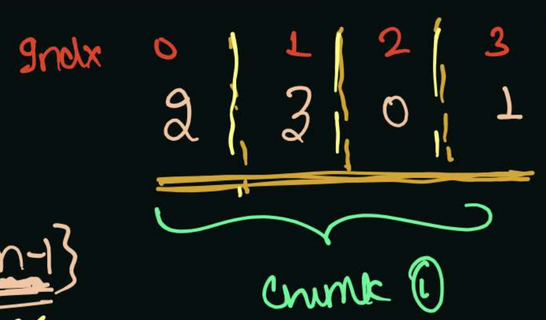
Max Chunks To Make Array Sorted 1

Saturday, 4 September 2021 9:59 AM

arr → size → n.

Elements → {0, n-1}

Chaining technique →

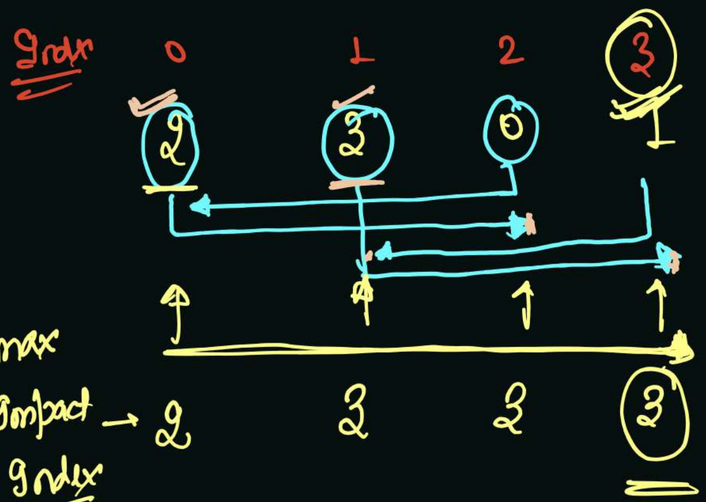


arr[i] = {0, 1, 2, ... N-1}

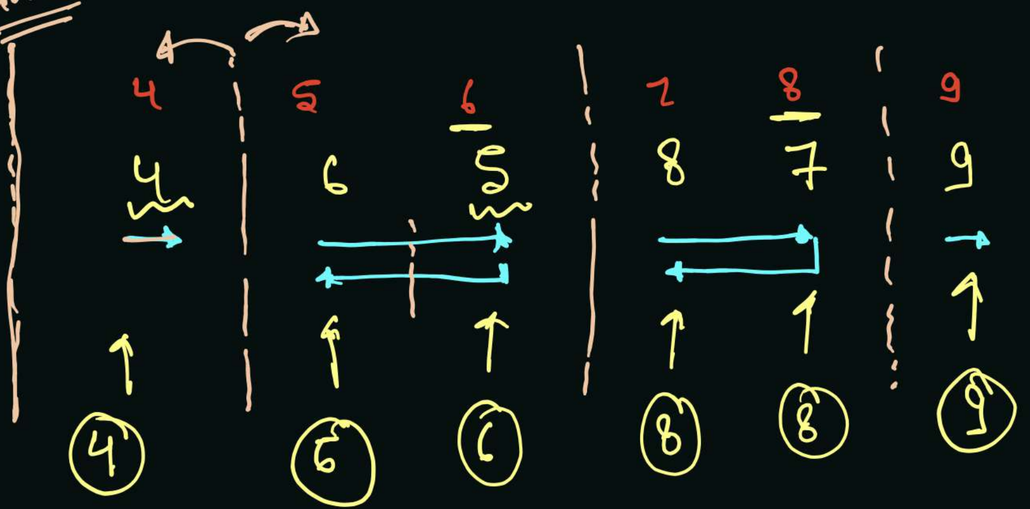
max. no. of chunks so that if we sort individuals chunks, the whole array will sorted.

NOTE: Rearrange of chunks are not allowed

size of chunk doesn't matter.



Partition.



Partition is possible → chunk++;

Chunk = 0 1 2 3 4 5 ⇒

min chunks →
- one → take complete array as chunk

max chunks →
- size of array.
- sorted array.

0	1	2	3	4	5	6	7	8	9
9	3	6	2	1	7	5	4	0	8
9	9	8	9	9	9	9	9	9	9

0 → N-1

maxSoFar = 0

chunk = ~~0~~ 1

```
if(arr[i] > max)
    max = arr[i]
max = Math.max(max, arr[i]);
if(max == i) chunk++;
```

Steps:

1. ~~Maximise~~ max impact index.
2. ~~Check~~ if it is equal to current index, if equals chunk++.

