

Computer Science & IT

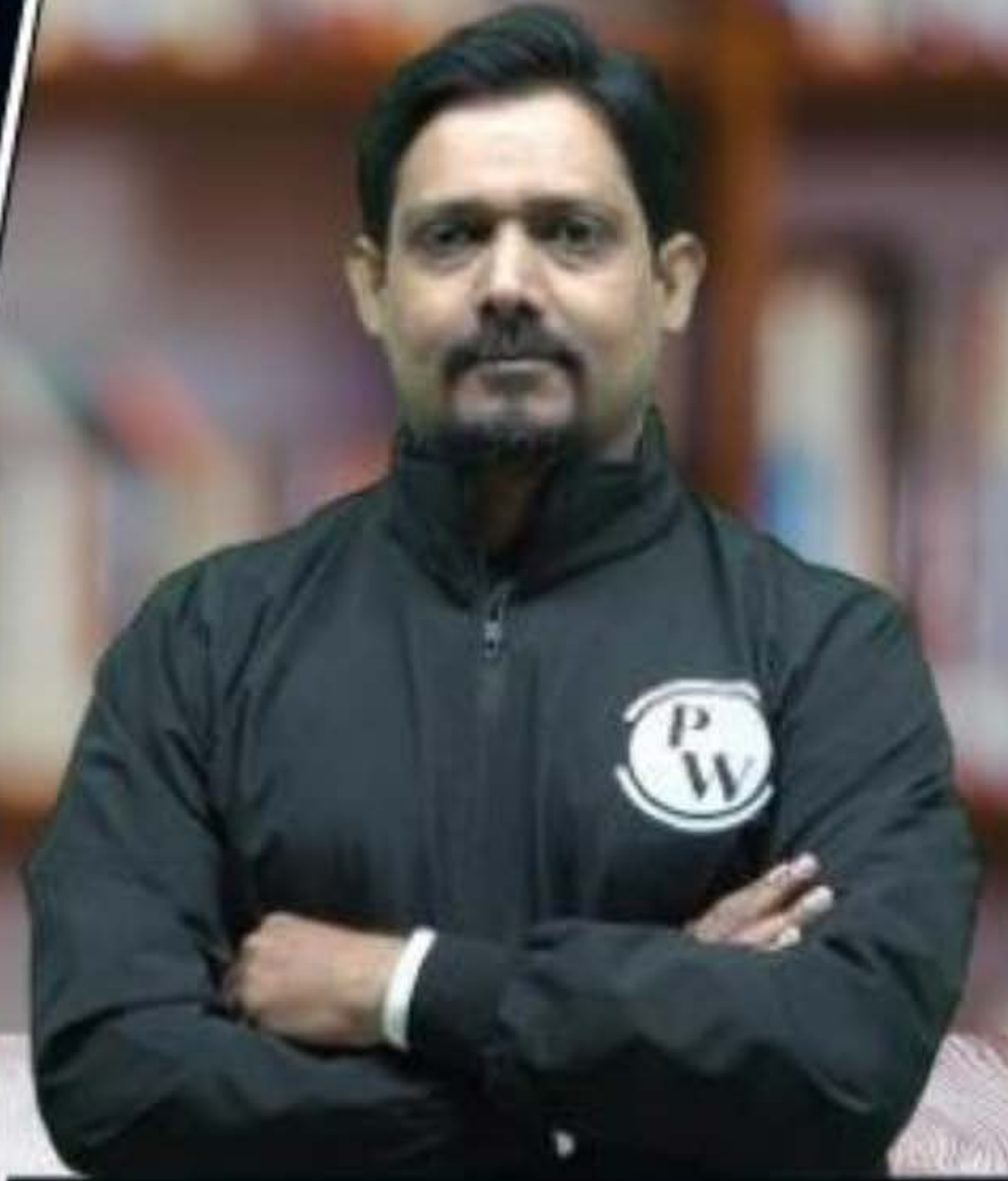
Data Structure & Programming



Tree

Lecture No. 09

By- Abhishek Sir



Recap of Previous Lecture



Topic

Heap

Topic

CBT, Height

Topic

Insertion & Deletion

Topic

Topic

Topics to be Covered



Topic

Adjust algorithm

Topic

Heapify algorithm

Topic

Counting

Topic

Graph

Topic



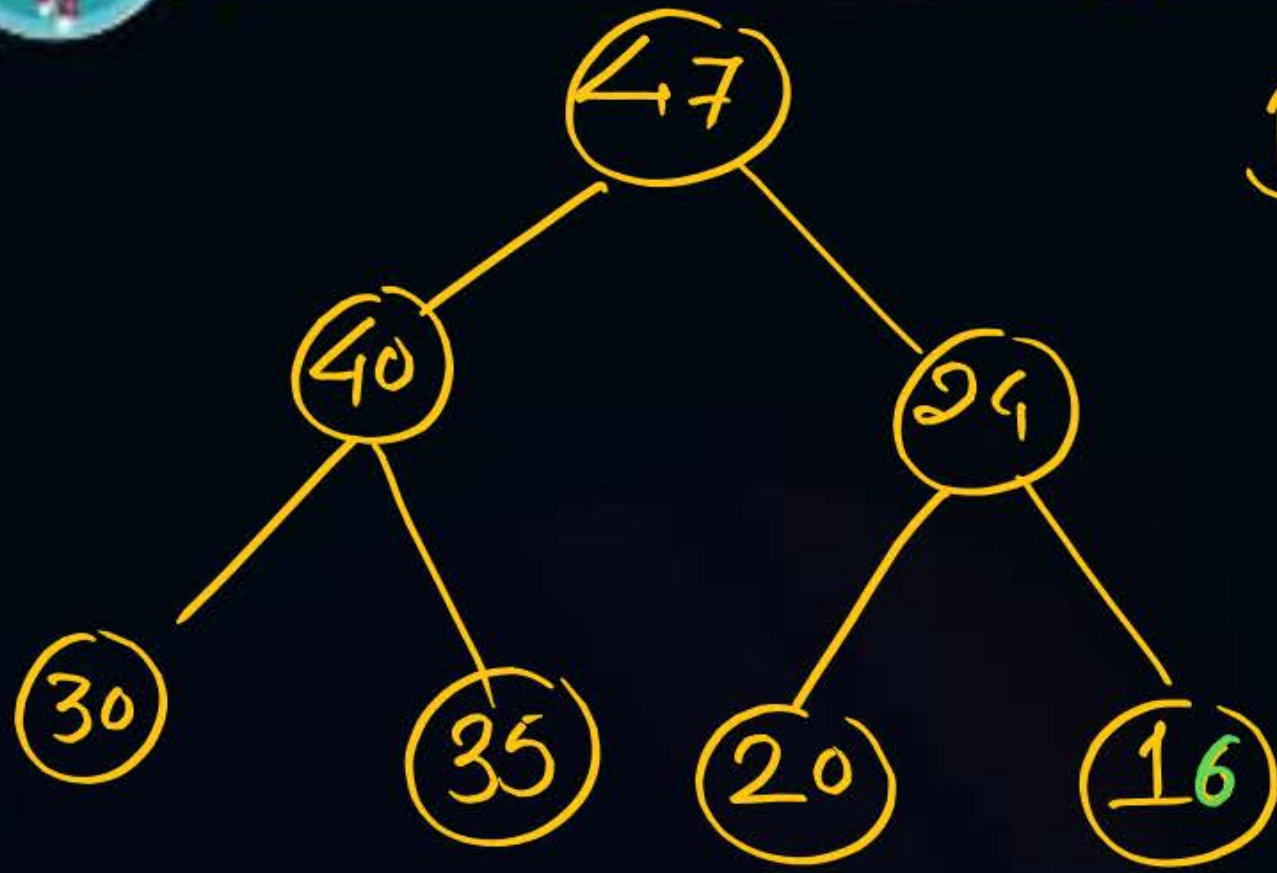
