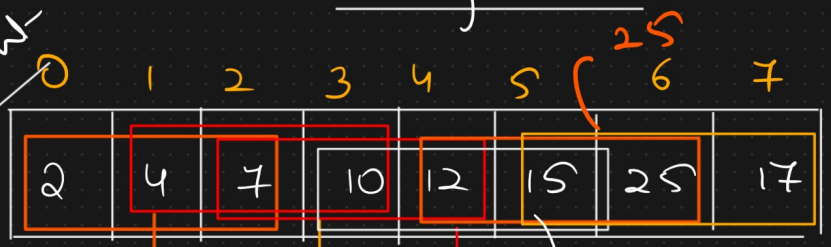
Sliding Window

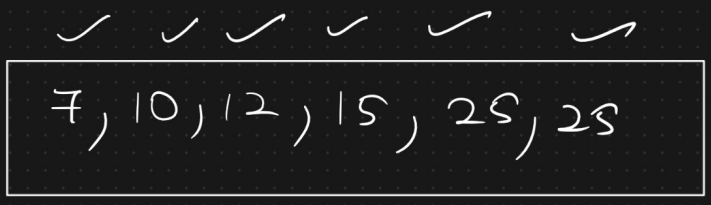
Problem Statement-



$n = 8$

$k = 3$

Left side
↳ one element
deletion



Right side
↳ one element
add

$n - k + 1$

$= 8 - 3 + 1 = 6$

1. 2, 4, 7 ——— 7
2. 4, 7, 10 ——— 10
3. 7, 10, 12 ——— 12
4. 10, 12, 15 ——— 15
5. 12, 15, 25 ——— 25
6. 15, 25, 17 ——— 25

Brute force approach

```
for (i = 0 to n) {          // N
    for (J = i to i+k-1) {  // K   (0+3-1 = 2)
        Logic to find max val in that window
        maxVal = max(arr[J], max)
    }
}
```

Time complexity

↳ $O(N*K)$

↳ $O(N)$

Sliding window algorithm using the operations of Deque.

Research part

Live Session