



Sorting → Arranging data in  $\uparrow$  or  $\downarrow$  order based on a parameter

	Based on						
	Value				factors		
Ex 1	3	8	9	14	19		
Ex 2	19	14	9	8	3		
Ex 3	1	5	3	9	6	10	12
#factors	1	2	2	3	4	4	6

Do you have a predefined sort function in your languages?

→ YES!

{ [TAKES  $O(n \log n)$  time] → how? why? } → we'll see in advance sessions

You can use intubt functions

Que 1. Given N array elements, at every step remove an array element  
 Cost to remove element - sum of array elements present in array  
 Find min cost to remove all the elements.

Ex 1: {2, 1, 4}

$$\begin{aligned}
 &\Rightarrow \text{Remove 2} \quad \{2, 1, 4\} \Rightarrow 7 \\
 &\Rightarrow \text{Remove 1} \quad \{1, 4\} \Rightarrow 5 \\
 &= \text{Remove 4} \quad \{4\} \Rightarrow 4 \\
 &\qquad\qquad\qquad \{ \} \quad \underline{16}
 \end{aligned}$$

$$\begin{aligned}
 &\Rightarrow \text{Remove 4} \quad \{2, 1, 4\} \Rightarrow 7 \\
 &\Rightarrow \text{Remove 2} \quad \{2, 1\} \Rightarrow 3 \\
 &\Rightarrow \text{Remove 1} \quad \{1\} \Rightarrow 1 \\
 &\qquad\qquad\qquad \underline{\underline{11}}
 \end{aligned}$$

(DUIU)

Ex:  $\{4, 6, 1\}$

$$\begin{aligned} \Rightarrow \text{Remove } 6 & \quad \{4, 6, 1\} \Rightarrow 11 \\ \Rightarrow \text{Remove } 4 & \quad \{4, 1\} \Rightarrow 5 \\ \Rightarrow \text{Remove } 1 & \quad \{1\} \Rightarrow 1 \\ & \quad \underline{\underline{17}} \end{aligned}$$

(DUIU)

Ex:  $\{3, 5, 1, -3\}$

$$\begin{aligned} \text{Remove } 5 & \Rightarrow \{3, 5, 1, -3\} \Rightarrow 6 \\ \text{Remove } 3 & \Rightarrow \{3, 1, -3\} \Rightarrow 1 \\ \text{Remove } 1 & \Rightarrow \{1, -3\} \Rightarrow -2 \\ \text{Remove } -3 & \Rightarrow \{-3\} \Rightarrow -3 \\ & \quad \underline{\underline{2}} \end{aligned}$$

Observation  
Elements are removed in decreasing order

(But why?)

Ex  $\{a, b, c, d\}$

$$\text{Remove } a \quad \{a, b, c, d\} \Rightarrow a + b + c + d$$

$$\text{Remove } b \quad \{b, c, d\} \Rightarrow b + c + d$$

$$\text{Remove } c \quad \{c, d\} \Rightarrow c + d$$

$$\text{Remove } d \quad \{d\} \Rightarrow d$$

$$\text{Total cost} = \frac{a+2b+3c+4d}{ }$$

We want to minimize cost  
which ele has highest coeff. ?  $\rightarrow d$

$d$  should be the lowest

1st ele  $\rightarrow$  1st max

2nd ele  $\rightarrow$  2nd max

3rd ele  $\rightarrow$  3rd max

4th ele  $\rightarrow$  4th max

Pseudocode sort in descending order

sum = 0

$(i=0; i < n; i++) \{$   

$$\quad \quad \quad \text{sum} = \text{sum} + [\text{arr}[i] * (i+1)]$$
  

$$\quad \quad \quad \}$$
  

$$\quad \quad \quad \text{return sum}$$

TC  $\rightarrow O(n \log n) + O(n)$

SC  $\rightarrow O(1)$

Ques 2. Given  $n$  ele, calc no. of noble integers present.  
 $\text{arr}[i]$  is said to be noble integer if  $\{ \text{No. of ele} < \text{arr}[i] = \text{arr}[i] \}$

Ex:  $\{ 1, -5, 3, 5, -10, 4 \}$   
# count less  $\rightarrow 2, 1, 3, 5, 0, 4$   
ans = 3

(Quiz 3)

Ex:  $\{ -3, 0, 2, 5 \}$   
# count less  $\rightarrow 0, 1, 2, 3$   
ans = 1

(Quiz 4)

Ex:  $\{ -10, -5, 1, 3, 4, 5, 10 \}$   
# count less  $\rightarrow 0, 1, 2, 3, 4, 5, 6$   
ans = 3

Simple Approach

For every ele, get no. of elements less than  $\text{arr}[i]$

TC  $\rightarrow O(n^2)$

count = 0  
for ( $i = 0$ ;  $i < n$ ;  $i++$ ) {

    less = 0

        for ( $j = 0$ ;  $j < n$ ;  $j++$ ) {

            if ( $\text{arr}[j] < \text{arr}[i]$ ) {

                less++

            }

        }

    if ( $\text{less} == \text{arr}[i]$ ) count++

}

return count

→ part which is taking time

What shall we do?  
→ Sort the array?

