

What is sorting? → Arrangement acc. to a particular parameter

① $\Rightarrow 1 \ 2 \ 3 \ 4 \ 5$ sorted in asc order

$\Rightarrow 6 \ 5 \ 4 \ 3 \ 2$ sorted in desc order

$\Rightarrow \begin{matrix} 1 & 7 & 2 & 4 & 9 & 6 \\ 1 & 2 & 2 & 3 & 3 & 4 \end{matrix}$

sorted in asc of no of factors

Why sorting? → search → easier

sort the data acc. of marks

Name	Marks
Mohit	0
Aman	100
Sheet	45
Kajal	32
Yukti	67
Aayush	45
Adilya	92

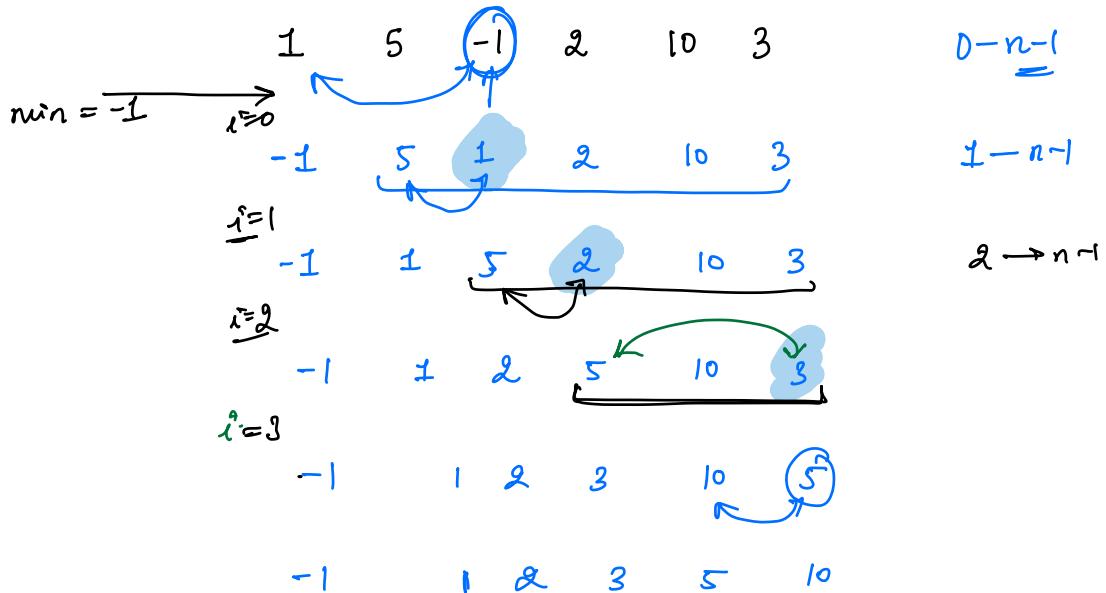
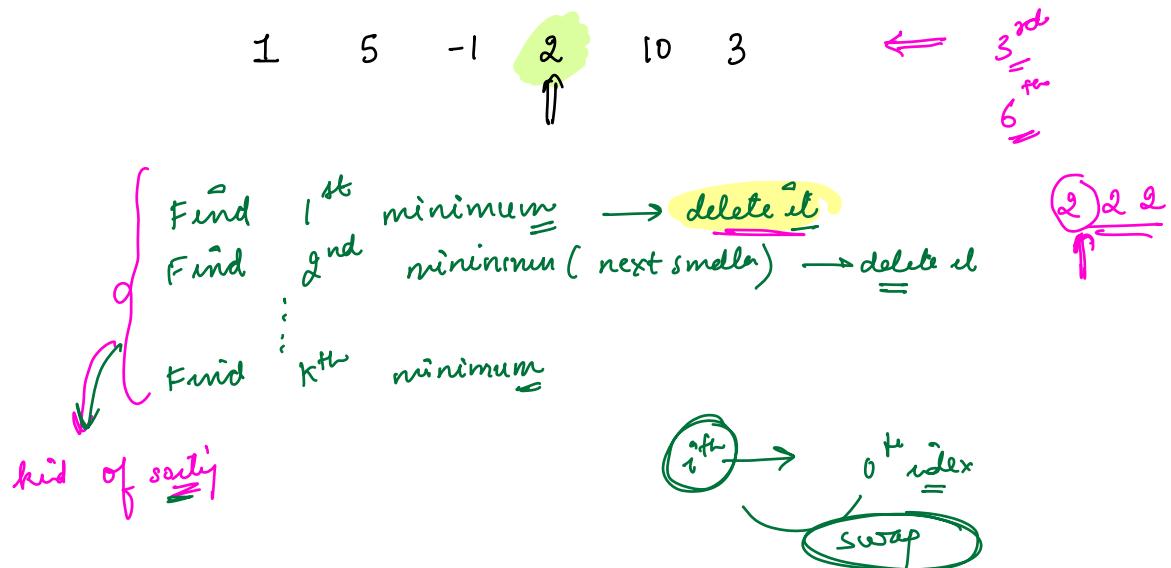
Name	Marks
Mohit	0
Kajal	32
Sheet	45
Aayush	45
Yukti	67
Adilya	92
Aman	100

