

# Insertion sort

---

**Problem** - Given an array of integers, sort the array.

```
const arr = [-6, 20, 8, -2, 4]
```

```
insertionSort(arr) => Should return [-6, -2, 4, 8, 20]
```

# Insertion sort idea

---

Virtually split the array into a sorted and an unsorted part

Assume that the first element is already sorted and remaining elements are unsorted

Select an unsorted element and compare with all elements in the sorted part

If the elements in the sorted part is smaller than the selected element, proceed to the next element in the unsorted part. Else, shift larger elements in the sorted part towards the right.

Insert the selected element at the right index

Repeat till all the unsorted elements are placed in the right order

# Insertion sort example

[ -6 20 8 -2 4 ]	NTI = 20	SE = -6	-6 > 20? No. Place 20 to the right of -6
[ -6 20 8 -2 4 ]	NTI = 8	SE = 20	20 > 8? Yes. Shift 20 to the right
[ -6 20 20 -2 4 ]	NTI = 8	SE = -6	-6 > 8? No. Place 8 to the right of -6
[ -6 8 20 -2 4 ]	NTI = -2	SE = 20	20 > -2? Yes. Shift 20 to the right
[ -6 8 20 20 4 ]	NTI = -2	SE = 8	8 > -2? Yes. Shift 8 to the right
[ -6 8 8 20 4 ]	NTI = -2	SE = -6	-6 > 2? No. Place -2 to the right of -6
[ -6 -2 8 20 4 ]	NTI = 4	SE = 20	20 > 4? Yes. Shift 20 to the right
[ -6 -2 8 20 20 ]	NTI = 4	SE = 8	8 > 4? Yes. Shift 8 to the right
[ -6 -2 8 8 20 ]	NTI = 4	SE = -2	-2 > 4? No. Place -4 to the right of -2
[ -6 -2 4 8 20 ]			Reached end of array. Array is sorted.