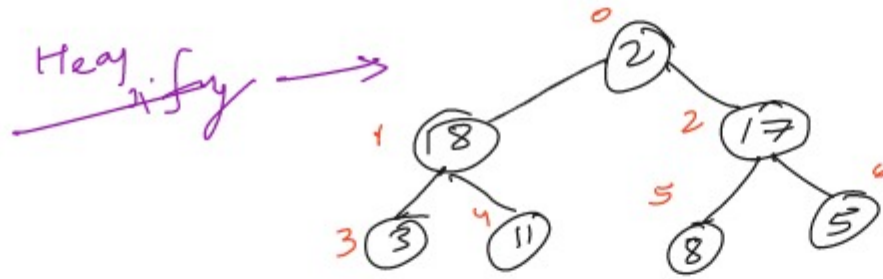


$[2, 18, 5, 3, 11, 8, 17]$

make heap

start at 5. $(7/2 - 1) = 2$



Current Index = 6
max Index = 6

make heap

start at Index ①

$Lidx = 3$ $Ridx = 3$
 $Ridx = 4$ $ARidx = 11$
 $mididx = 6$

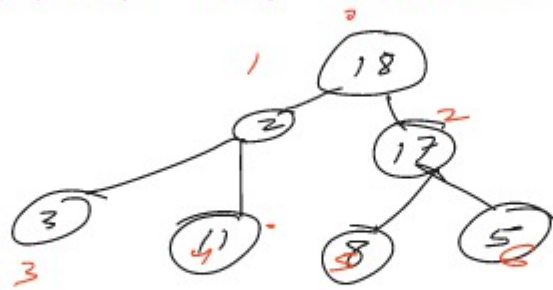
Heapify

Since at index 1
heap is already present
we get out of heapify.

Make heap. start at index 0

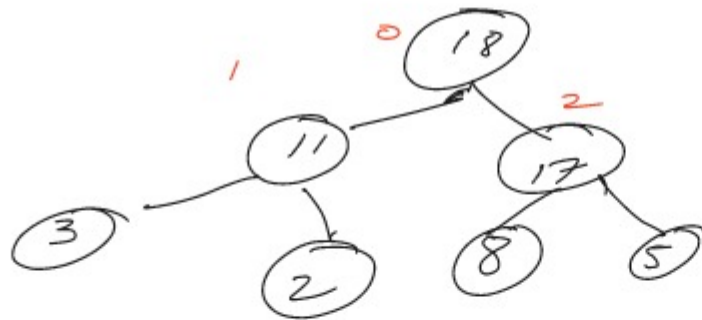
$lidx = 1$ $alidx = 18$

$ridx = 2$ $aridx = 17$



Current index = 1
 $lidx = 3$ $alidx = 3$
 $ridx = 4$ $aridx = 11$
 $alidx = 7$

heapify at index 1



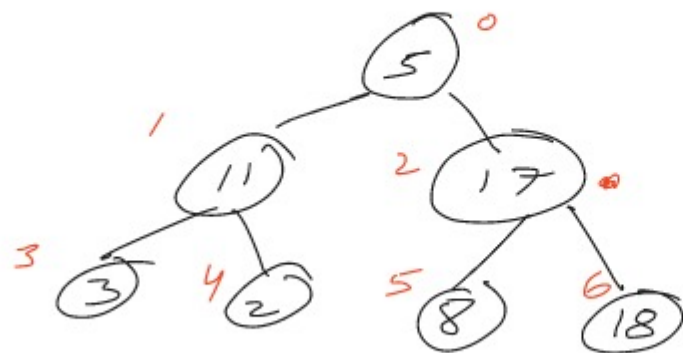
Current index = 4
 $lidx = 9$ $alidx = \text{undefined}$
 $ridx = 10$ $aridx = \text{undefined}$
 get out of loop

[18, 11, 17, 3, 2, 8, 5]

max = 6

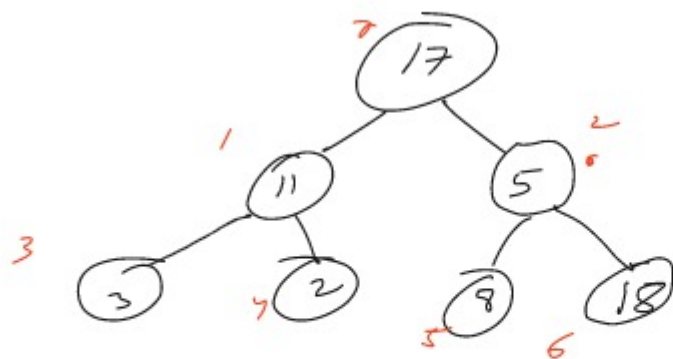
Swap 18 with 5.

