Ques Merge sort

i

1	3	9	11	22	27	30

j

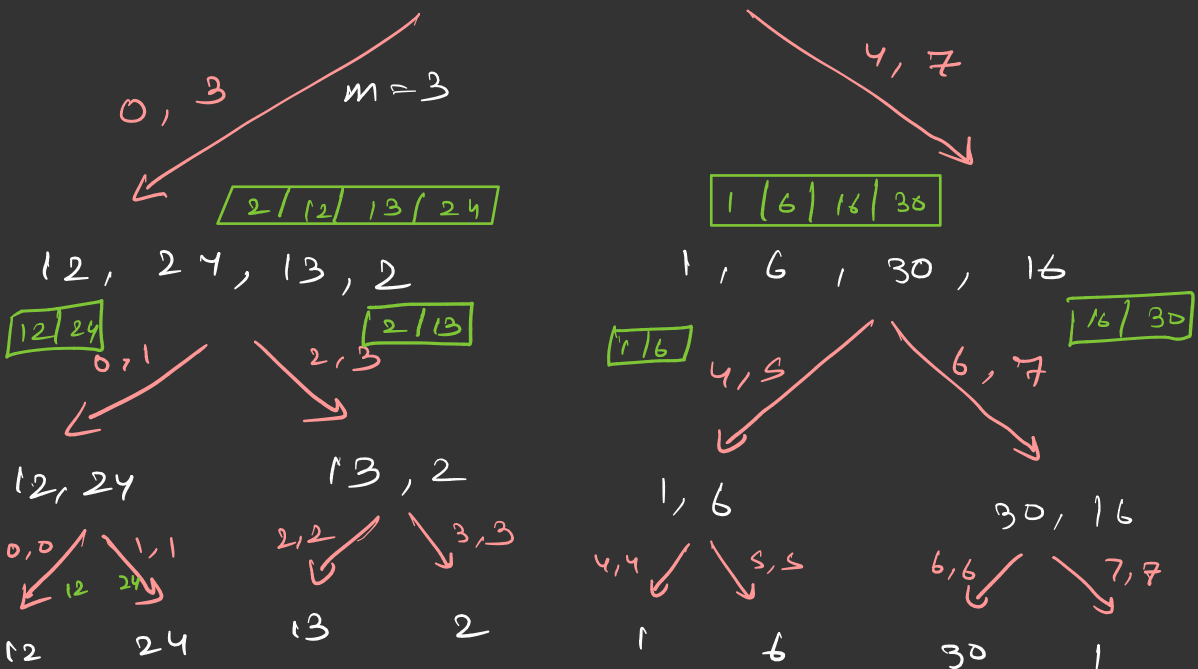
2	4	5	6	24

0	1	2	3	4	5	6	7	8	9	10	11
1	2	3	4	5	6	9	11	22	24	27	30

k

$m = l + r / 2$

0	1	2	3	4	5	6	7
12	24	13	2	1	6	30	16



no of levels = k

$TC = O(n \times k)$

Level	⇒	Size	⇒	
1		$n/2$	⇒	$n/2^1$
2		$n/4$	⇒	$n/2^2$
3		$n/8$		$n/2^3$
k		1		$n/2^k$

$$\frac{n}{2^k} = 1$$

$$n = 2^k$$

$$\log_2 n = \log(2^k)$$

$$\Rightarrow k \log_2(2)$$

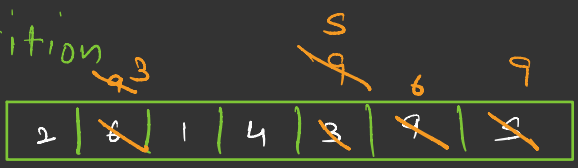
$$\log n \Rightarrow k$$

$$TC \Rightarrow O(n \log_2 n)$$

Quick Sort

pivot = 5

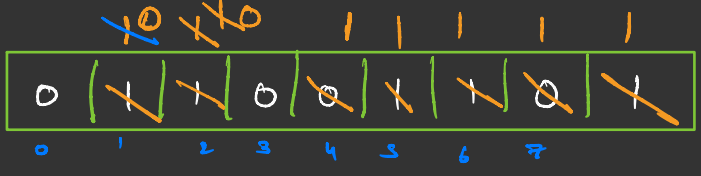
Partition



1. Put it in sorted position
2. All the elements on left should be smaller and all on right should be greater

---

Ques 2 way sorting, Sort 0 and 1, Sort odd even



R1

j

i

0 - (i-1) ⇒ 0's → Region 1

i - j ⇒ Unknowns → 2

(j+1) → n-1 ⇒ 1's → 3

```
Quick Sort (int arr[], int l, int r) {
    int pi = partition(arr, l, r);
    quickSort(arr, l, pi-1);
    quickSort(arr, pi+1, r)
}

int partition (int a[]; int l, int r) {
    int pivot = a[r];
    int i = l; j = r-1;
    while( i <= j) {
        if (a[i] < pivot) i++;
        else { swap (a, i, j);
               j--;
        }
    }
    swap(a, i, r);
    return i;
}
```