Topic : Tree



Heap adjust algorithm
Counting of heap
Graph representation
Traversal

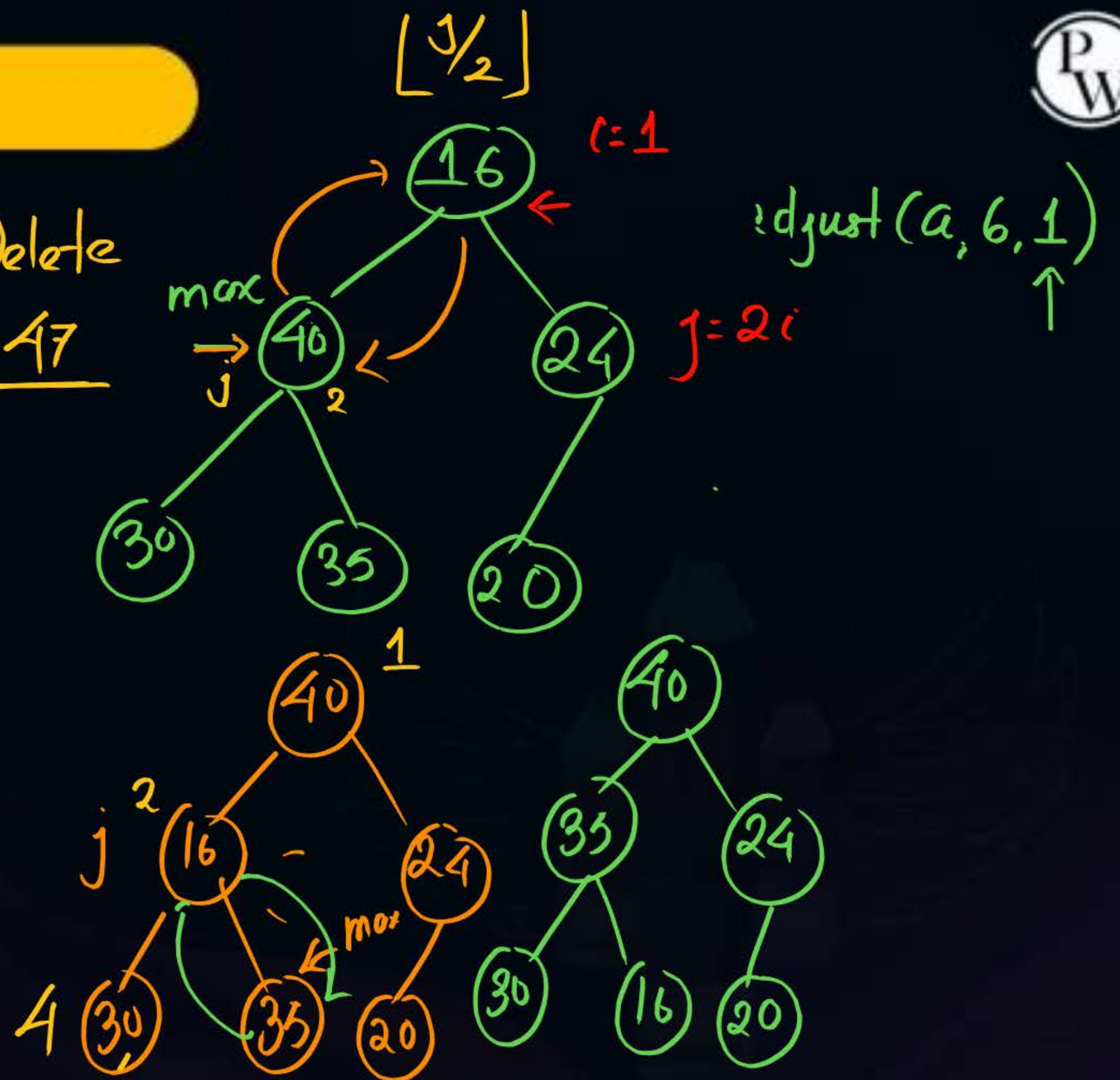


Topic : Heap



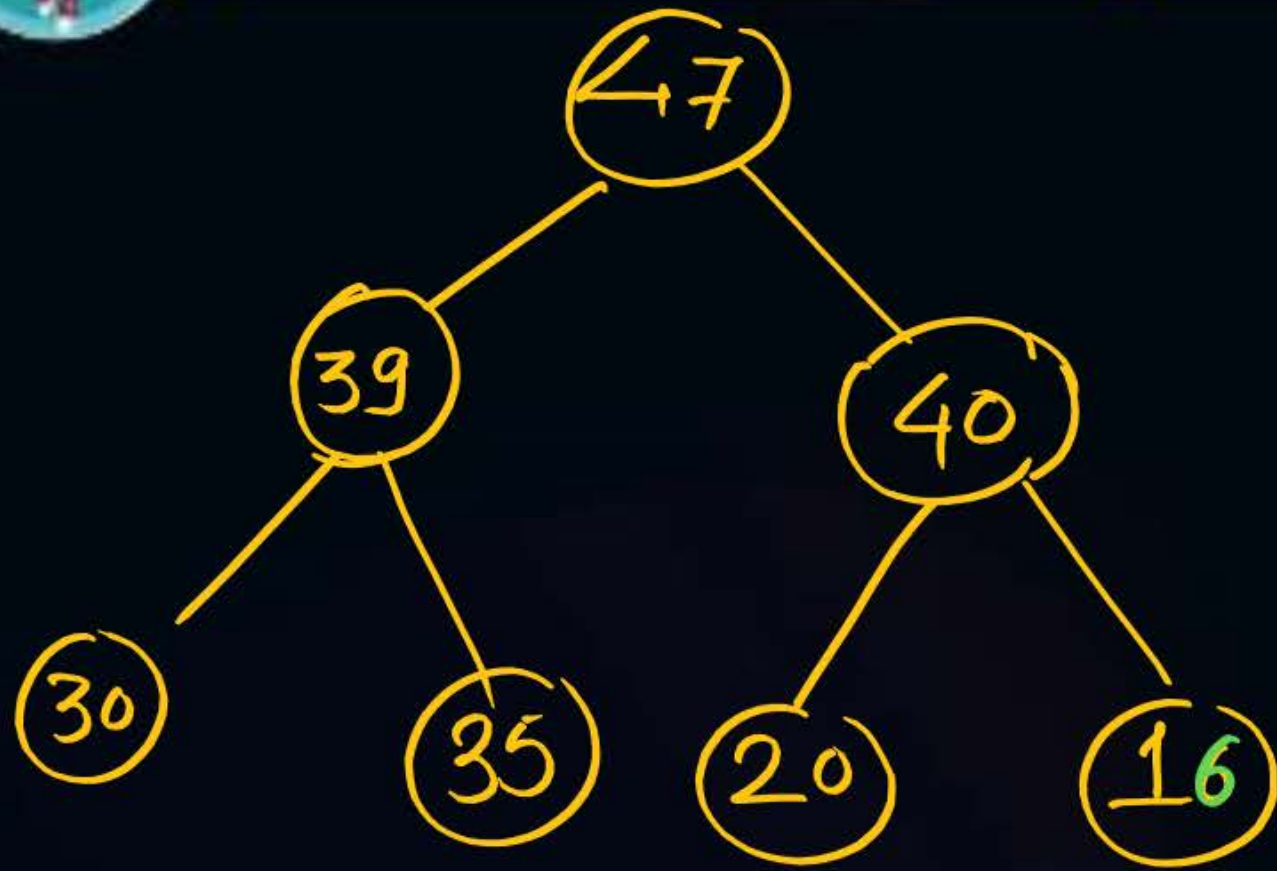
Max heap

Delete
47

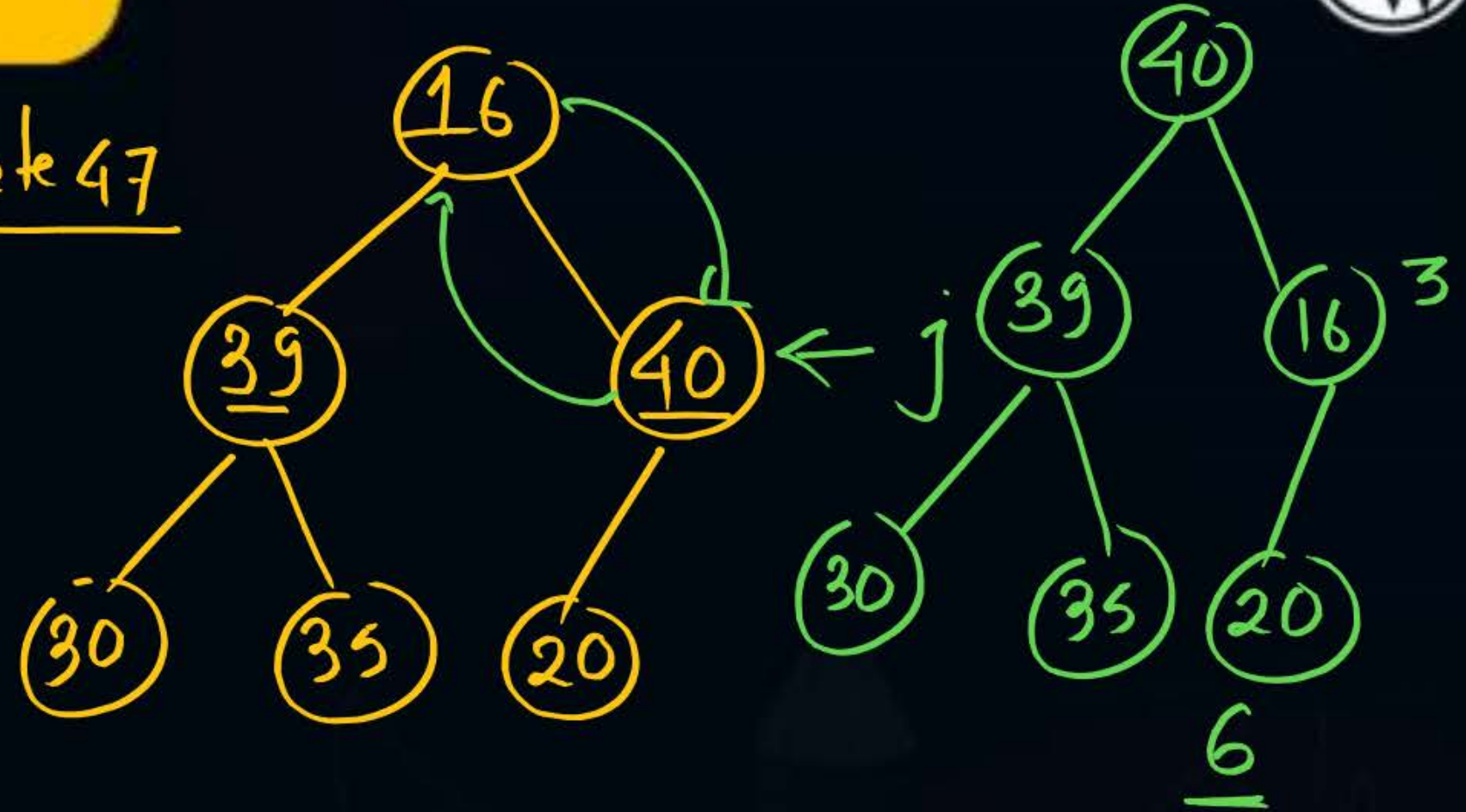




Topic : Heap



Delete 47





Topic : Heap

(Max heap) Algorithm $\text{adjust}(a, n, i) \{$

Left child
is max

