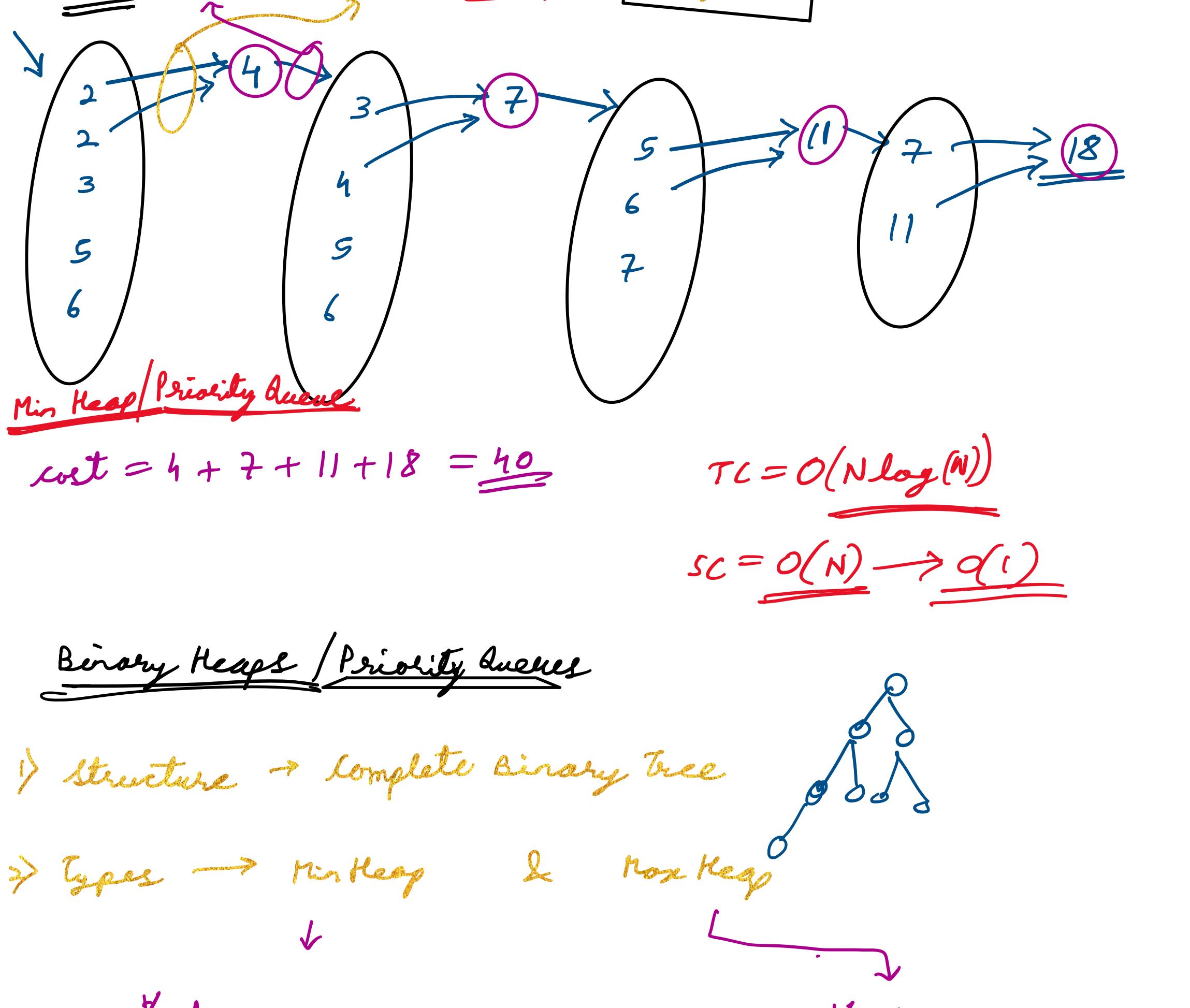