Stable sort

If 2 data pairs have same parameter value and the data points are in same relative order → stable sort

inplace sorting ⇒ without extra space = $O(1)$

\therefore array elements, Find k^{th} minimum/smallest element



Selection Sort

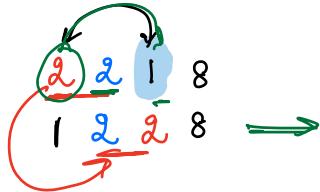
```

for( int i=0; i<n; i++ ) ↵
{
    // pick minimum
    int mn = INT_MAX; int ind=-1;
    for( j=i; j<n; j++ )
    {
        if( arr[j] < mn )
        {
            mn = arr[j];
            ind = j;
        }
    }
    swap( arr[i], arr[ind] );
}

```

T.C = $O(n^2)$

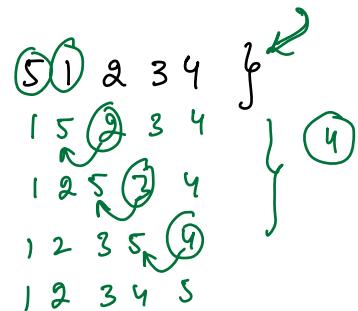
Stable? →
1
unstable



H.W - Stable

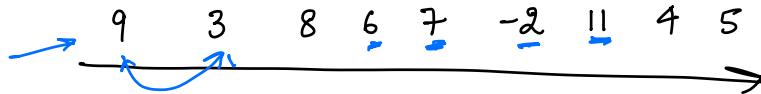
Unplace? — S.C. $O(1)$ — yes

swaps = $n-1$



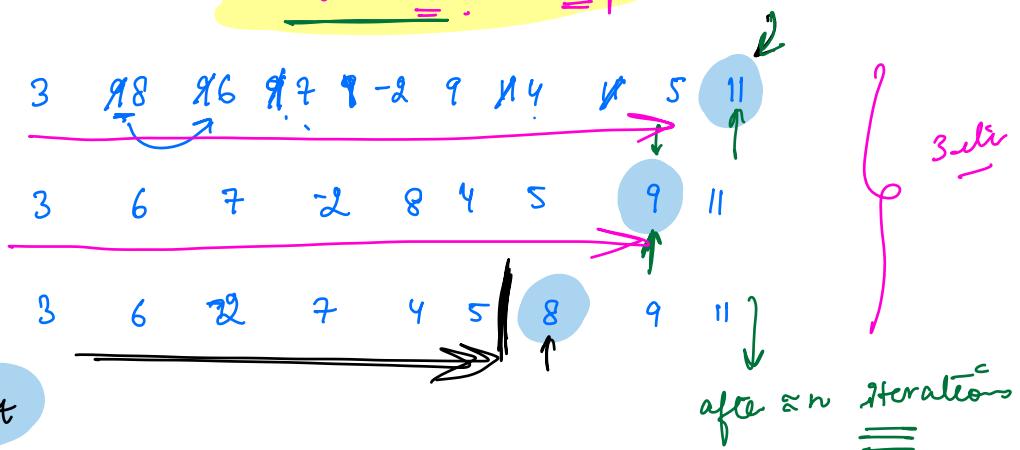
Q

swap adjacent elements



When will you try to swap adjacent!

i , $i+1$
 $a[i] > a[i+1]$ — swap



cont no of swaps
 break thr G.P

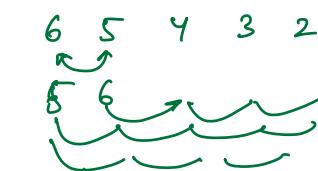
do n iterations —
 $i = 0 \rightarrow n$
 $n-1$
 $n-1-i$
 swap (i^{th} & $i+1^{\text{th}}$ element if required)

