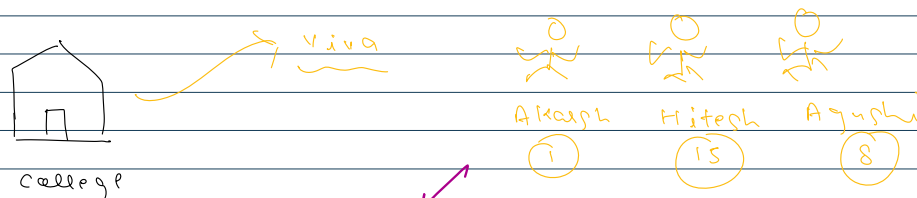
Sorting



order wise

Roll no → 1, 8, 15 → Sorting

Sorting → ascending (smaller to largest)
Sorting → descending (largest to smallest)

Sorting is a process of arranging data in a specific order, usually ascending to descending.

1. Numerical order :) 1, 2, 3, 5, 7, 8
2. Alphabetical order :) A, B, C, D
3. Chronological order :) date/time

a) Sort numbers
b) Sort words :) ["Akash", "Vanshika", "Rajesh"]

["Akash", "Rajesh", "Vanshika"]

Stories to remember → 1

marks → [72, 92, 82, 0, 55, 22] :) [0, 22, 55, 72, 82, 92]

Rank 6 → highest marks
Rank 1 → lowest marks

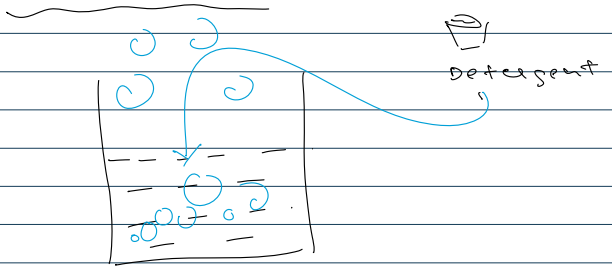
back-benchers

0	22	55	72	82	92

Sorting algorithms → Bubble, selection

Sorting algorithms
- Bubble
- Selection
- Insertion

Bubble Sort



"Bubble the max element at last"

(1)	(5)	2	6	4
1	(5)	(2)	6	4
1	2	(5)	(6)	4
1	2	5	(6)	(4)
1	2	5	4	(6)

arr[i] > arr[i+1] ⇒ swap

1 > 5 → X
5 > 2 → ✓
5 > 6 → X
6 > 4 → ✓

→ 1st

(1)	(2)	5	4	6
1	(2)	(5)	4	6
1	2	(5)	(4)	6
1	2	4	5	6

1 > 2 → X
2 > 5 → X
5 > 4 → ✓

→ 2nd

(1)	(2)	4	5	6
1	(2)	(4)	5	6
1	2	4	5	6

1 > 2 → X
2 > 4 → X

→ 3rd

(1)	(2)	4	5	6
1	2	4	5	6

1 > 2 → X

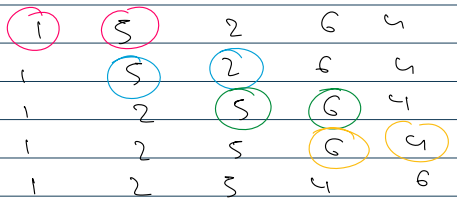
→ 4th

(1)	(5)	2	6	4
1	(5)	(2)	6	4
1	2	(5)	(6)	4
1	2	5	(6)	(4)
1	2	5	4	(6)

arr[i] > arr[i+1] ⇒ swap

1 > 5 → X
5 > 2 → ✓
5 > 6 → X
6 > 4 → ✓

How many times the above process get repeated ?



arr[i] > arr[i+1] ? swap

1 > 5 → X

5 > 2 → ✓

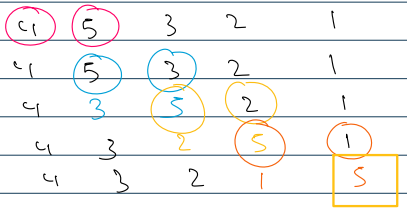
5 > 6 → X

6 > 4 → ✓

```
public class Main {
    public static void main(String[] args) {

        int[] arr = {4, 5, 3, 2, 1};
        int n = arr.length;
        bubbleSort(arr);

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+ " ");
        }

    }

    public static void bubbleSort(int[] arr){
        int n = arr.length;
        for(int turn=1; turn<n; turn++){
            for(int i=0; i<n-turn; i++){
                if(arr[i]>arr[i+1]){
                    int temp = arr[i];
                    arr[i] = arr[i+1];
                    arr[i+1] = temp;
                }
            }
        }
    }
}
```

