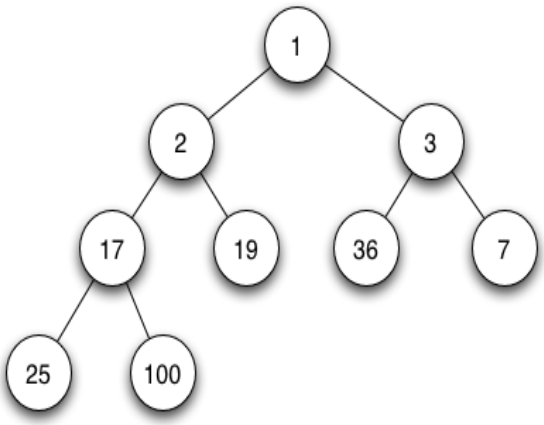
UNIT – I

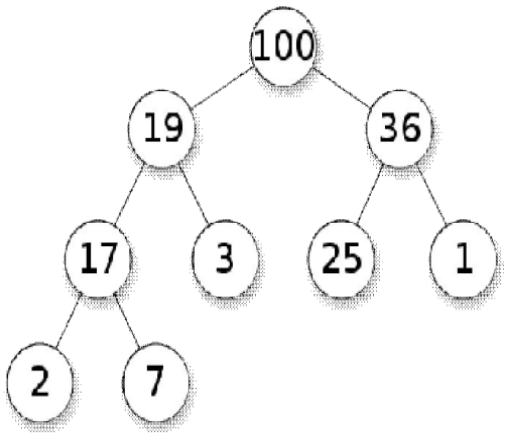
BINARY – HEAPS

- ***Complete Binary Tree:*** A Complete Binary tree is a binary tree in which every level, except possibly the last, is completely filed, and the nodes are as far left as possible.
- ***Max tree:*** It is a tree in which the value at each node is greater than or equal to those in its children.
- ***Min tree:*** It is a tree in which the value at each node is less than or equal to those in its children.
- ***Binary Heap/Heap Tree:*** A Heap is a Complete Binary tree with elements from partially ordered set which satisfies Heap Ordering property. The ordering can be of 2 types:
-

1. ***Min Heap:*** For each node N in a complete binary tree, the value of N is less than or equal to the value of its children's, such a heap tree is called a Min Heap.



2. ***Max Heap:*** For each node N in a complete binary tree, the value of N is greater than or equal to the value of its children's, such a heap tree is called a Max Heap.



Array Representation of Heap Tree/Binary Heap:

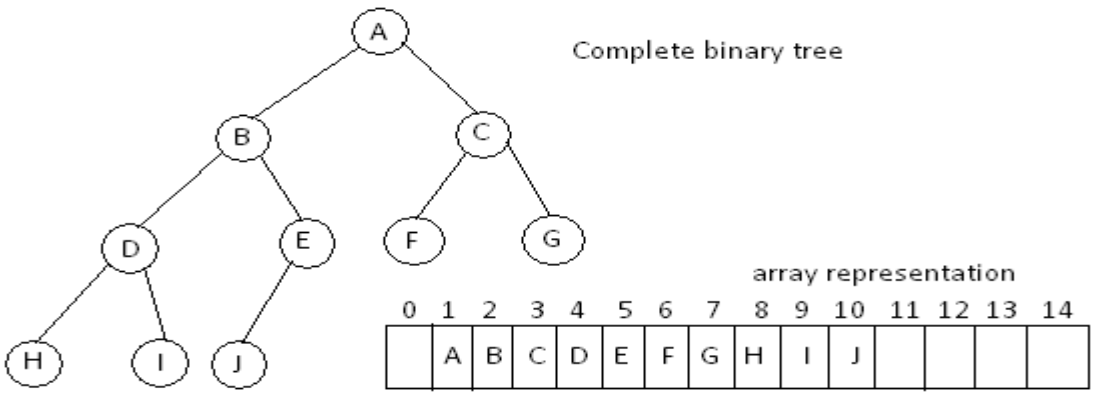
- A heap tree can be represented using array and linked list.
- For an element in array position i, the left child is in the position 2i, the right child is at position (2i+1) i.e in the cell after the left child.
- The parent is in position floor(i/2).

Heaps (occasionally called as partially ordered trees) are a very popular data structure for implementing priority queues.

Binary heaps are refer to merely as heaps, like binary search trees, heaps have two properties, namely, a **structure property** and a **heap order property**.

Structure property:

A heap is a binary tree that is completely filled, with the possible exception of the bottom level, which is filled from left to right, such tree is known as a complete binary tree as shown below



A binary heap is a complete binary tree with elements from a partially ordered set, such that the element at every node is less than (or equal to) the element at its left child and the element at its right child.

It is easy to show that a complete binary tree height ' h ' has between 2h and 2h+1-1 nodes. This implies that the height of a complete binary tree is $\lfloor \log N \rfloor$, which is clearly O(log N).

One important observation is that because a complete binary tree is so regular, it can be represented in an array and no pointers are necessary.

Since a heap is a complete binary tree, the elements can be conveniently stored in an array. If an element is at position i in the array, then the left child will be in position 2i, the right child will be in the position (2i+1), and the parent is in position $\lfloor i/2 \rfloor$

The only problem with this implementation is that an estimate of the maximum heap size is required in advance, but typically this is not a problem.

Because of the heap property, the minimum element will always be present at the root of the heap. Thus the find min operation will have worst case O(1) running time.

Heap – order property:

It is the property that allows operations to be performed quickly. Since we want to be able to find the minimum quickly, it makes sense that the smallest element should be at the root.

If we consider that any sub tree should also be a heap, then any node should be smaller than all of its descendants.

Applying this logic, we arrive at the heap order property. In a heap, for every node X, the key in the parent of X is smaller than (or equal to) the key X, with exception of the root. (Which has no parent)?

NOTE: Binary heaps were first introduced by Williams in 1964.

NOTE: Binary Heap is either a min – heap or a max – heap. A min heap supports the insert and delete min operations while a max heap supports the insert and delete max operations

Reference Links

1. https://www.codesdope.com/blog/article/priority-queue-using-heap/
2. https://bradfieldcs.com/algos/trees/priority-queues-with-binary-heaps/

Video Links

1. https://www.youtube.com/watch?v=EoVilfrlZIA

Questions

1. What is binary heap? Explain the procedure to insert an element into binary heap.
2. Write an algorithm to insert an element in to a heap. Explain with a suitable example.
3. Discuss about binary heap concept.