T.C: $O(n^2)$

stable? → ✓

inplace? → ✓

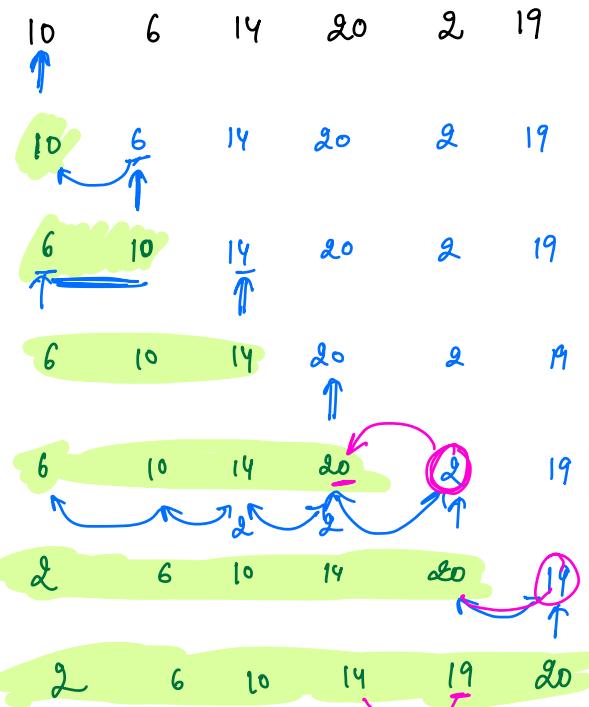
swaps? → $O(n^2)$



ascendi order data
 bubble

Swap
 prearr
 0

Q



Insertion Sort

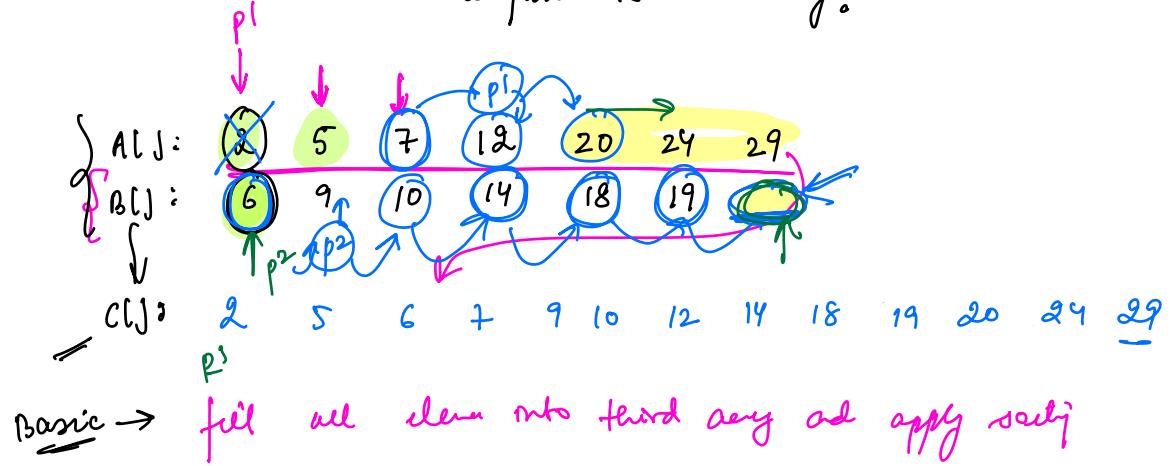
```
for( i=1; i<n; i++)
{
    // [0-i] will be sorted
    int j=i; j>=1
    while ( arr[j]<arr[j-1])
    {
        swap(arr[j], arr[j-1])
        j--;
    }
}
```

$O(i)$
 (\sim)

T.C: $O(n^2)$
S.C: $O(1)$
Inplace
Stable ✓

already sorted $O(n)$

Q Given 2 sorted arrays, merge 2 sorted arrays to get a final sorted array!



2 5 6 7 9 10 12 14 18 19 _____.

```