0	1	2	3	4	5
2	3	4	5	0	1
2	3	4	5	5	5

max impact index

chunk = ~~0~~ ~~1~~ ~~2~~ 3

6	7	8	9
7	6	9	8
7	7	9	9

0	1	2	3	4
4	3	2	1	0
4	4	4	4	4

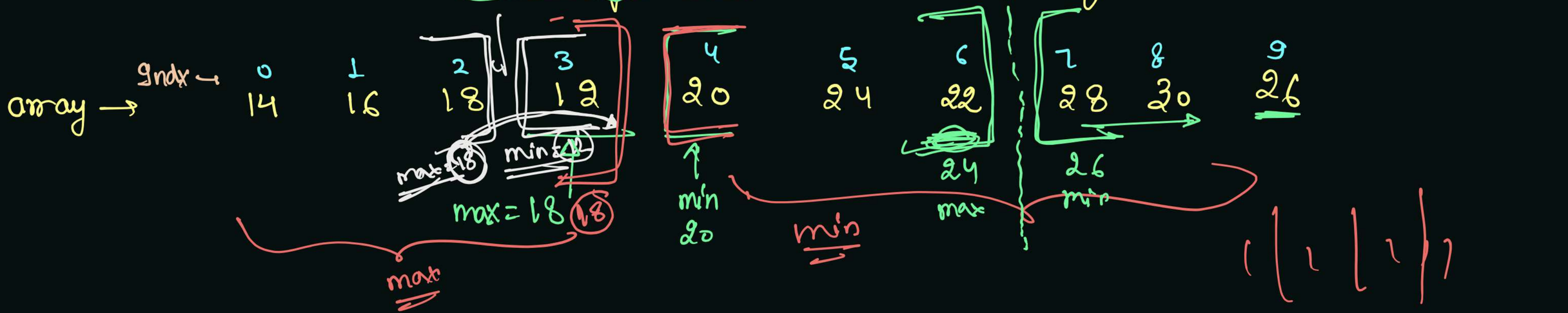
Single chunk.

chunk = ~~0~~ 1

Max Chunks To Make Array Sorted 2

Saturday, 4 September 2021 9:59 AM

Max. possible chunks, such that if we sort individually chunks without rearrange them, the whole array will be sorted.



partition is possible when:

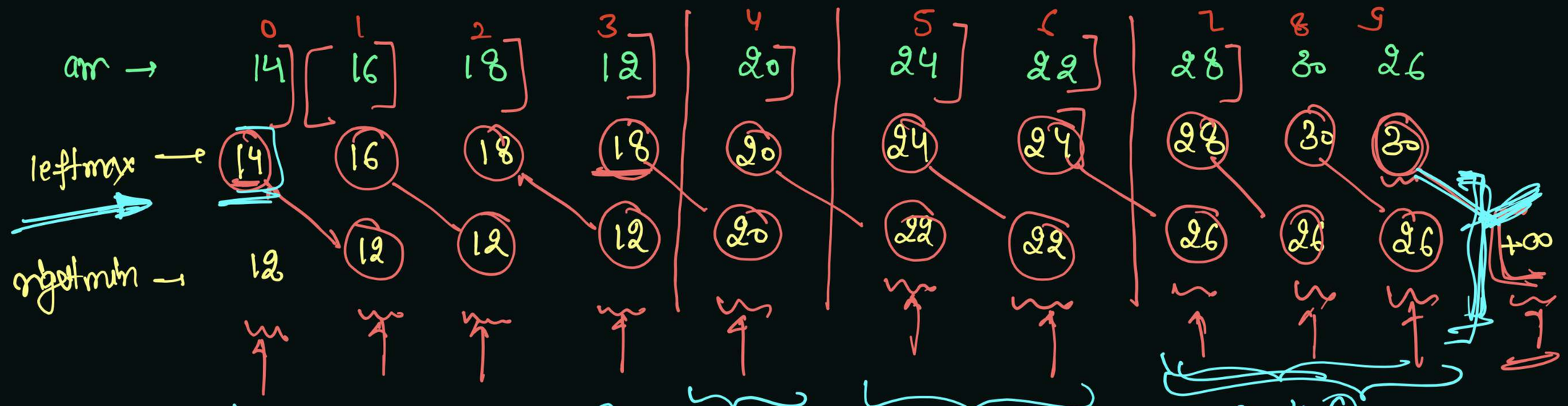
① left max is smaller than right min

leftmax[i] <= rightmin[i+1]

chunks++

Hint → ① max of left part
② min of right part
and figure out the relation.

