Segment Tree

Concepts & Qns...

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

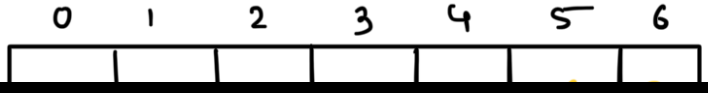
codestorywithMIK →

"No more fear of Segment Tree"

video - 1

Segment Tree ???

जो भी है,
डरना नहीं है।



n = 7

index: (1,4) → Sum →

index: (2,5) → Sum →

index: (0,5) → sum →

$O(Q * n)$

large $n \rightarrow 10^6$

$Q = 10^4$

$O(10^4 * 10^6)$

An efficient data structure that allows

⇒ Efficient Querying of intervals/range

⇒ Efficient updating of intervals/range

range queries to find

↳ Sum ✓

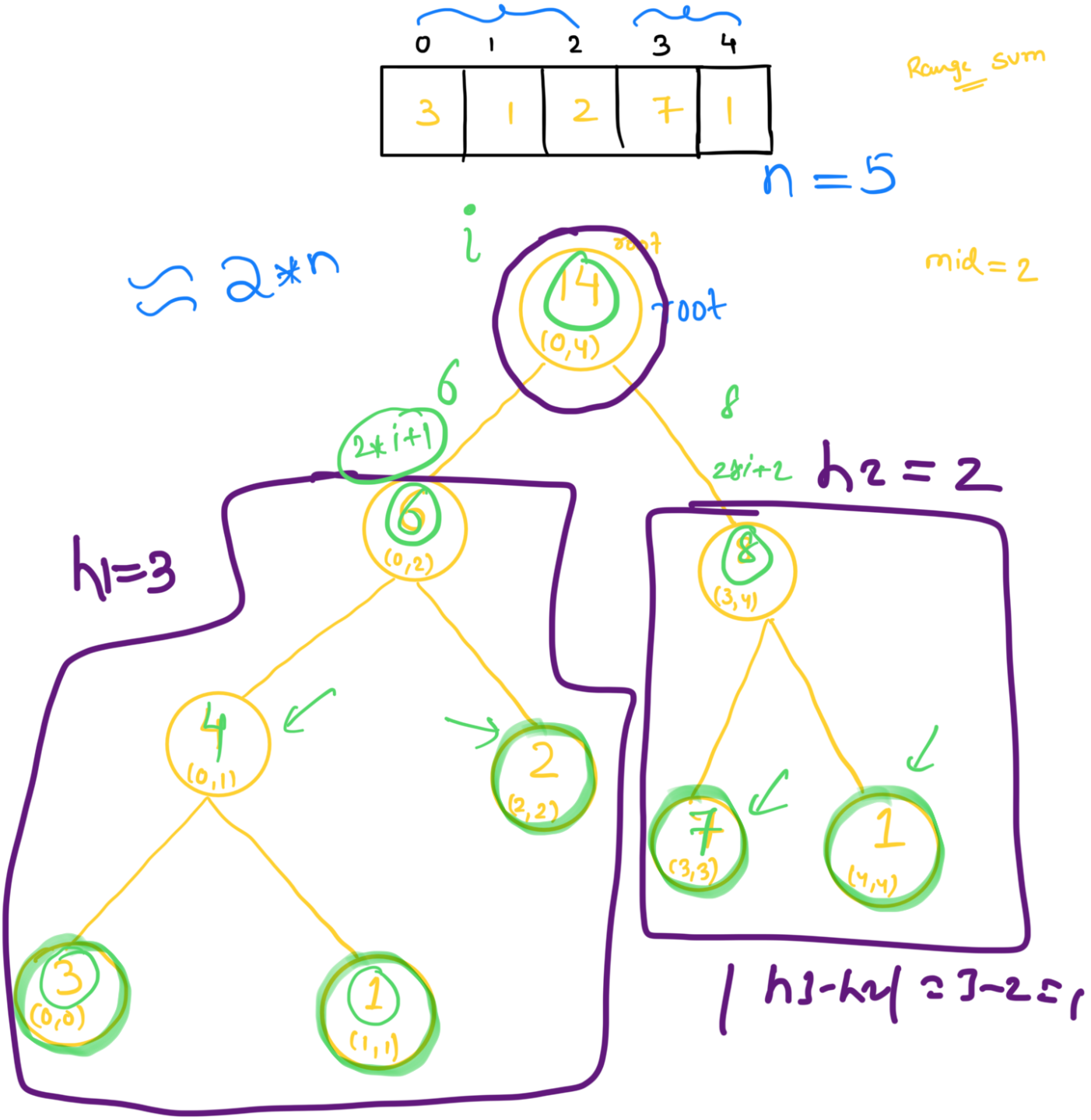
↳ minimum ✓

↳ maximum ✓

etc....

Segment Tree

How to Build Segment Tree??



1. Binary Tree
2. 2 children of all non-leaf nodes.
3. → Leaf Nodes – Represents a single element in an array.

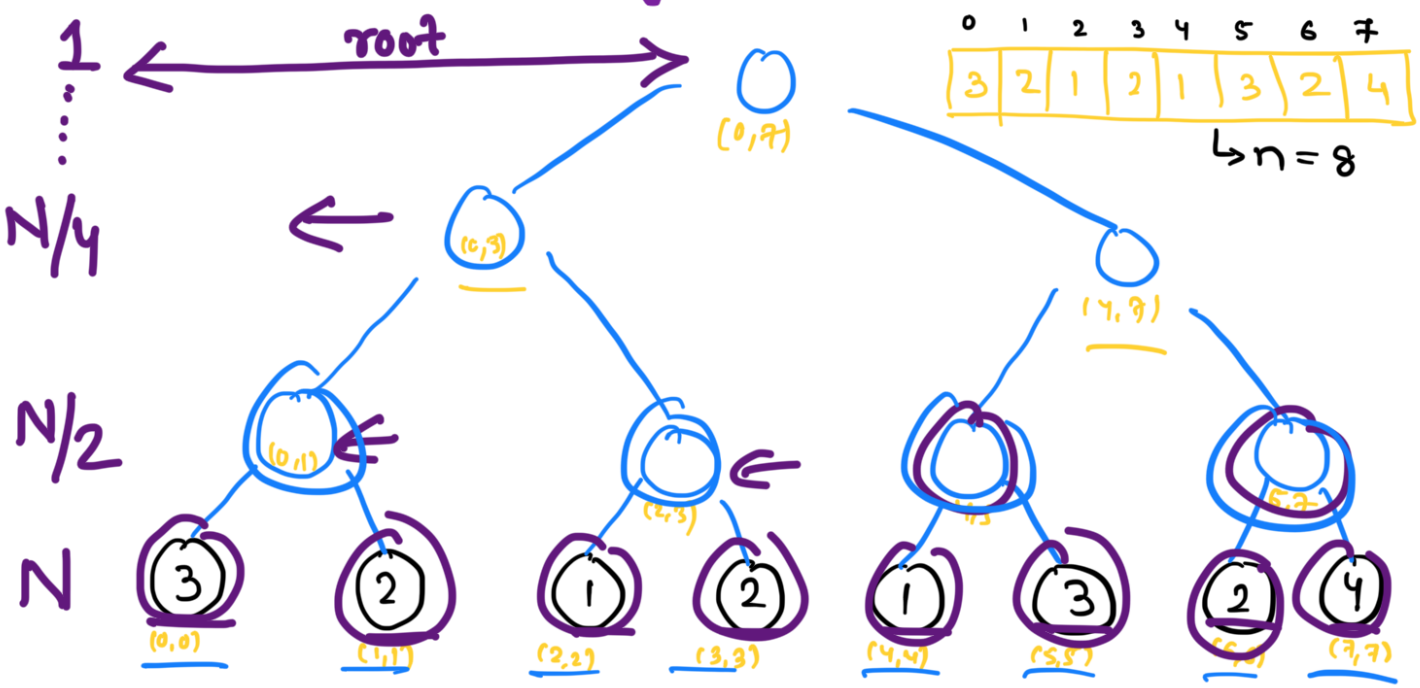
→ Root Node – Represents entire array.