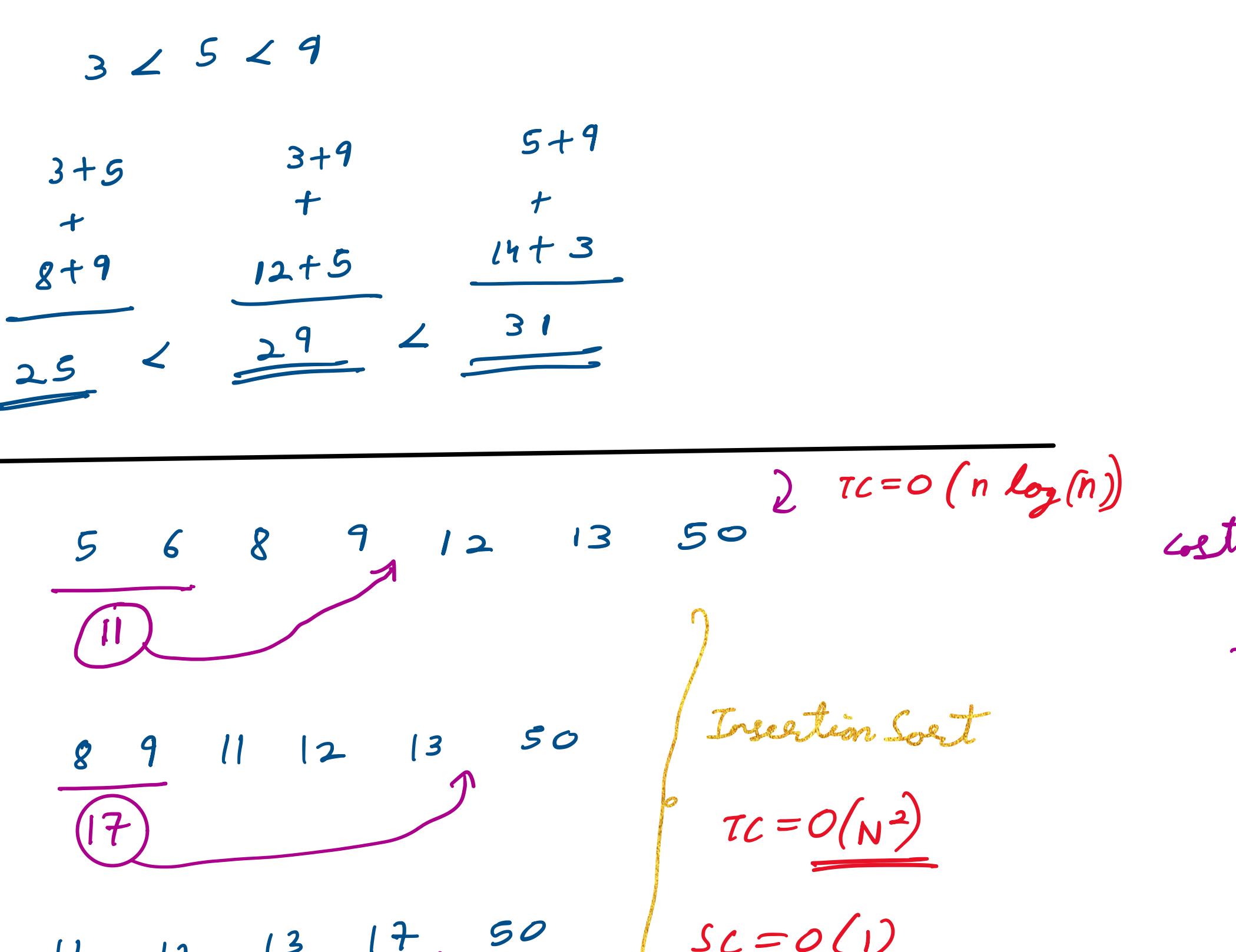