4	5	3	2	1

turn : 1



4 times

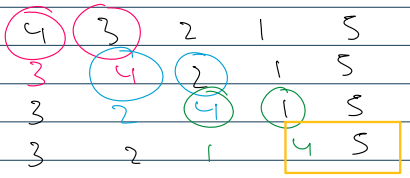
4 > 5 → X

5 > 3 → ✓

5 > 2 → ✓

5 > 1 → ✓

turn : 2



3 times

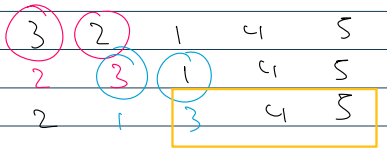
4 > 3 → ✓

4 > 2 → ✓

4 > 1 → ✓

```
public static void bubbleSort(int[] arr){
    int n = arr.length;
    for(int turn=1; turn < n; turn++){
        for(int i=0; i<; i++){
            if(arr[i]>arr[i+1]){
                int temp = arr[i];
                arr[i] = arr[i+1];
                arr[i+1] = temp;
            }
        }
    }
}
```

turn : 3

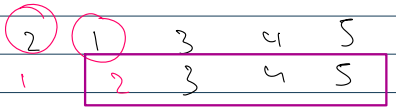


2 times

3 > 2 → ✓

3 > 1 → ✓

turn : 4



1 times

2 > 1 → ✓

turn | internal loop | arr.length

turn	internal loop
1	4
2	3
3	2
4	1

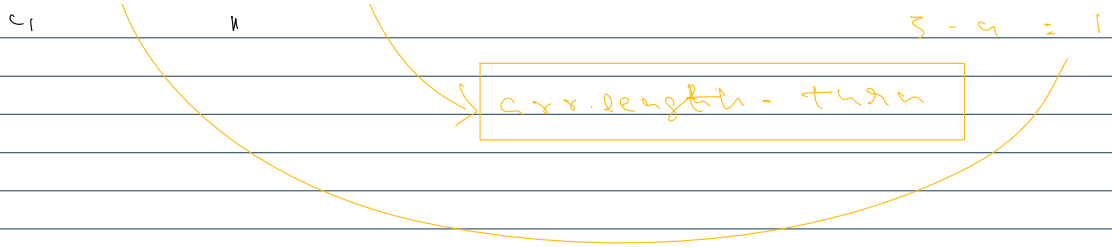
5

5 - 1 = 4

5 - 2 = 3

5 - 3 = 2

5 - 4 = 1



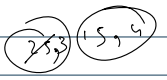
Stories to remember → 0

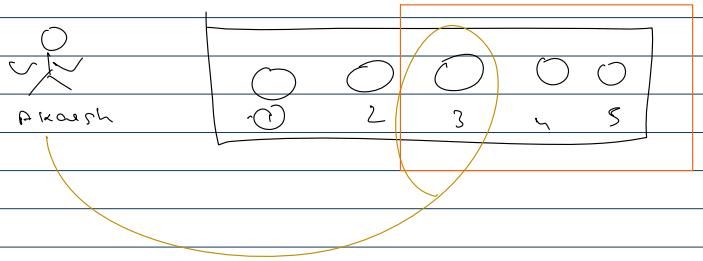
marks → [72, 82, 92, 0, 22, 55]

0	22	55	72	82	92

Selection Sort

"put minimum element at first position"