int merge ( int A[], int B[], int n, int m, int C[])
{
    int p1=0, p2=0, ps=0;
    A   B   C
    while ( p1 <= n && p2 < m )
    {
        if ( A[p1] <= B[p2] )
        {
            C[ps]=A[p1];
            p1++; ps++;
        }
        else {
            C[ps]=B[p2];
            p2++; ps++;
        }
    }
}

```

```

while( p1 < n ) {   c[p3] = A(p1); p1++; p3++; }
while( p2 < m ) {   c[p2] = B(p2); p2++; p3++; }

```

}

T-C $\Theta(n+m)$

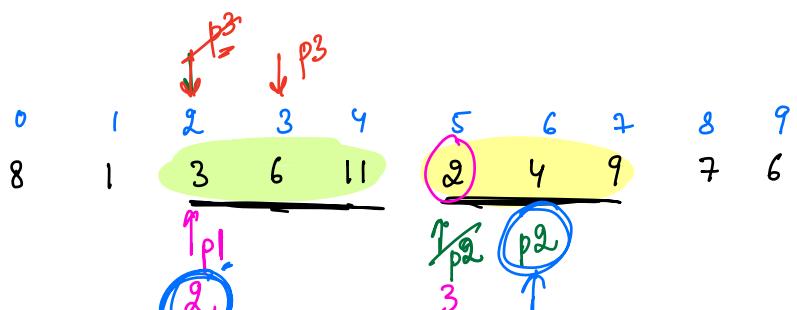
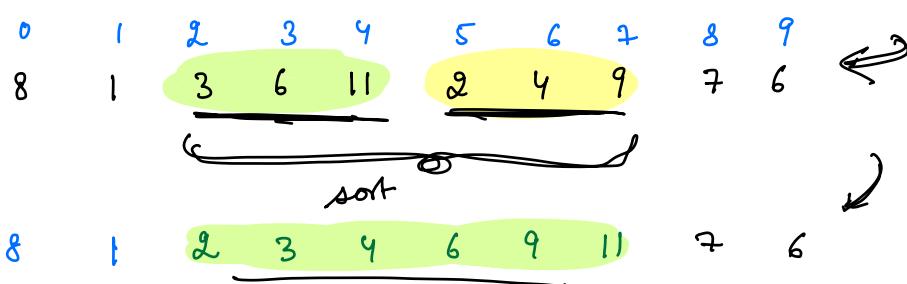
Q

only a single array. N elements (l, m, r)

