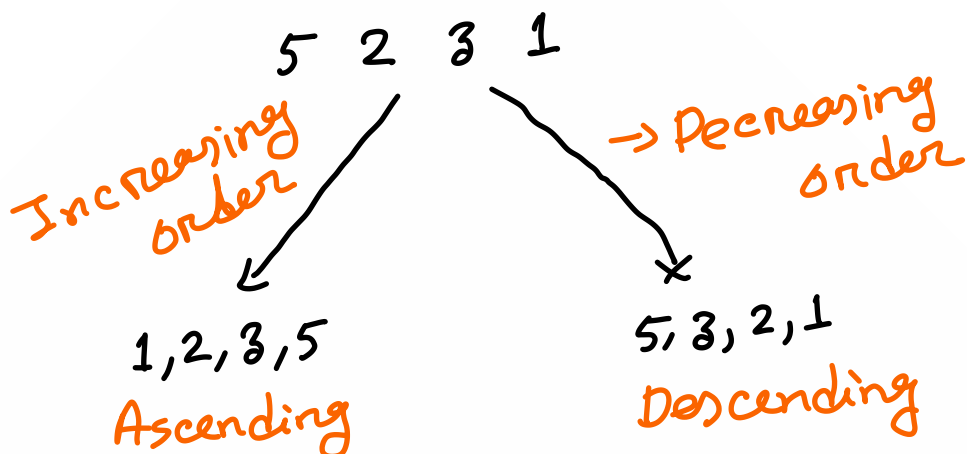


# Sorting

↳ Particular order



Time complexity  $O(n^2)$

↳ Bubble Sort  $(n-1)$

↳  $(n-1)$  iterations

adj element compare

larger element push in last  
using → swap

↳ [4, 1, 5, 2, 3]

1st iteration:

4 1 5 2 3  
1 4 5 2 3  
1 4 5 2 3  
1 4 2 5 3  
1 4 2 3 5

↳ In 1st iteration we push the max element in the last  $i=0$

2nd iteration:

1 4 2 3 5  
1 4 2 3 5  
1 2 4 3 5  
1 2 3 4 5

↳ In 2nd iteration we push the 2nd largest element  $i=1$

3rd iteration:

1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5

↳ In 3rd iteration we push the 3rd largest element  $i=2$

4th iteration:

1 2 3 4 5  
1 2 3 4 5

↳ 4 elements are in right place so we don't need any iteration  $i=3$

↳ This is the process we follow on bubble sort

Pseudocode:

↳ We will run the outer loop  $(n-1)$  time

so, For  $i=0, j=4$   
 $i=1, j=3$   
 $i=2, j=2$   
 $i=3, j=1$

our inner loop will be run

```
for (int i=0, i<n-1, i++) { → outerloop  
    for (int j=0, j<n-i-1, j++) { → innerloop  
        if (A[j] > A[j+1])  
            swap(A[j], A[j+1])  
    }  
}
```



Shohanur Rahman Shohan  
CSE  
B210101004