3 → 25

4 → 15

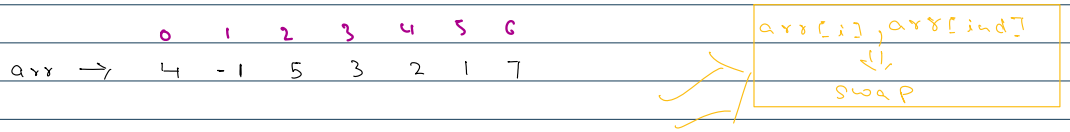
5 → 50

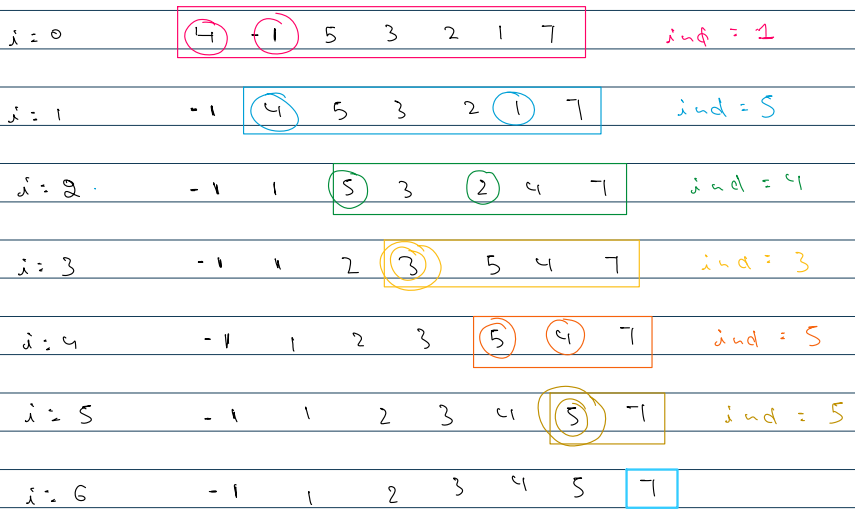
Can you find the index of minimum value, in an array of length n starting from the given index ?

```
public class Main {
    public static void main(String[] args) {

        int[] arr = {4, 5, 3, 2, 1, -1, -5, 0, -6, 111};
        int ind = 3;
        int n = arr.length;

        System.out.println(minimumElementIndex2(arr, ind));
    }

    public static int minimumElementIndex1(int[] arr, int ind){
        int minVal = arr[ind];
        int minInd = ind;

        int n = arr.length;

        for(int i=ind; i<n; i++){
            if(arr[i] < minVal){
                minVal = arr[i];
                minInd = i;
            }
        }
        return minInd;
```

```
    }

    public static int minimumElementIndex2(int[] arr, int ind){
        int minInd = ind;

        int n = arr.length;

        for(int i=ind; i<n; i++){
            if(arr[i] < arr[minInd]){
                minInd = i;
            }
        }
        return minInd;
    }

}
```

ind → index of minimum element in the array.