→ Other nodes - Represents an interval or range of an array.

4. Height $= \lceil \log_2(n) \rceil$

* 5. Balanced Binary Tree.

How many nodes ???



nodes = N + N/2 + N/4 + ... + 1

↳ G.P.

$$Sum_\infty = \frac{a}{1-r} = \frac{N}{1-\frac{1}{2}} = (2 * N)$$

leaf = N

internal = (N-1) } N+N-1 = (2N-1)

≈ 2N

Why I am not using Prefix Sum Array???

nums =

0	1	2	3	4
1	3	4	2	2

(1, 4)

PrefixSum =

0	1	2	3	4
1	4	8	10	12

12 - 1

= 11 ✗

① range sum

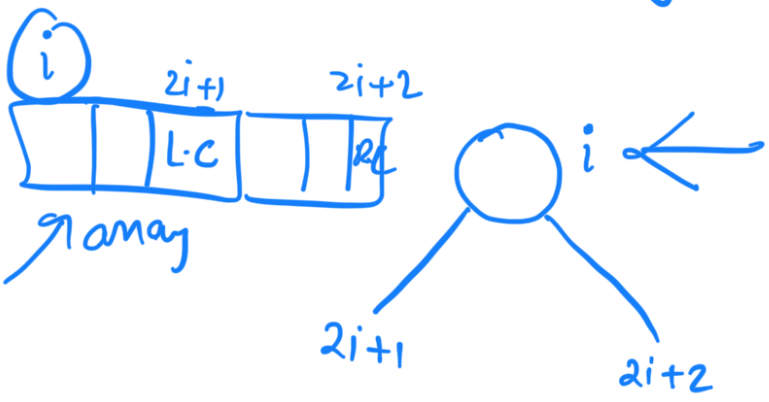
② range max-elem.