$j = \underline{2i}$, $x = a[i]$;

while ($j \leq n$) {

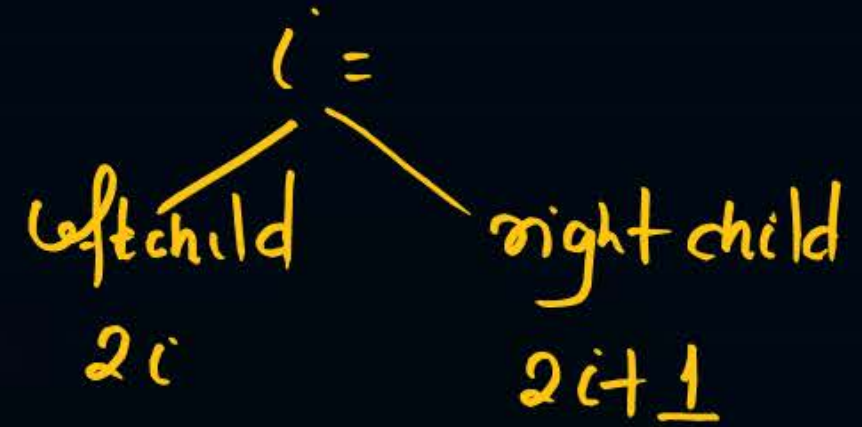
if ($j < n$ and $\overset{\text{left child} < \text{right child}}{a[j] < a[j+1]}$)

$j = j+1$; // max is right child;

if ($x > a[j]$) then break; // value in correct position

Swap ($a[j]$, $a[\lfloor j/2 \rfloor]$); // swap max & parent
 $j = 2j$ update j

}



$O(h)$ CBT $h = \lfloor \log_2 n \rfloor$