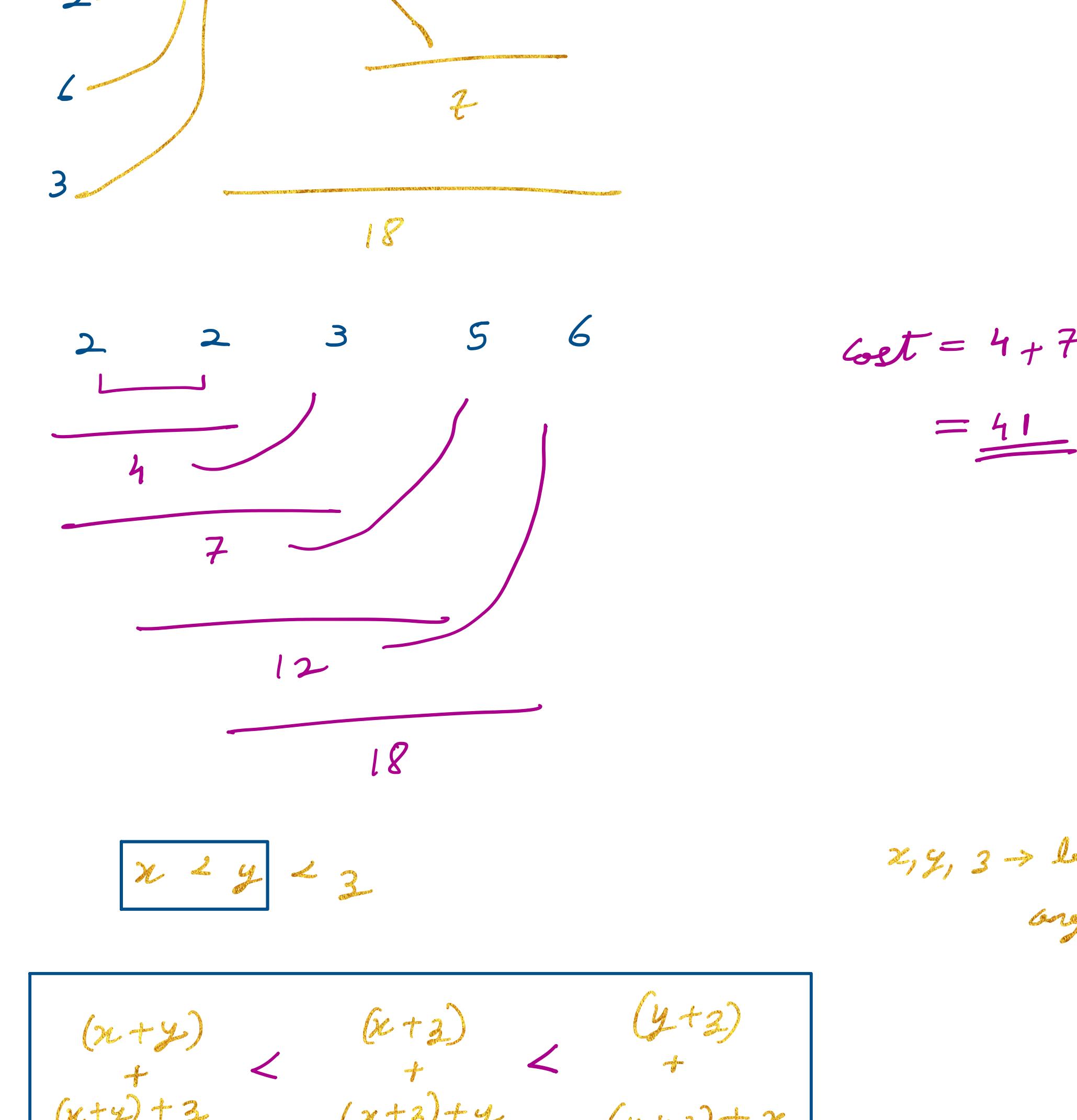