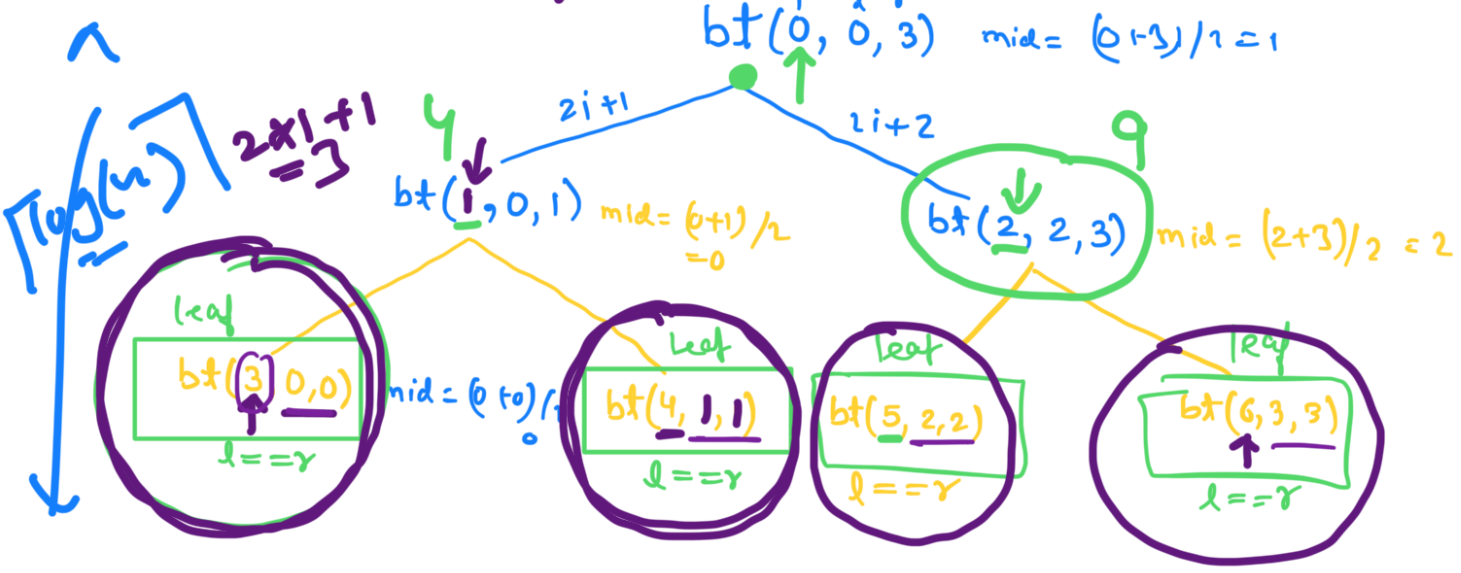
③ " min

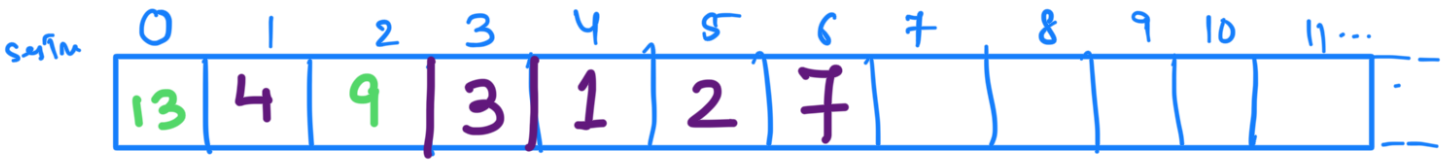
How to actually build the Segment tree in code:-

0	1	2	3

Tree ⇒ as an array represent







n

(n-1)

n + n-1

= 2n-1

2*n

= 8-1 = 7

Story to Code :-

buildTree (0 , l , r) ;
(i = 0, l = 0, r = n-1)

buildTree (i , l , r) {

// Base Case

if (l == r) {

segTree[i] = nums[r] ;

return ;

}

int mid = (l+r)/2 ;

buildTree (2*i+1 , l , mid) ;

buildTree (2*i+2 , mid+1 , r) ;

} Recursion Leap of Faith

segTree[i] = segTree[2*i+1] + segTree[2*i+2] ;

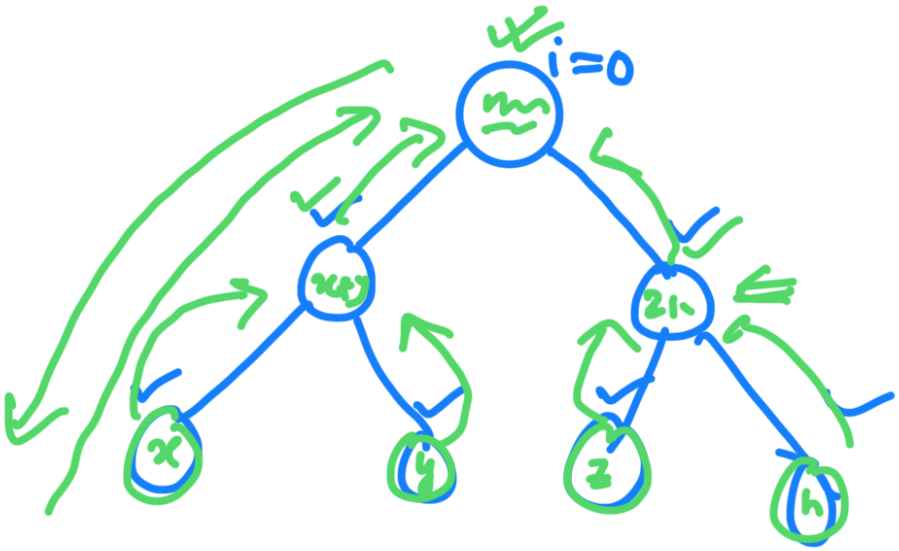
}

What will be the size of segtree array ???

2*n.

Time Complexity

S.C = O(2n)



Visiting all nodes twice.

$\simeq \frac{1}{2} * (2 * n)$

$\simeq O(n)$

T.C of building S.T = $O(n)$.

↳ what is segment Tree

↳ Build Segment Tree.

↳ Update Segment Tree ✓

↳ Query Segment Tree ✓

(l, r) = Sum

(l, r) ...

- Range Sum Query.
- Range Max / Min / Prod