 $O(\log_2 n)$

j is last
Index

No right child
exists



Topic : Heap



Adjust algorithm: when for a given $a[i]$ (element at index i) dissatisfy the heap property but left & right subtree is Heap



Topic : Heap



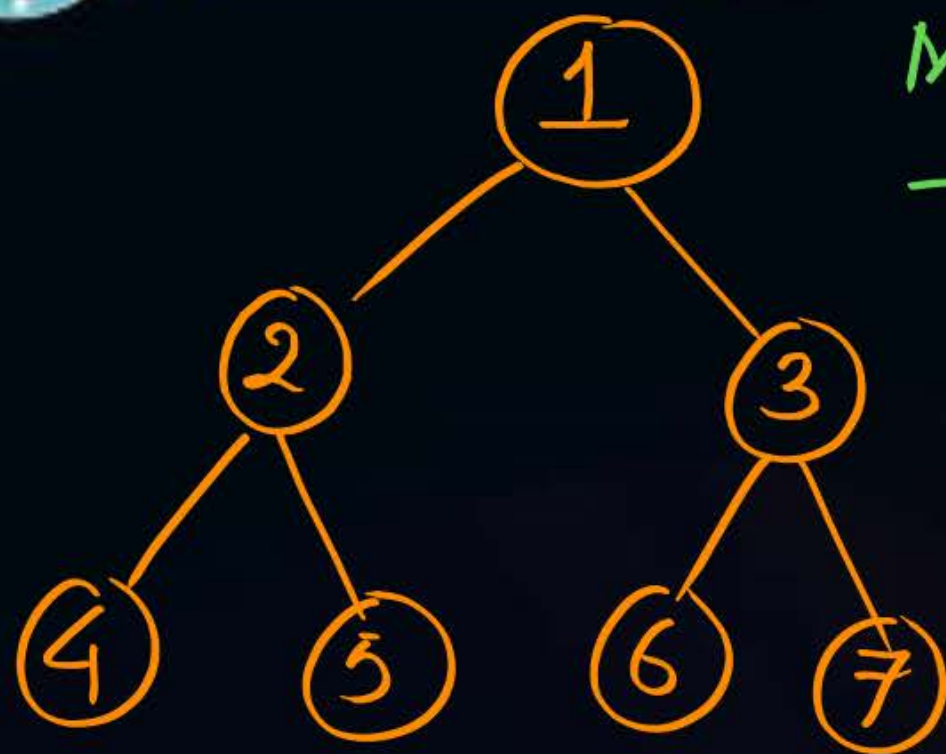
Heapify or Build Heap

Heapify or build heap take any arbitrary array as input & convert it into max/min Heap.