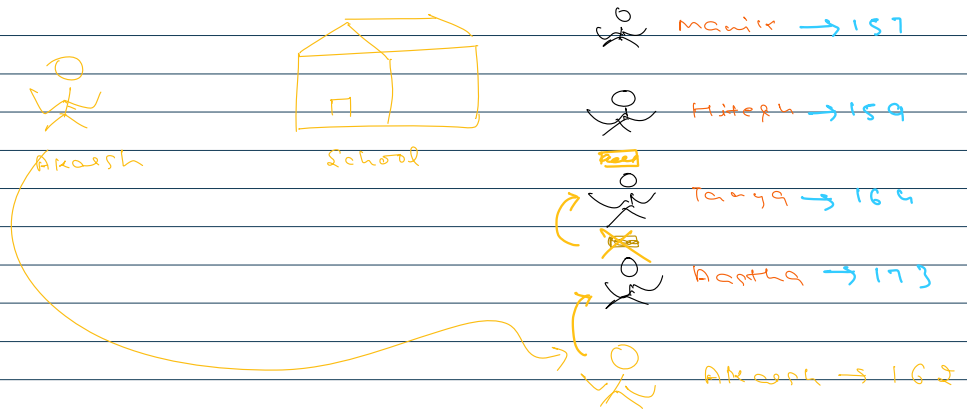


https://pythontutor.com/render.html#mode=display

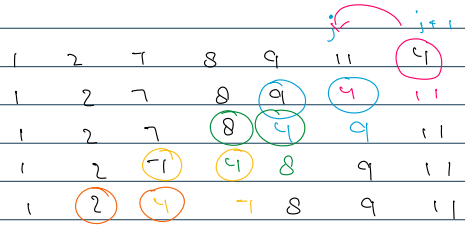
```
public class Main {
    public static void main(String[] args) {

        int[] arr = {4, 5, 3, 2, 1, -1, -5, 0, -6, 111};
        int ind = 3;
        int n = arr.length;

        selectionSort(arr);

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+" ");
        }

    }

    public static void selectionSort(int[] arr){
        int n = arr.length;
        for(int i=0; i<n; i++){
            int idx = minimumElementIndex(arr, i);
            // arr[i], arr[idx] => Swap
```

```
            int temp = arr[i];
            arr[i] = arr[idx];
            arr[idx] = temp;
        }
    }

    public static int minimumElementIndex(int[] arr, int ind){
        int minInd = ind;
        int n = arr.length;

        for(int i=ind; i<n; i++){
            if(arr[i] < arr[minInd]){
                minInd = i;
            }
        }
        return minInd;
    }

}
```

Stories to remember → (3)



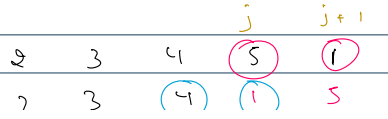

```
arr[j] > arr[j+1]
    ⇅
  swap
```



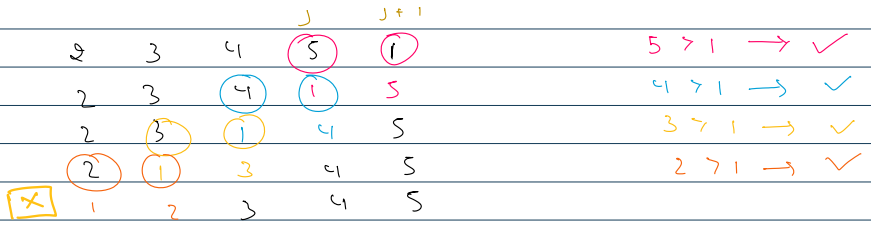
11 > 4 → swap
9 > 4 → swap
8 > 4 → swap
7 > 4 → swap
2 > 4 → ✗

```
while(arr[j] > arr[j+1]){
    // swap both arr[j] & arr[j+1]
    j--;
}
```

j >= 0 ✗



5 > 1 → ✓
4 > 1 → ✓

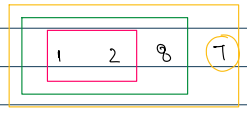


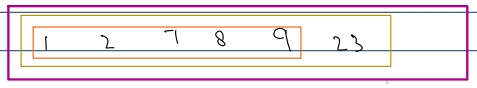
Can you write a fn to move the last element to its correct sorted position ?
Assume the rest of the array is already sorted !

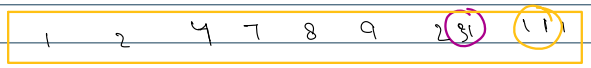
```
public class Main {
    public static void main(String[] args) {

        int[] arr = {1, 2, 7, 8, 9, 11, 4};
        int ind = 3;
        int n = arr.length;

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+" ");
        }

        System.out.println();
        insertLastElement(arr, n-1);

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+" ");
        }
    }

    public static void insertLastElement(int[] arr, int i){
        int j = i-1;

        while(j>=0 && arr[j]>arr[j+1]){
            // swap => arr[j], arr[j+1]
            int temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;

            j--;
        }
    }

}
```

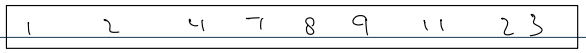
Insertion Sort

Repeat the above process for n times i.e 1 to n-1









```
public class Main {
    public static void main(String[] args) {

        int[] arr = {1, 2, 7, 8, 9, 11, 4};
        int ind = 3;
        int n = arr.length;

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+" ");
        }

        System.out.println();
        // insertLastElement(arr, n-1);
        insertionSort(arr);

        for(int i=0; i<n; i++){
            System.out.print(arr[i]+" ");
        }
    }

    public static void insertionSort(int[] arr){
        int n = arr.length;
        for(int i=1; i<n; i++){
            insertLastElement(arr, i);
        }
    }

    public static void insertLastElement(int[] arr, int i){
        int j = i-1;

        while(j>=0 && arr[j]>arr[j+1]){
            // swap => arr[j], arr[j+1]
            int temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;

            j--;
        }
    }

}
```