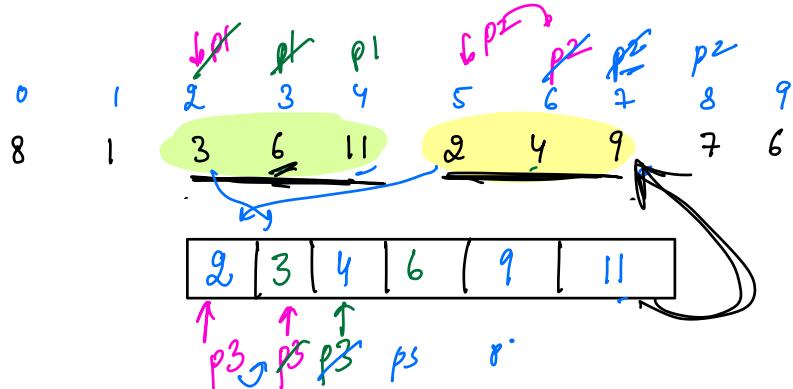
array sorted $\rightarrow l$ to $m-1$
sorted $\rightarrow m$ to r

array \rightarrow sort $\circlearrowleft l$ to r is sorted

$l \quad m \quad r$
 $2, 5, 7$



size of exchange
 $\underline{r-l+1}$
 $\underline{\text{temp}(r-l+1)}$



```
int merge( int arr[], int l, int m, int r)
```

```
{
    int p1 = l;
    int p2 = m;
    int p3 = 0;
    int temp[r-l+1];
    while ( p1 < m && p2 <= r )
    {
        if ( arr[p1] <= arr[p2] )
        {
            temp[p3++] = arr[p1++];
        }
        else
        {
            temp[p3++] = arr[p2++];
        }
    }
    while ( ) { }
    while ( ) { }
}
```

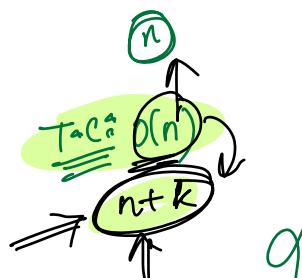
```
for( i=0; i<r-l+1; i++)
{
    arr[l+i] = temp[i];
}
```

}

$0 \rightarrow l+0$
 $1 \rightarrow l+1$
 $2 \rightarrow l+2$
 $3 \rightarrow l+3$

\underline{O} 1 2 1 3 5 3 1 2 5 4 5 3 4 2 $\leftarrow \underline{s_{nk}}$
 $\underline{\text{freq of every element}}$
 $\underline{\text{map your index to values}}$
 $a \rightarrow b$
 $\underline{\text{map}}$

Step :- create your freq array
 for($i=0 \rightarrow i < n; i++$)
 $\quad \text{freq}[a[i]]++;$



$\{$
 for(int $j=1; j \leq b; j++$)
 $\quad \{$
 $\quad \quad \text{while}(\text{ freq}[j] > 0)$
 $\quad \quad \quad \{$
 $\quad \quad \quad \quad \text{cout} \ll j;$
 $\quad \quad \quad \quad \text{freq}[j]--;$
 $\quad \quad \quad \}$
 $\quad \quad \}$
 $\quad \}$
 $\}$

Count sort \rightarrow plug

$10^9 \rightarrow 0$
 $10^9 + 1 \rightarrow 1$
 $10^9 + 100 \rightarrow 100$
 $10^9 + 1000 \rightarrow 1000$
 $\text{cout}[(10^9)]++; ?$

a
 \downarrow
 0
 \downarrow
 1
 \downarrow
 i
 \downarrow
 b
 \downarrow
 $i-a$

$\dots\dots\dots$
 a
 \downarrow
 0
 $a+1$
 \downarrow
 1
 $a+2$
 \downarrow
 2
 $\dots\dots\dots$
 b
 \downarrow
 $a+b$

$0 \rightarrow 10^9$
 $\approx 10^{10}$ value
 \rightarrow
 $\approx 10^6$