array - BST
AVL
Min Heap



Topic : Heap



Min Heap \rightarrow Max heap

Don't Adjust the Leaf Node

In any CBT with n element No. of
Non-leaf Node.

$$= \left\lfloor \frac{n}{2} \right\rfloor$$



Topic : Heap



Min Heap \rightarrow Max heap

Don't Adjust the Leaf Node

In any CBT with n element No. of
Non-leaf Node.

$$\left\lfloor \frac{n}{2} \right\rfloor$$

Leaf Node : $\left\lceil \frac{n}{2} \right\rceil$

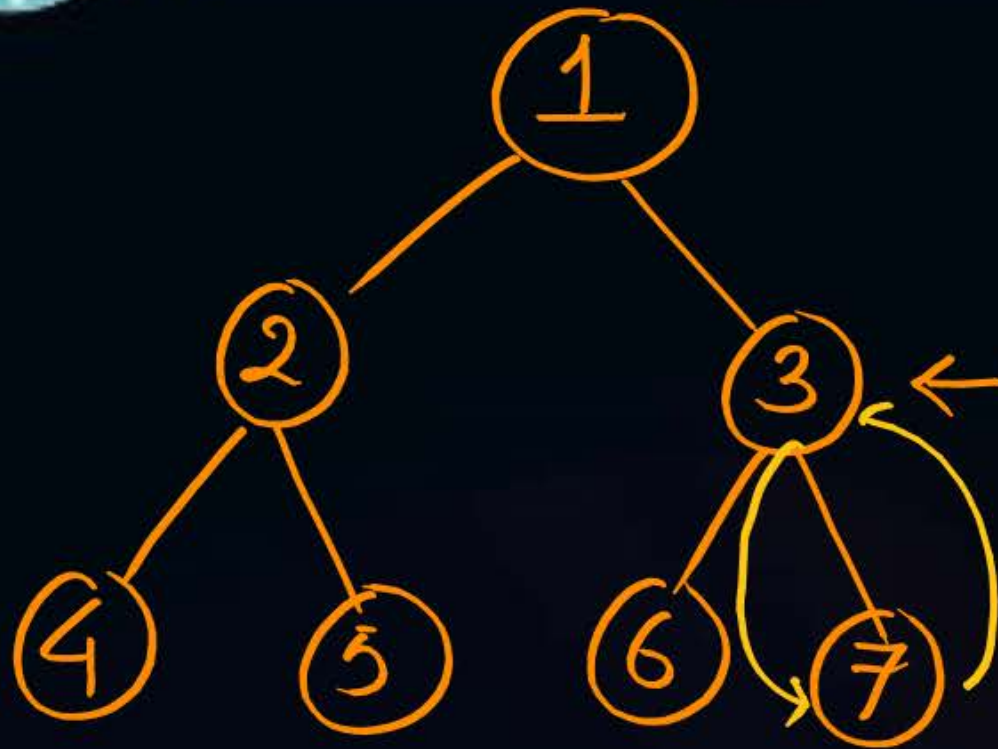


Topic : Heap

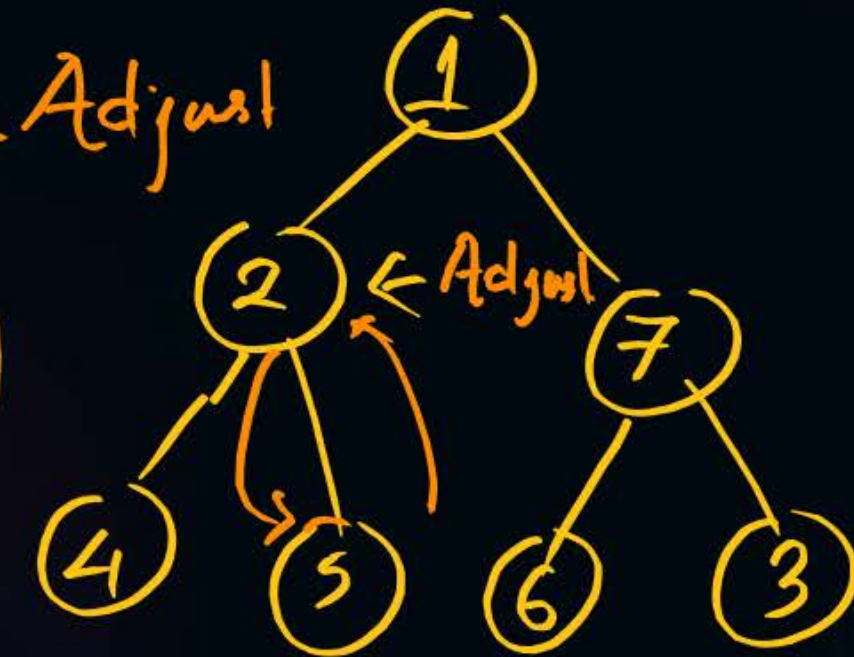


3.

