3	8	4	19	20	2	36

$O(n)$
given you smallest no.

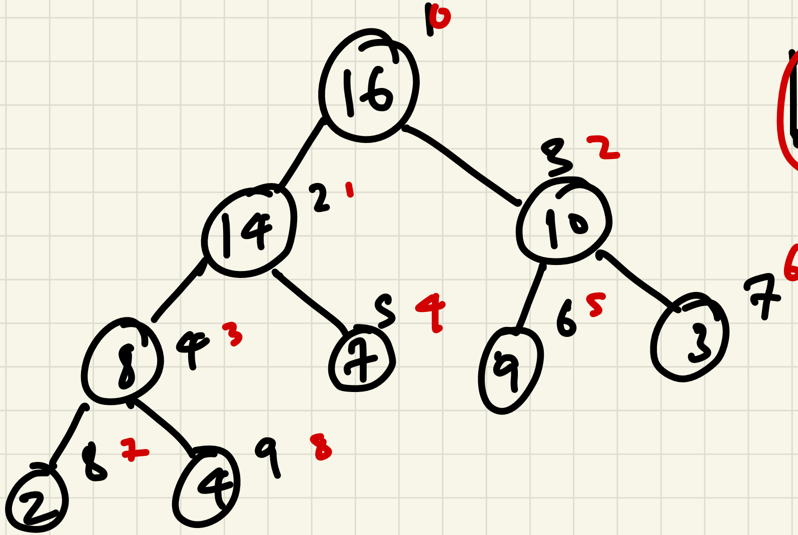
$O(1)$

2	3	4	8	19	20	36	. . .

$\to O(1)$ ✓✓

$O(N \log N)$ to insert one item.

Can you reduce this ?? $\longrightarrow$ Heaps



1	2	3	4	5	6	7	8	9	10
16	14	10	8	7	9	3	2	4	

1. Complete binary tree
2. Every node value $\geq$ All of its children

may not be sorted

No pointers required.

root $\Rightarrow$ $i = 1$

height: $\log(n)$

$parent(i) = \frac{i}{2}$

$left(i) = 2 * i$

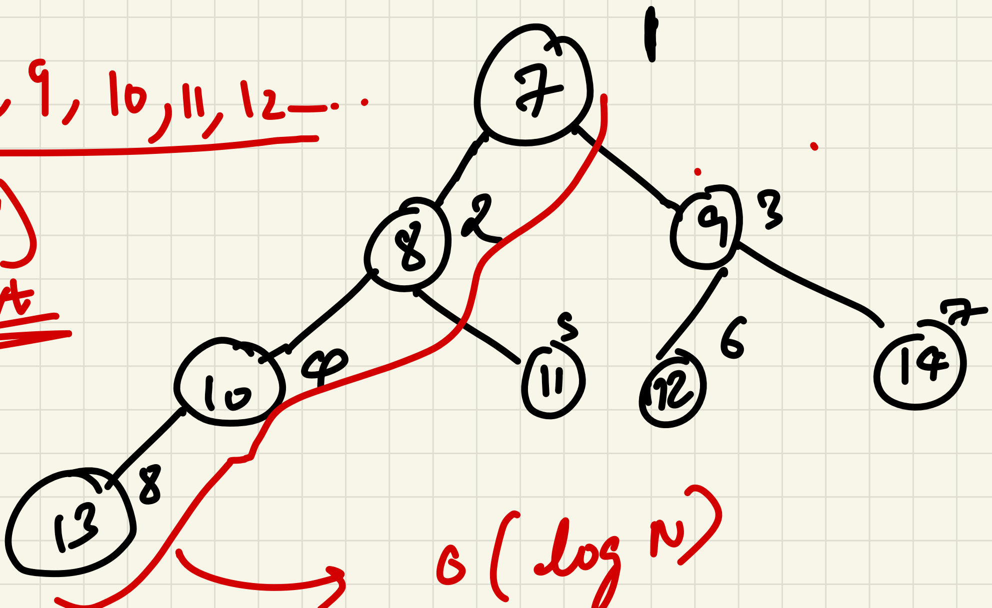
$right(i) = 2 * i + 1$

1	2	3	4	5	6	7	8	9	10	11
7	8	9	10	11	12	14	13			

7, 8, 9, 10, 11, 12 ...

$O(n \log n)$

~~Heapsort~~



$O(\log N)$

val(node) ≤ all it's children

For 10,

13 > 10

5 < 10

then go c

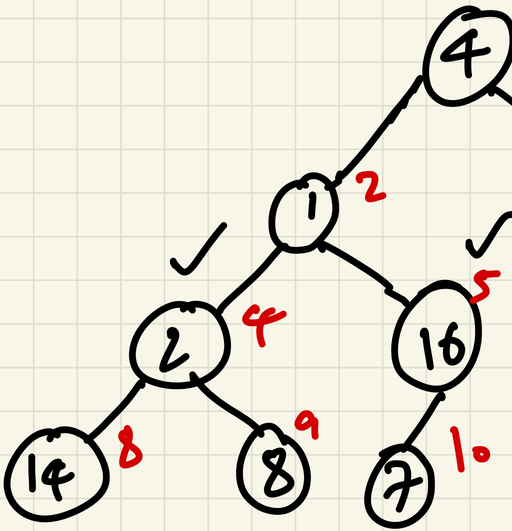
5 < 13

$3 \to 4 \to 8 \to 10 \to 18$

$O(N)$ time

pq → LL ✗

1	2	3	4	5	6	7	8	9	10
4	1	3	2	16	9	10	14	8	7



why start from $\frac{N}{2}$?

$\left[\frac{N}{2} + 1 \ldots n\right]$

leaf nodes

for (i = $\frac{N}{2}$ till 1)

downheap (i)

$O(N * \log(N))$ X

Leaf nodes $= \frac{N}{2} = 0$

One level above leaf nodes $= \frac{N}{4} = c$

level 2 $= \frac{N}{8} = 2c$

level 3 $= \frac{N}{16} = 3c$

$\vdots$

root $= 1 = \log N * c$

$$\frac{N}{4} \times C + \frac{N}{8}(2C) + \frac{N}{16}(3C) + \dots + 1(C \log N)$$

$$\frac{N}{4} = 2^k$$

$$= C\, 2^k \underbrace{\left(\frac{1}{2^0} + \frac{2}{2^1} + \frac{3}{2^2} + \frac{4}{2^3} + \dots + \frac{k+1}{2^k} \right)}$$

Bounded by a constant.

Sum of $i x^i$ to ∞ $= \dfrac{x}{(1-x)^2}$

Sums in maths

$$O(2^k) = O(N)$$