

Quick Sort

- Now we found a wrong one on the LHS. Lets go to the RHS

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- If $a[j] > x$ then its fine just decrement j.

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- If $a[j] > x$ then its fine just decrement j.

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- If $a[j] > x$ then its fine just decrement j.

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Now we have found a wrong one on the right side

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Switch the two element $a[i]$ & $a[j]$

X=11

10 35 30 11 03 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Switch the two element $a[i]$ & $a[j]$

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Going on again.....

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[i] < x$ means ++i

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Found a wrong one

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[j] > x$ means --j

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[j] > x$ means --j

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Found 2 wrong ones means interchange

X=11

10 03 30 11 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Found 2 wrong ones means interchange

X=11

10 03 11 30 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Going on...

X=11

10 03 11 30 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[i] < x$ NO... so got a wrong one on LHS

X=11

10 03 11 30 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Moving on to RHS

X=11

10 03 11 30 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[j] > x$ means --j

X=11

10 03 11 30 35 67 27

i j

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- $a[j] > x$ NO... stop found a wrong one....
- When i & j converge STOP

X=11

10 03 11 30 35 67 27

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Now we can see all on the left of 11 are less & right more

X=11

10 03 11 30 35 67 27

DATA STRUCTURES@NAVLAKHI™

Quick Sort Programmers View

```

L = 0
R = n-1
X = a[(L+R)/2];
While (i < j)
{
.....
}

```

DATA STRUCTURES@NAVLAKHI™

Quick Sort Programmers View

```

L = 0
R = n-1
X = a[(L+R)/2];
While (i < j)
{
if(a[i] < x) -- i;
if(a[j] > x) -- j;
.....
}

```

DATA STRUCTURES@NAVLAKHI™

Quick Sort Programmers View

```

L = 0
R = n-1
X = a[(L+R)/2];
While (i < j)
{
if(a[i] < x) -- i;
if(a[j] > x) -- j;
if(i < j) /*SWITCH*/
{ t = a[i]; a[i] = a[j]; a[j] = t; }
}

```

DATA STRUCTURES@NAVLAKHI™

Quick Sort

- Now we can repeat this for left & right side

X=11

10 03 11 30 35 67 27

DATA STRUCTURES@NAVLAKHI™