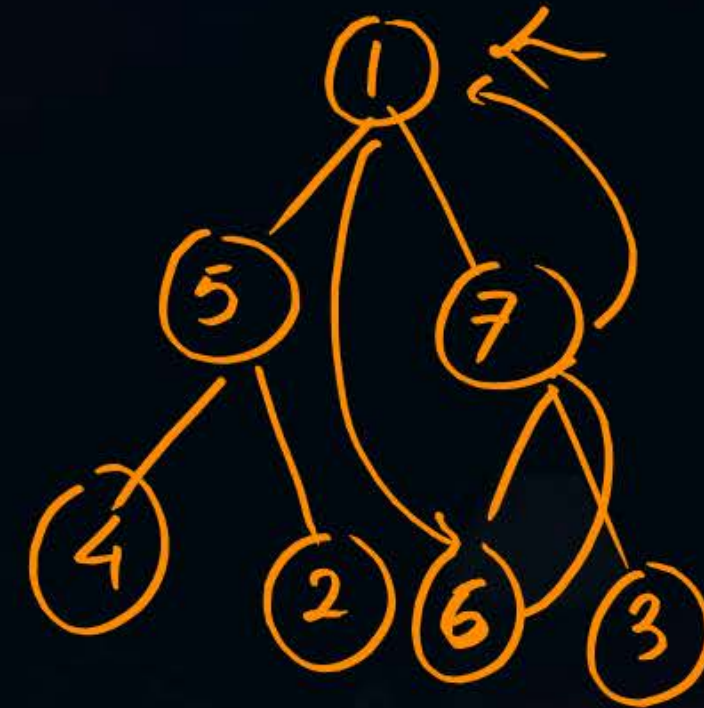
$$i = \lfloor n/2 \rfloor + 1 \text{ to } 1$$



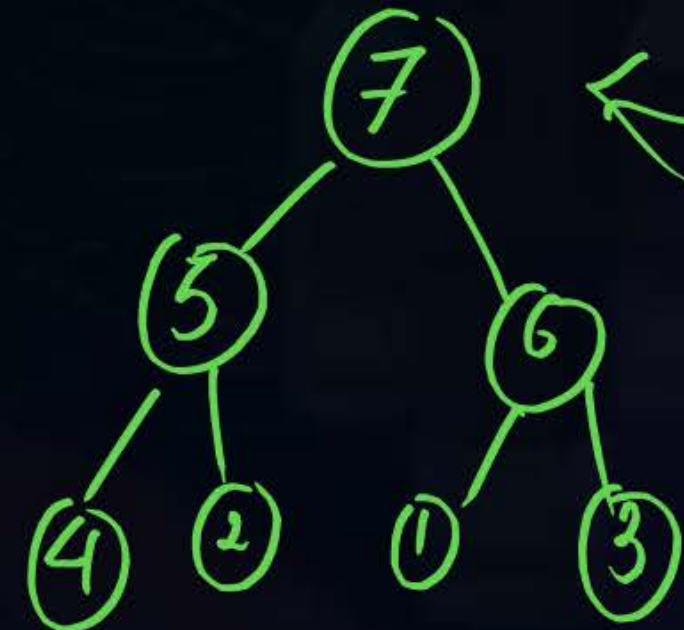
Min Heap



Adjust or Heapify



Max heap





Topic : Heap



Build Heap or Heapify

calls adjust algorithm for every Non leaf

in decremental order

for $i = \lfloor n/2 \rfloor + 1$ to 1

adjust(a, n, i),

Complexity = ??
 $O(n)$

why: Later.



Topic : Heap



Counting of max/min Heap

$n = 1, 1$

No. of min Heap with 1, 2



1 draw CBT
Structure

$T(n)$ is No. of
max/min Heap then

$$T(1) = 1 \quad T(2) = 1$$



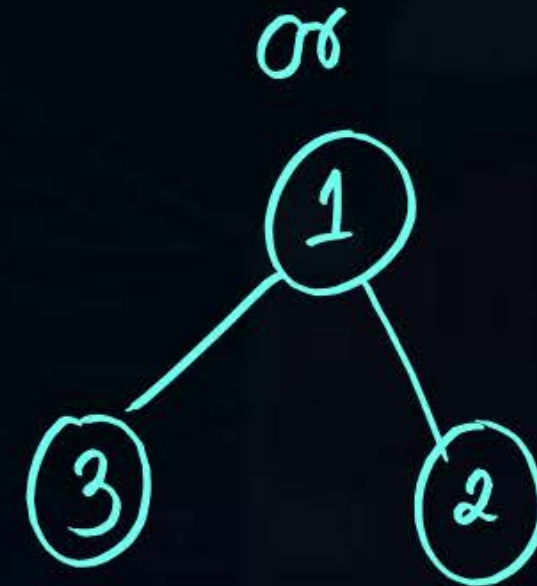
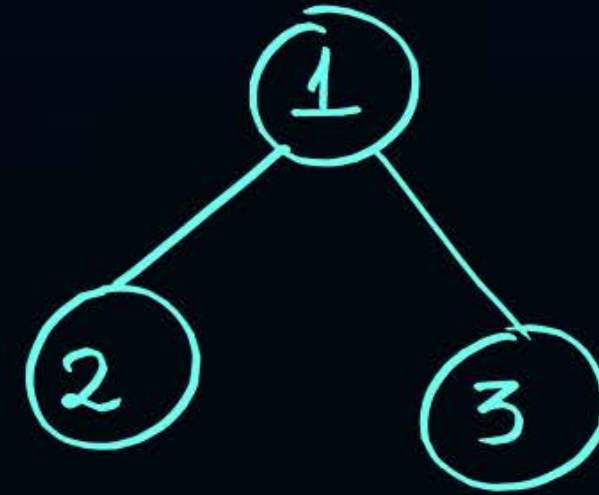
Topic : Heap



Counting:

No. of max/min with 1, 2, 3

$$T(3) = 2$$





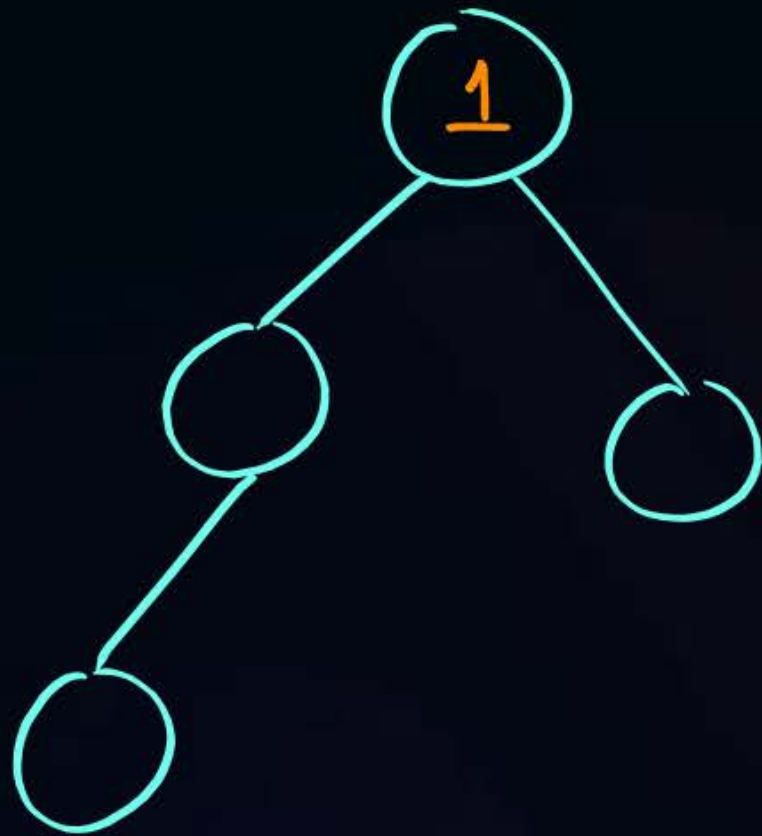
Topic : Heap



$$T(4) = \underline{3}$$

No. of min Heap with key, 1, 2, 3, 4

1. first CBT structure



$${}^3C_2 = 3 \text{ ways}$$

L

2, 3

R

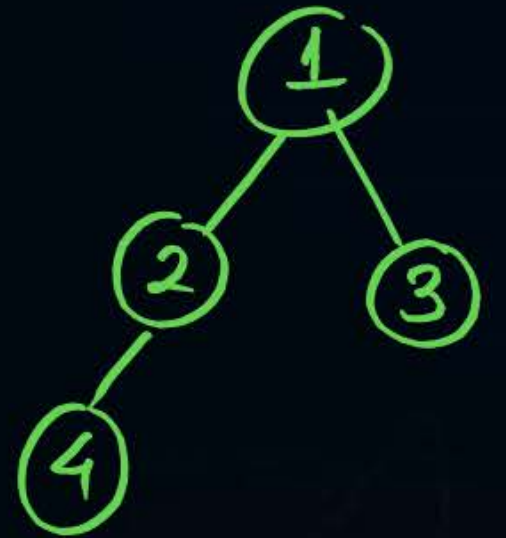
4

2, 4

3

3, 4

2





Topic : _____



1. Draw CBT structure
2. Since one element is root & Suppose
Left subtree contains k elements
No. of way we can select k elements
from n-1 elements is : ${}^{n-1}C_k$