PreRequisite → [leftmax, Rightmin] of array



leftmax[i] → max value from index 0 to i

rightmin[i] → min value from index i to n-1

leftmax[i] <= rightmin[i+1] → chunk++

Situation

chunk = 0 1 2 3 ④

Steps - ①

① Prepare Rightmin

② Travel and maya a variable for leftmax

③ Increase chunk count on valid situation


14 16 18 12 20 18 24 22 28 30 26
leftmax →
rightmin
12 12 12 12 18 18 22 22 26 26 26
12 14 16 18 18 20 22 24 26 28 30
chunk = 1 2 3 4

Reverse Vowels of a String

Saturday, 4 September 2021 12:24 PM

String → k l a m a b e r o u s a t i u o m a z

Input →

Vowels with order → a a e o u a i u o a

output → k l a m o b u r i a s u t o e a m a z

Index → 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Character array → k l a m a b e r o u s a t i u o m a z

left → →

right ← ←

while (left < right)

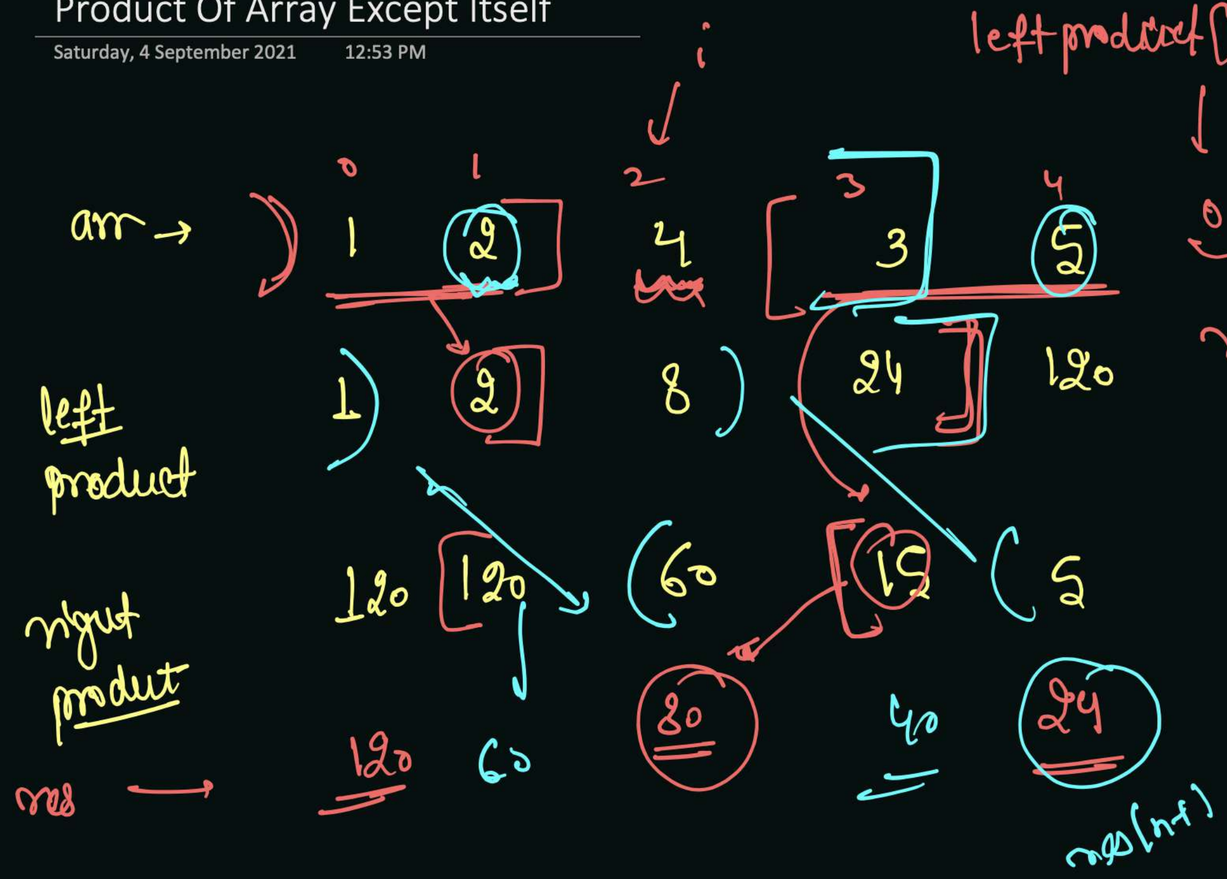
l

l5 l4 l3 l2 r4

→ klamobur iasutoe amaz

Product Of Array Except Itself

Saturday, 4 September 2021 12:53 PM



① Product of all Number

② To fill res[i] = prod / arr[i]] x

x → arr[i] contain 0

	a	b	c	d	0	prod = a*b*c*d
res	b*c*d	a*c*d	a*b*d	a*b*c	...	prod = 0
prod / arr[i]	0 / a	0 / b	0 / c	0 / d	0 / 0	x
	0	0	0	0	x	

left product[i] → product from elements of index 0 to i

right product[i] → product from elements of index i to n-1

Wiggle Sort 1

Saturday, 4 September 2021 12:54 PM

time → O(n)

space → ~~X~~

arr →

arrange value of array in zig-zag fm

arr[0] ≤ arr[1] ≥ arr[2] ≤ arr[3] ≥ arr[4] ... arr[n-1]

TRY IT

wiggle sort 2

arr →

arrange valu of array in zig-zag form

arr[0] < arr[1] > arr[2] < arr[3] > arr[4] ... arr[n-1]

TRY IT

Evening class → 6:30