Ex 3:  $\{ -10, -5, 1, 3, 4, 5, 10 \}$

TC  $O(n \log n)$

Elements can repeat

	0	1	2	3	4	5
Ex1 :	{	(0)	2	2	4	4
#count	0	1	1	3	3	5
				6	ans = 1	

Quiz 5

Ex 2:  $\{ -10, 1, 1, 3, 100 \}$

Soln: Sort & count all less ele  $\rightarrow O(n^2)$   $\Rightarrow$  Should work

Ques: Will second sol work? - NO

	0	1	2	3	4	5
Ex:	0	2	2	4	4	6
#count less	0	1	1	3	3	5

	0	1	2	3	4	5	6	7	8
Ex:	-10	1	1	1	4	4	4	7	10
#count less	0	1	1	1	4	4	4	7	8
	↓	↓	↓	↓	↓	↓	↓	↓	↓

(Quiz 6)	-10	1	1	2	4	4	4	4	8	10
Ex:	0	1	1	3	4	4	4	7	8	
cnt < arr[i]:	0	1	1	3	4	4	4	7	8	

$arr = 5$

(Quiz 7)	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Ex:	{ -3	0	2	2	5	5	5	5	8	8	10	10	10	14 }
cnt < arr[i]	0	1	2	2	4	4	4	4	8	8	10	10	10	13

$arr = 7$

(Quiz 7)	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Ex:	{ -3	0	2	2	5	5	5	5	8	8	10	10	10	14 }
cnt < arr[i]	0	1	2	2	4	4	4	4	8	8	10	10	10	13

$arr = 7$

#1 When element comes for first time  
no. of ele less than arr[i] = i

#2 If it comes not for first time  
then cnt is same as previous ele cnt.

How to know if occurring for first time?

↳ we can just check with the previous ele

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

// Sort array

```
int ans = 0
int less = 0
for(i=1; i<N; i++) {
    if(arr[i] != arr[i-1])
        less = i
    else { // ele is repeating
        // less won't change
        if(less == arr[i]) {
            ans ++
        }
    }
}
return ans
```

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

## Comparator

Q. Given N array ele, sort them in increasing order of their no. of factors.  
 If two ele have same factors, ele with less value will come first.

# No extra space.

Ex: {  
 # factors      { 9, 3, 4, 8, 16, 37, 6, 13, 15 } }  
 3      13      37      4      9      6      8      15      16

Ex {  
 # factors      0      1      2      3      4      5      6  
 1      21      6      23      10      14      25 }

1      4      4      2      4      4      3

{ 1      23      25      6      10      14      21 }

# Do we know any sorting technique as of now? - NO!  
 # We just know that there exists an inbuilt function sort()

# Modify one inbuilt sort function using comparator.

Ex 1      25      16      }      In final order  
              ↓      ↓      }      25 should come first  
              3      5

Ex 2       $\begin{array}{r} 10 \\ \downarrow \\ 4 \end{array}$        $\begin{array}{r} 9 \\ \downarrow \\ 3 \end{array}$       }      9 should come before 10

$$\text{Ex 3} \quad \begin{array}{r} 49 \\ \times 29 \\ \hline 432 \\ + 98 \\ \hline 1421 \end{array} \quad \left. \begin{array}{l} \downarrow \\ 3 \end{array} \right\} \begin{array}{l} \text{since factors are same} \\ 29 < 49 \\ \text{so 29 will come first} \end{array}$$

```

int f1 = factors(a)
int f2 = factors(b)
if (f1 < f2) return true
if (f1 == f2 and a < b) return true
return false
}

```

`sort(ar[], comp)`

Q Given N array ele, sort in increasing order of no. of digits  
 If two ele have same digits, ele with more value should come first.

93	2	37	639	8	100	345	49
↓	↓	↓	↓	↓	↓	↓	↓
2	1	2	3	1	3	3	2

Ans → 8    2    93    49    37    639    345    100

```
bool comp(int a, int b) {
    int d1 = digits(a)
    int d2 = digits(b)
    if(d1 < d2) return true
    if(d1 == d2 and a > b) return true
    return false
}
```