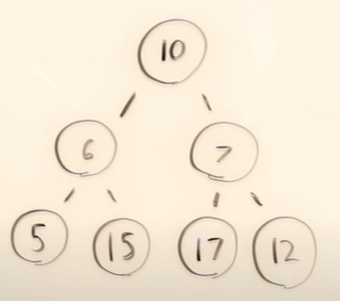
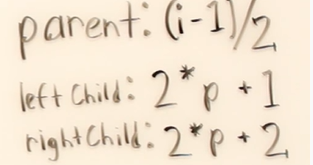
# **HEAP**

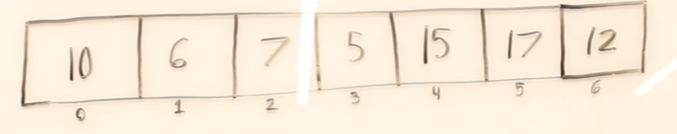
**Heapify** is a process of creating a heap

Heap is not considered an abstract data type. Heap is a specialized tree-based data structure

# **REPRESENTATION OF A BINARY TREE IN ARRAY**

**1-st variant. Start array from index [0]:**

** **

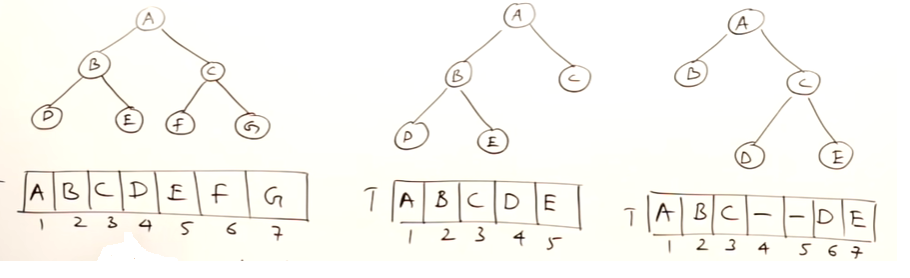


**Formula:**

If [i] is node, where [i] = 0,1,2,3,4 …

* Left child is [2\*i+1]
* right child is [2\*i+2]
* parent is [(i-1)/2]

**2-nd variant. Start array from index [1]:**



**Formula:**

If [i] is node, where [i] = 1,2,3,4 …

* Left child is [2\*i]
* right child is [2\*i+1]
* parent is [i/2]

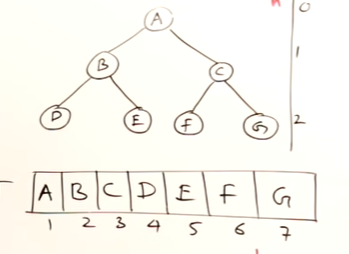
# **BINARY TREE – COMPLETE AND FULL**

**DEF1. Complete binary tree** is a binary tree where all levels filled out except possibly last and in the last level there is no gaps between nodes

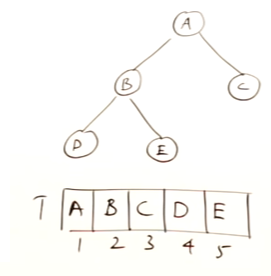
**DEF1. Complete binary tree** is a binary tree and if make an array representation all elements will be filled out

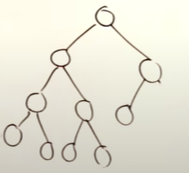
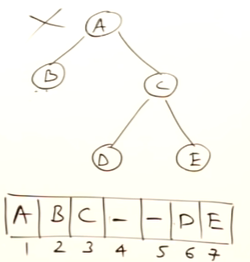
**Full binary tree** is a binary tree where all elements are filled out

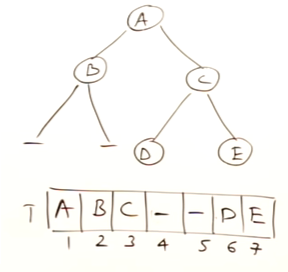
*FULL BINARY TREE*



*COMPLETE BINARY TREE*





# **HEAP**

**Heap** is a binary tree that satisfies two conditions

1. Complete binary tree
2. Every node has grater elements than all child (for max heap)

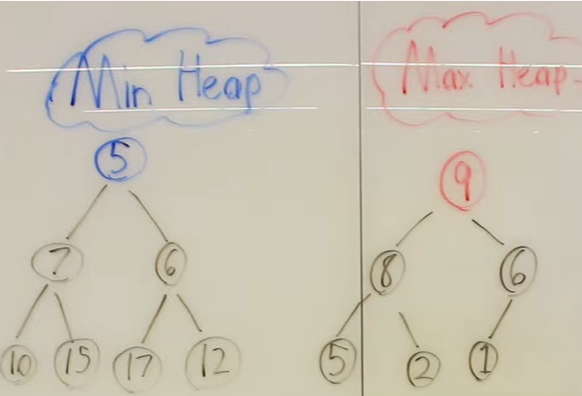
Facts:

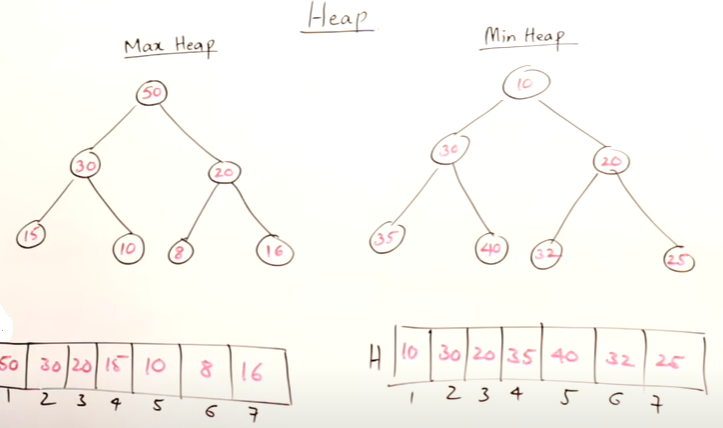
* Heap is not considered an abstract data type. Heap is a specialized tree-based data structure
* The most efficient implementation of heap is PriorityQueue
* Height of heap - O(log(n))
* Insertion into a heap - Ω(1), O(log(n))

Type of heaps:

Min Heap – root has a minimum element among a tree

Max Heap – root has a maximum element among a tree





# **INSERTION IN A HEAP**

* Insertion into a heap - Ω(1), O(log(n))

# **DELETION IN A HEAP**

You can delete only root node in a heap

* Deletion into a heap - Ω(1), O(log(n))

# **HEAP SORT**

Heap Sort consists of two parts

* Create and population of a heap
* Extraction - Deletion from a heap – after deletion elements are sorted

# **HEAPIFY**

**Heapify** is a process of creating a heap

* Heapify - O(n)