

## Sorting



Arranging data in ascending or descending order based on some rule

Ex 1      3      8      9      14      17

ascending order based on value of number

Ex 2      98      94      75      29      11

descending order

1      13      9      6      12  
1      2      3      4      6

How to use sorting?

Arrays.sort(arr)      y Arrays

Collections.sort(arraylist)      y Array list

TC for sorting :  $n \log n$  ← n is size of array

How/Why? Adv DSA

Q1) Given N array elements, at every step remove an element. Cost to remove = Sum of elements present.  
Find Min Cost to remove all elem.

Ex 1  $\{2, 1, 4\}$

Remove 2	7	$\{1, 4\}$
Remove 1	5	$\{4\}$
Remove 4	<u>4</u>	
	<u>16</u>	

Remove 4	7	$\{1, 2\}$
Remove 2	3	$\{1\}$
Remove 1	<u>1</u>	
	<u>11</u>	

$\{4, 6, 13\}$

11	$\{4, 13\}$
5	$\{13\}$
<u>1</u>	$\{4\}$
<u>17</u>	

$3, 5, 1, -3$

6	$\{3, 1, -3\}$
1	$\{1, -3\}$
-2	$\{-3\}$
<u>-3</u>	
$6 + 1 - 2 - 3$	$= 2$

Obs: We delete numbers in desc order.

$a[0]$     $a[1]$     $a[2]$     $a[3]$

Assume desc order.

Remove $a[0]$	Cost =	$a_0 + a_1 + a_2 + a_3$
Remove $a_1$	Cost =	$a_1 + a_2 + a_3$
Remove $a_2$	Cost =	$a_2 + a_3$
Remove $a_3$	Cost =	$a_3$

---

Total cost =  $a_0 + 2a_1 + 3a_2 + 4a_3$

$(i+1) * a_i$

Code

```
sort(arr) // descending order.  
ans = 0  
for (i=0 ; i < n ; i++) {  
    ans += arr[i] * (i+1)  
}  
return ans.
```

TC:  $n \log n + n \Rightarrow O(n \log n)$

SC:  $O(1)$

## Q2 Noble Integer

Given  $N$  distinct elements, calc no of noble integers in the array.

$$\text{Noble integer} = \{ \text{No of elem} < \text{arr}[i] \} \\ = \text{arr}[i]$$

Ex	1	-5	3	5	-10	4
less	2	1	3	5	0	4

$$\text{ans} = 3$$

	-3	-0	2	5	
less	0	1	2	3	ans = 1
idx					

	-10	-5	1	3	4	5	10
idx	0	1	2	3	4	5	6
less	0	1	2	3	4	5	6

Obs 1: For every elem in sorted array,  
less = idx

Obs 2: To check noble, less = i  
thus  $i = a[i]$

sort(arr)

// asc order

int ans = 0

for (i = 0 ; i < N ; i++) {

if ( i == a[i] )

ans++

}

return ans.

TC:  $O(n \log n)$

SC:  $O(1)$

-10	1	1	2	4	4	4	8	10
0	1	1	3	4	4	4	7	8

Q3 Noble Integer 2

Given  $N$  elements, calc no of noble integers in the array.

Noble integer = {No of elem  $\leq$  arr[i]}  
= arr[i]

$\Sigma n$       0      2      2      4      4      6  
 idn      0      1      2      3      4      5  
 less      0      1      1      3      3      5

$[0, i-1]$   
 $b - a + 1$

-10      1      1      3      1 0 0       $i-1 - 0 + 1$   
          0      1      1      3      4       $= i$   
     $i$

-3    0    2    2    5    5    5    5    8    8    10    10    10    14  
 idn   0    1    2    3    4    5    6    7    8    9    10    11    12    13  
 less   0    1    2    2    4    4    4    4    8    8    10    10    10    13

Obs 1 : If element is repeated :  
less value is same

How to check repeated :  $a[i] == a[i-1]$

Obs 2: If element is not repeated,  
less = idx  
 $a[i] \neq a[i-1]$

Pseudo code:

sort(arr)

// ascending order

int ans = 0

int less = 0

if (  $a[0] == 0$  ) // check  $a[0]$   
ans++

for (  $i=1 ; i < n ; i++$  ) {  
if (  $a[i] == a[i-1]$  ) {

// continue

else

less = i

break  
back at 10:20

if (  $a[i] == less$  )  
ans++;

}

return ans;

TC:  $O(n \log n)$  SC:  $O(1)$

Simpler algo  $\rightarrow$  today  $n^2$   
Advanced algo  $\rightarrow$  Adv DSA  $n \log n$



## Selection sort

Sort students in a line. Sort acc to height

5	6	4	2
2	5	6	4
2	4	5	6

```
void selectsort ( int ar[], int n) {  
    int i, j, minindex  
    for ( i=0 ; i < n-1 ; i++) {  
        minindex = i  
        for (j=i+1 ; j < n ; j++) {  
            if ( ar[j] < ar[minindex] )  
                minindex = j  
        }  
        swap ( ar[i] , ar[minindex] )  
    }  
}
```

TC:  $O(n^2)$

# ● Insert Sort

Deck of cards (Diwali Parties)

3      5      2      4      7      6

2    3    4    5    6    7

3	5	2	4	7	6
2	3	4	5	6	7

3	5	2
	j	i

```
for (i=1; i<n; i++) {  
    cur = ar[i]  
    j = i-1  
    // 0 to i-1 is sorted. Find place of j  
    while (j > 0 && ar[j] > cur) {  
        ar[j+1] = ar[j]  
        j--  
    }  
    ar[j+1] = cur  
}
```

{done}

0 1 2  
2 3 5

$$cur = 2$$

$$i = 2$$

$$j = -1$$

