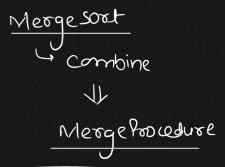


Quick Soot



$$(mid = l + (n-l)/2)$$

h =5

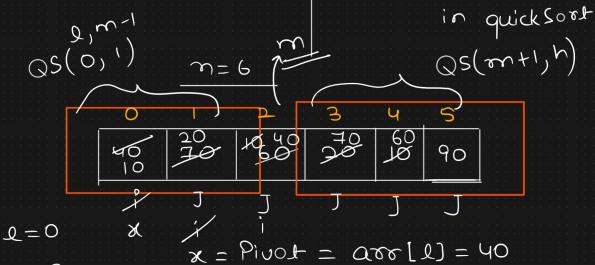
Quicksoot

- Partition

Divide the array

into two subarrays

4 No combine required



 $i \rightarrow +6$ take smaller elements

than pivot

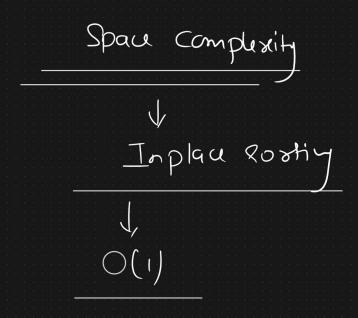
J- to take larger element

Partition (arr, l, h) & i = QPluot = arr(l); for (j= 2+1; j <= n; j++)d if (arr(j) x = pivot) d i= 1+1% swap (ar (i), arc()) Swap (arr (2), arr(i)) correct _ index of Pivot element Jehn ij Mote

1. Pivot Element seacher to its correct position
2. Left 21de of Pivot - Smaller element
Right 21de of Pivot - bigger element

Quick Sout Algorithm quickSort (arr, l, h) d if (Q < n) d 1. Divide the array m = partition (arr, l,h); 2. Conquer the subproblems Via recusion m-1-2+1 7(m-l) quick Sost (avo, l, m-1); - Recession quid(Sost (arr, m+1)) T(h-m) h - (m+1) + 1h-m-/+/ Recubrence Relation

algorithm Right ()S(0,2) 10 30 40 (50) 70 60 10 (50) 40 60 Bert Care 4 DOUI- Care Average care $T(\eta) = T(\eta - I) + m$ T(n) = T(n/2) + T(n/2) + h= 2T(n/2)+n= 0 (Nogn) Sorked/Almost Sosted 30 QuickSort -> Geniune Rcenario U Highly uneosted



- Not a stable algorithm

10,10, (6,10,10)

Randonnized QuickSort

Pluot element

(Randomly closen)

75 4 6 5 1 3