Topic : Graph

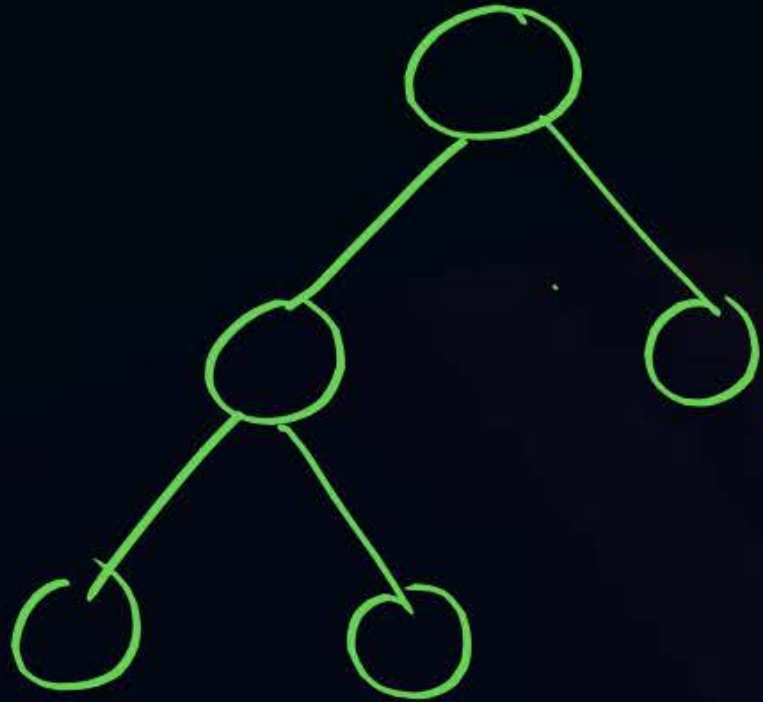
$$T(5) = 8$$

No. of min Heap with keys 1, 2, 3, 4, 5

$$K=3$$
$$n=5$$

No. of ways we can select 3 elements

$$\text{is } {}^4C_3 = 4$$



L

2, 3, 4

2, 3, 5

2, 4, 5

3, 4, 5

R

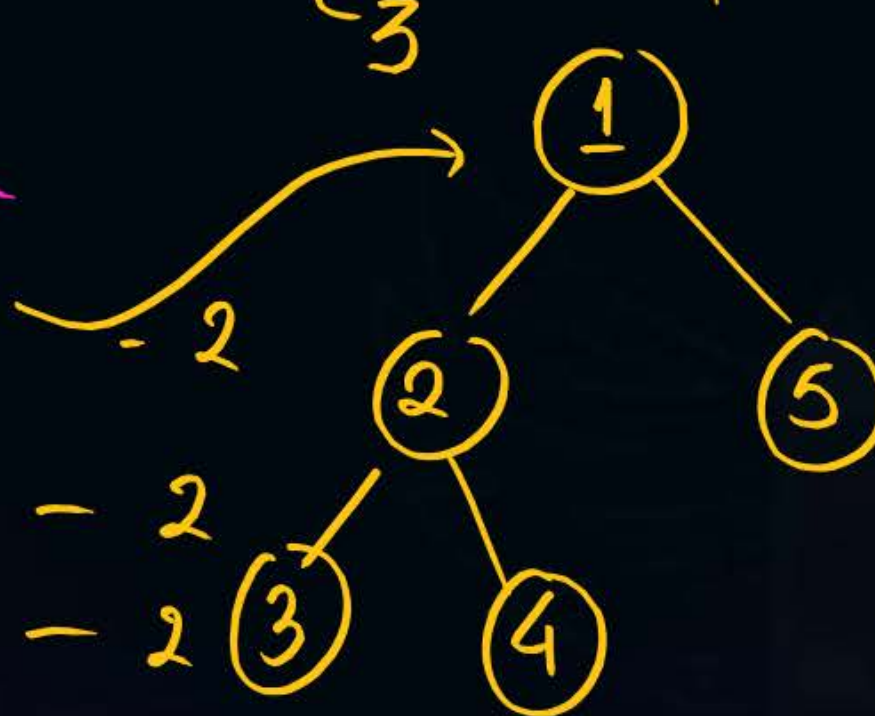
5 - 2

4 - 2

3 - 2

2 - 2

8





Topic : Heap



No. of Heap (Max | Min)

$n = 6$

$$T(n) = \sum_{k=0}^{n-1} T(k) * T(n-1-k)$$

k is No. of element in Left subtree of CBT



Topic : Heap



```
Algorithm Adjust(a,i,n){  
  j := 2i; x:= a[i];  
  while (j ≤ n) do {  
    if ((j < n) and (a[j] < a[j + 1]))  
      then j := j + 1;  
    if (item ≥ a[j]) then break;  
    swap(a[j/2 ] := a[j]);  
    j := 2j;  
  }  
}
```




Topic : Heap

Algorithm DelMax(a,n,x)

```
{  
    if (n = 0) then{  
        write ("heap is empty"); return false;  
    }  
    x := a[1];  
    a[1] := a[n];  
    Adjust(a, 1, n - 1);  
    return true;  
}
```



Topic : Heap



```
Algorithm buildheap (a,n){  
    for i =  $\lfloor n/2 \rfloor$  to 1 step -1 do  
        Adjust(a,i,n)  
}
```




Topic : Heap



Suppose a binary search tree with 37 distinct elements is also a complete binary tree. The tree is stored using the array representation of binary heap trees. Assuming that the array indices start with the 1 largest element of the tree is stored at index X and smallest element is stored at Index y then $X+Y$ is _____.

HW



Topic : Heap

Consider a binary min-heap containing 169 distinct elements. Number of leaf present in heap is

GATE
2024

$$\left\lceil \frac{169}{2} \right\rceil = \lceil 84.5 \rceil = 85 \text{ leaves}$$

$$169 - 85 = \underline{\underline{84}}$$

Non leaf
Node

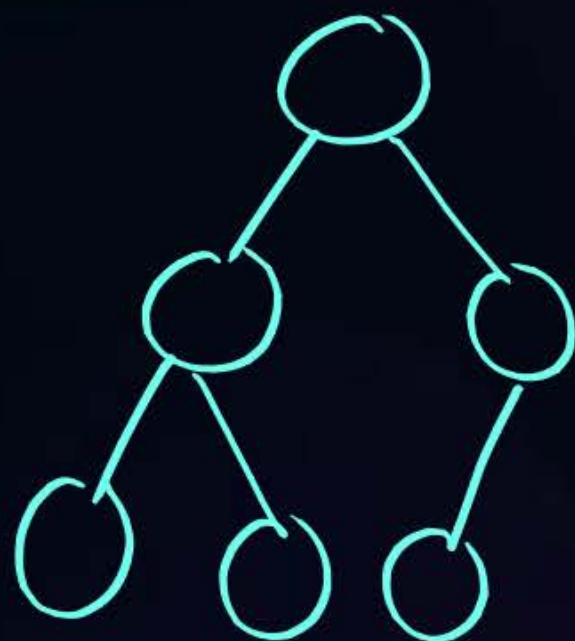


Topic : Heap

Number of possible with max heap with 1,2,3,4,5,6 is

$$n=6$$

$$T(n) = \sum_{k=0}^{n-1} T(k) * T(n-1-k)$$



$$k=3 : 5C_3 * T(3) * T(2)$$

$$= \frac{5!}{2! 3!} \times 2 \times 1 = \frac{5 \times 4^2}{2 \times 1} \times 2 \times 1$$

(20)



Question: Heapify

An array [82, 101, 90, 11, 111, 75, 33, 131, 44, 93] is heapified. Which one of the following options represents the first three elements in the heapified array?

- (A) 82, 90, 101
- (B) 82, 11, 93
- (C) 131, 11, 93
- (D) 131, 111, 90

Max heapify

H.W

No. of swap in adjust
Operation

Min Heap

first element 11



2 mins Summary



Topic

Heap adjust

Topic

Heapify

Topic

Counting

Topic

Topic

THANK - YOU