$[5, 11, 17, 3, 2, 8, 18]$ & make max
index = 6.



current index = 0
max index = 6

start heapify at index 0

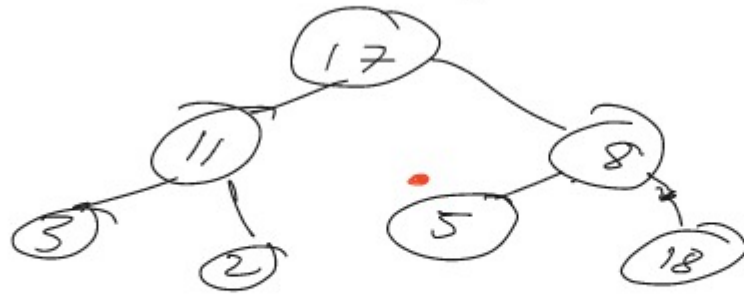


Current index = 2
max index = 6

Left index = 5
Right index = 6

Compare 5 with 8 & skip 18
8 is greater than 5

Swap the root.



Current Index = 5

L Index = 11

R Index = 12

L Index & R Index greater
than Max Index = 6
So Skip Heapify.

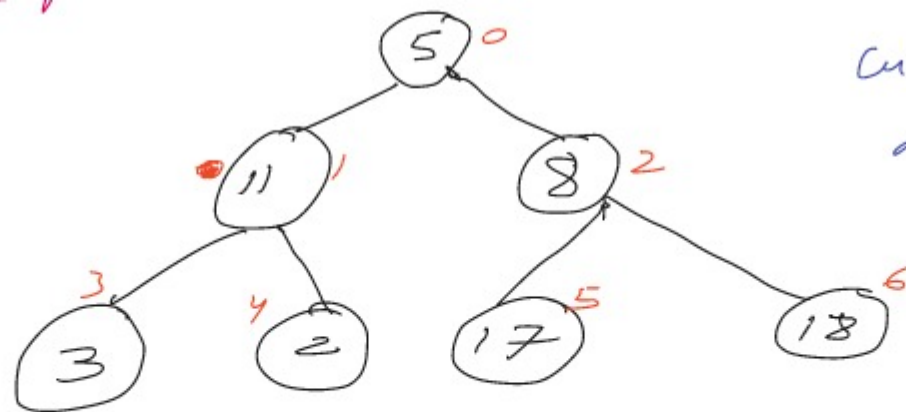
[17, 11, 8, 3, 2, 5, 18]

Sort Max Index = 5

Swap 17 with index Value 5

[5, 11, 8, 3, 2, 17, 18]

Heapify with max Index = 5

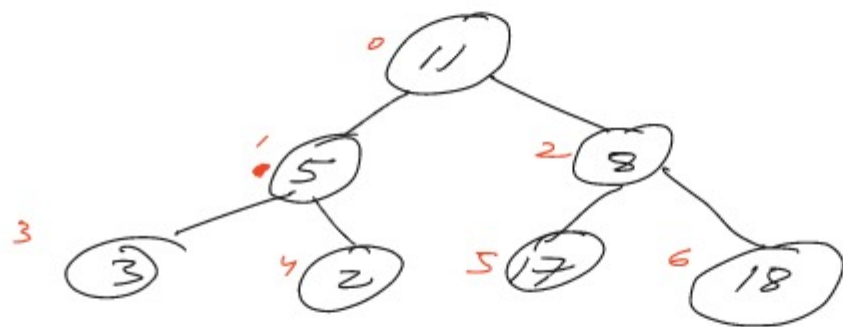


Current $idx = 0$

max $idx = 5$

L $idx = 1$

R $idx = 2$



Current $idx = 1$

max $idx = 5$

L $idx = 3$

R $idx = 4$

Exit from Heapify since at idx 1

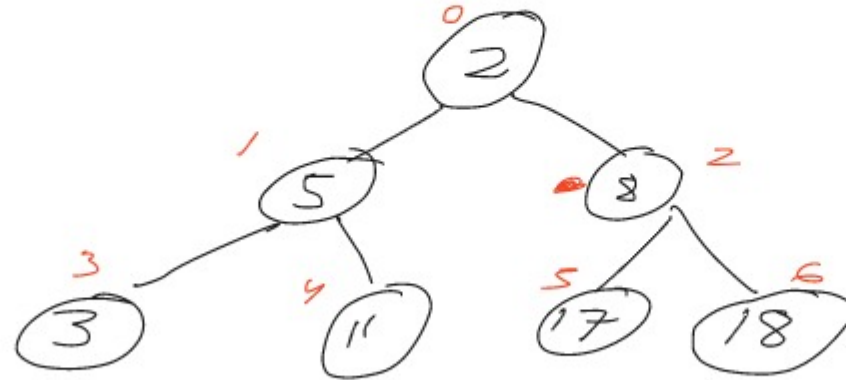
it is already a heap

[11, 5, 8, 3, 2, 17, 18]