Heaps 1

Tuesday, 5 October 2021 9:02 PM

Q) Given N ropes with their length in an array.

Cost of connecting 2 ropes = sum of length of both the ropes.
Goal → connect all the ropes with min cost.



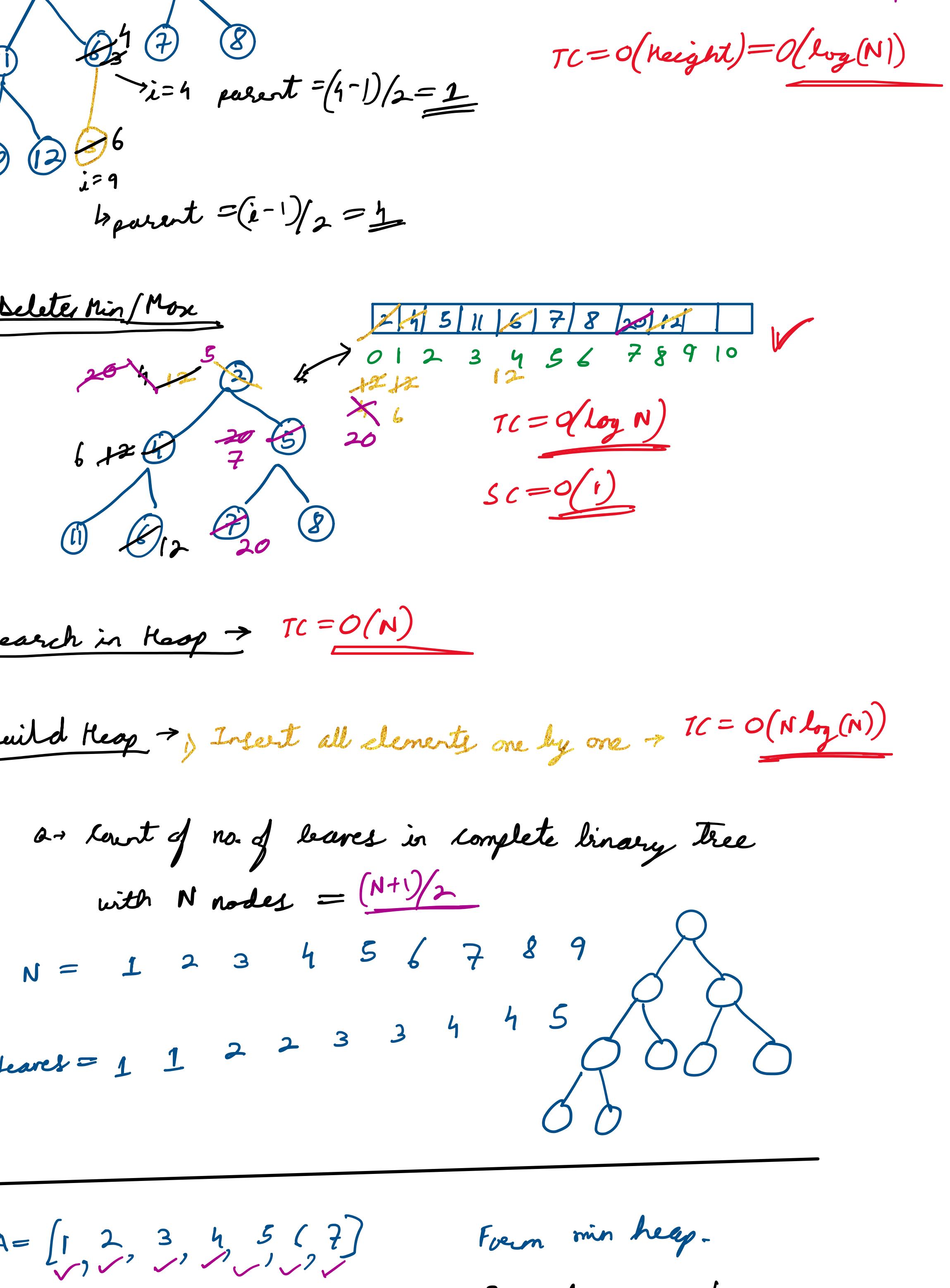
$$x < y < z$$

$x, y, z \rightarrow$ length of any 3 ropes

$$\begin{array}{c} (x+y) \\ (x+y)+z \\ (x+y)+z+x \end{array} < \begin{array}{c} (x+z) \\ (x+z)+y \\ (x+z)+y+z \end{array} < \begin{array}{c} (y+z) \\ (y+z)+x \\ (y+z)+x+z \end{array}$$

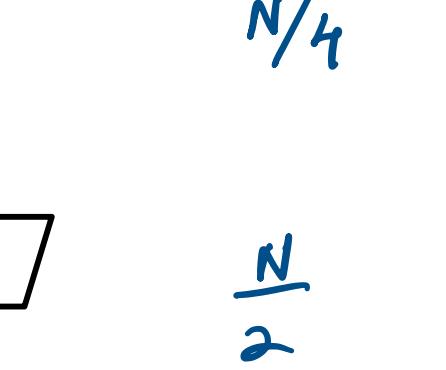
$$3 < 5 < 9$$

$$\begin{array}{c} 3+5 \\ + \\ 8+9 \\ \hline 25 \end{array} < \begin{array}{c} 3+9 \\ + \\ 12+5 \\ \hline 29 \end{array} < \begin{array}{c} 5+9 \\ + \\ 14+3 \\ \hline 31 \end{array}$$



Binary Heaps / Priority Queues

→ Structure → Complete Binary Tree



→ Types → MinHeap & MaxHeap

↓

MinHeap

node.data ≤ node.left.data

node.data ≤ node.right.data

MaxHeap

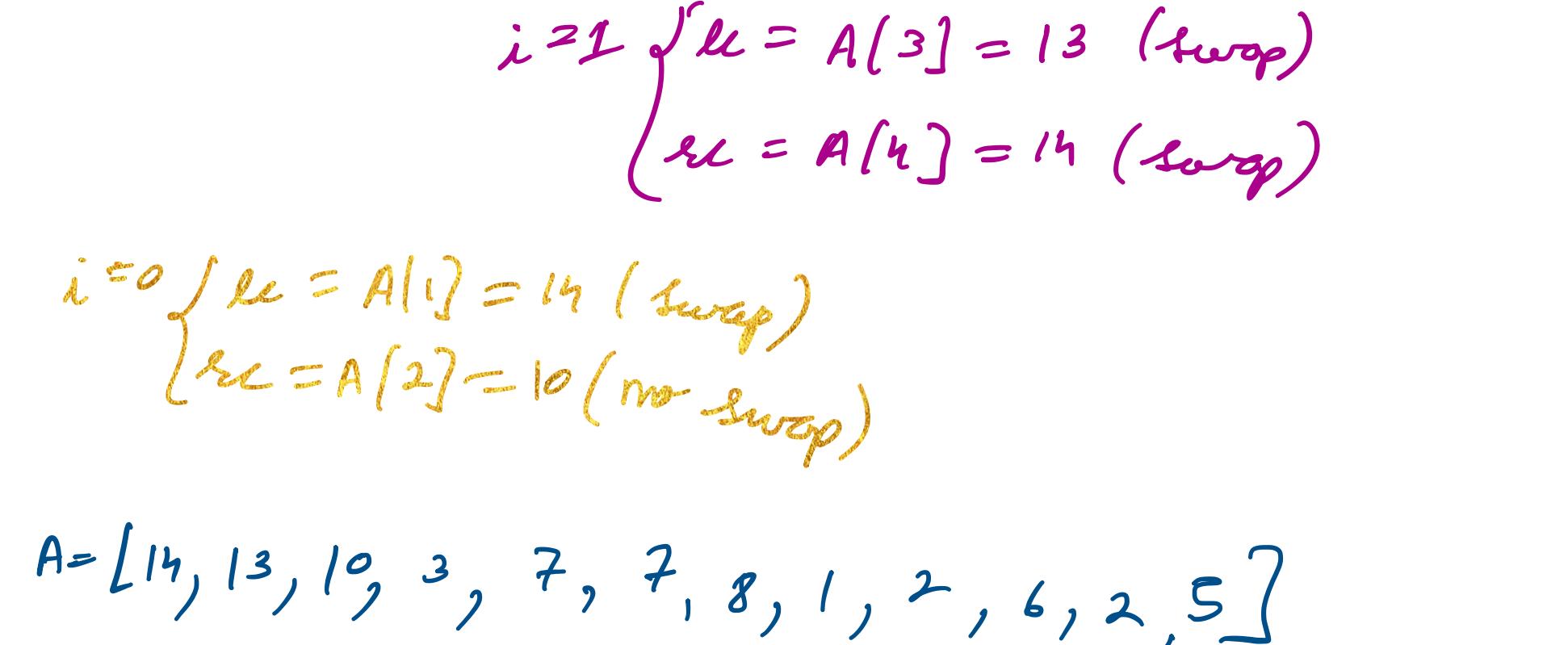
node.data ≥ node.left.data

node.data ≥ node.right.data

3) No relation b/w left & right subtree.

4) Heapsify → maintaining the properties of heap after any insertion/deletion.

5) Array representation of tree → Root → 0



Insertion

0	1	2	3	4	5	6	7	8	9	10
2	4	5	11	6	7	8	10	12	1	9



$i = 1$ $lc = A[3] = 2$ (swap)

$lc = A[2] = 1$ (swap)

$i = 0$ $lc = A[1] = 1$ (swap)

$lc = A[0] = 0$ (no swap)

A = [14, 13, 10, 3, 7, 1, 8, 1, 2, 6, 2, 5]

$sc = O(1)$

$TC = O(N)$

Convert array to Max Heap.

to Max Heap.

Elements $\frac{N}{8}$ swaps 2

$N/4$ 1

$\frac{N}{2}$ 0

Binary Heap

$TC = \frac{N}{2} * 0 + \frac{N}{4} * 1 + \frac{N}{8} * 2 + \dots$

$TC = \frac{N}{2} * 2 = O(N)$

$sc = O(N) \rightarrow O(1)$

Inplace Heap Build

Convert input array to Heap.

A = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2 6 2 5

14 13 10 3 7 1 8 1 2