Sort:- Reduce Max Index by 1.
Max Index = 4

Swap 11 with 2 (14x4)

[2, 5, 8, 3, 11, 17, 18]



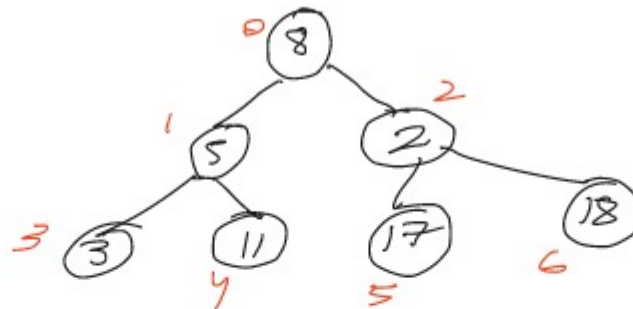
Current idx = 0

L idx = 1

R idx = 2

Max idx = 4

Heapify at idx = 0



Current idx = 2

L idx = 5

R idx = 6

Max idx = 4

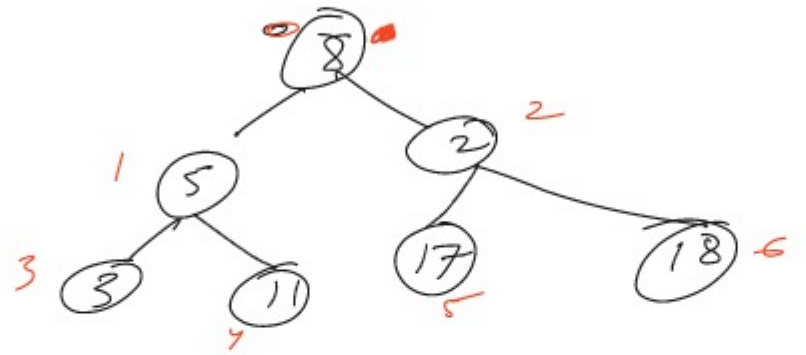
Since both L & R idx greater than max idx Exit the

loop.
[8, 5, 2, 3, 11, 17, 18]

Sort Reduce the Max idx by One

$$\text{max idx} = 3$$

$$\text{Current idx} = 0$$



$$L \text{ idx} = 1$$

$$R \text{ idx} = 2$$

$$C \text{ idx} = 0$$

$$M \text{ idx} = 3$$

at index 0 it is already a
Heap. Exit Heapify

[8, 5, 2, 3, 11, 17, 18]

Sort

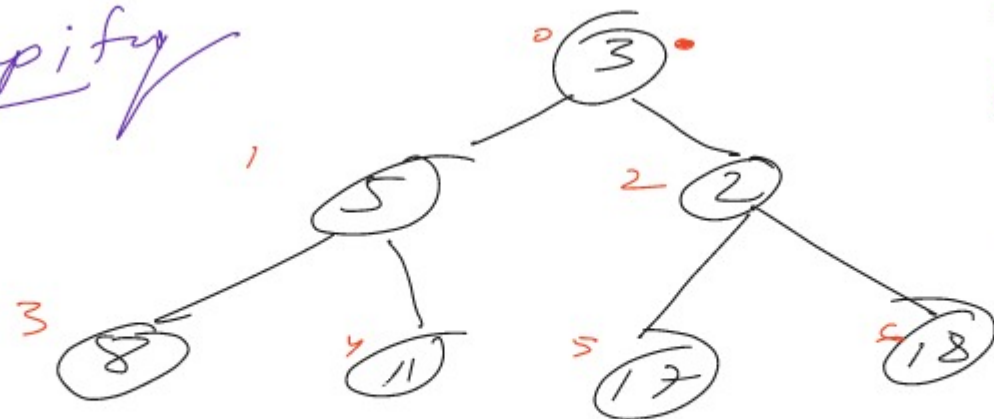
$$= 3$$

Swap 8 by index 3

[3, 5, 2, 8, 11, 17, 18]

$$\text{max idx} - 1 = 2$$

Heapify

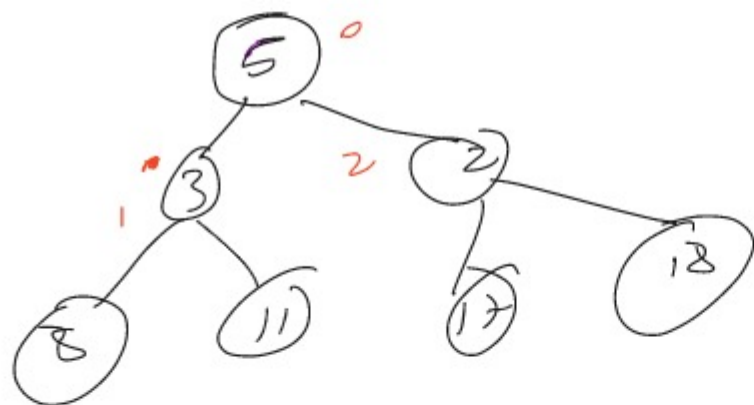


M idx = 2

Cw idx = 0

L idx = 1

R idx = 2



C idx = 1

L idx = 3

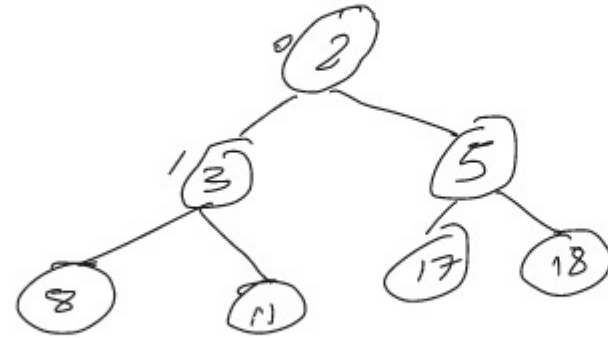
R idx = 4

M idx = 2

Since L idx & R idx Greater than M idx Exit Heapify.

[5, 3, 2, 8, 11, 17, 18]

Sort swap 5 with 2
 $[2, 3, 5, 8, 11, 17, 18]$



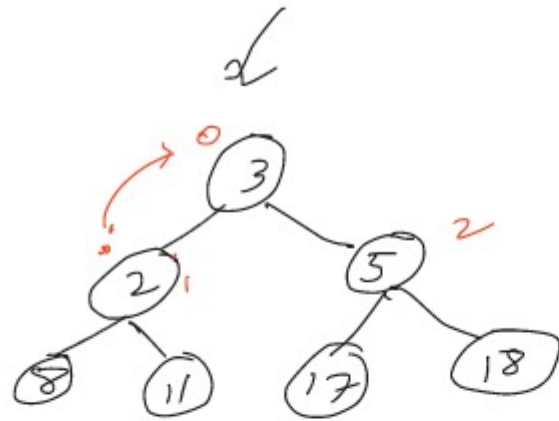
$$\text{maxIndex} = \text{maxIndex} - 1 \\ = 1$$

$$\text{Current Index} = 0$$

$$L \text{ idx} = 1$$

$$R \text{ idx} = 2$$

$$M \text{ idx} = 1$$



$$\left[\begin{array}{l} v(L \text{ idx}) \geq v(R \text{ idx}) \\ \parallel \\ R \text{ idx} > M \text{ idx} \end{array} \right] \text{ swap left node}$$

$$\left[\begin{array}{l} v(L \text{ idx}) \geq v(C \text{ idx}) \\ L \text{ idx} < M \text{ idx} \end{array} \right]$$

$[3, 2, 5, 8, 11, 17, 18]$
 ↑

Sort swap 3 with 2

$[2, 3, 5, 8, 11, 17, 18]$

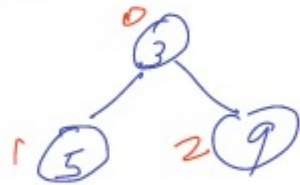
Left Swap

$$arr[Lidx] \geq arr[Ridx] \quad || \quad ridx \geq midx$$

$$arr[Lidx] \geq arr[Cidx]$$

$$Lidx < Midx.$$

Right Swap



$$(arr[Ridx] \geq arr[Lidx])$$

$$Lidx \geq Midx$$

$$arr[Ridx] \geq arr[Cidx]$$

$$Ridx < Midx$$