

# *SORTING*

Arrange in an order

# Basic Algorithms

Bubble Sort

Selection Sort

Insertion Sort

Counting Sort

**Advanced**

## Bubble Sort

{ 5, 4, 1, 3, 2 }    unsorted array  
{ 1, 2, 3, 4, 5 }    increasing order  
{ 5, 4, 3, 2, 1 }    decreasing order

# Bubble Sort

## Inspiration



## Idea

Large elements come to the end of array by swapping with adjacent elements

# Bubble Sort

$n = 5$

5, 4, 1, 3, 2

# Selection Sort

## **Idea**

pick the smallest (from  
unsorted), put it at the  
beginning

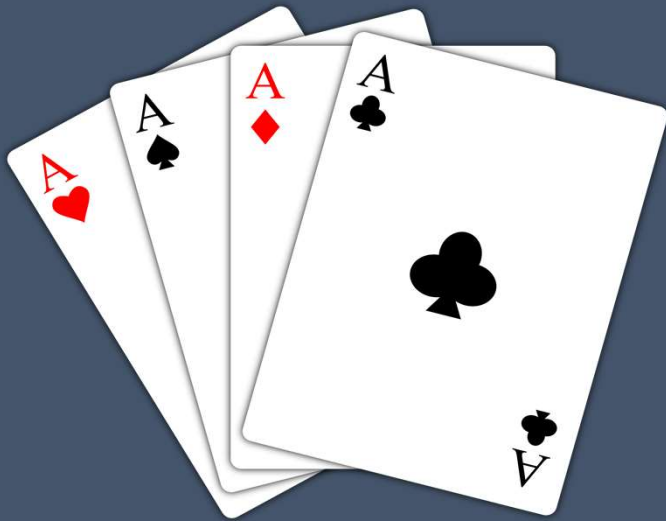
## Selection Sort

**n = 5**

5, 4, 1, 3, 2

# Insertion Sort

## Inspiration



## Idea

Pick an element (from  
unsorted part) &  
place in the right pos  
in sorted part



# Insertion Sort

$n = 5$

5, 4, 1, 3, 2

## Inbuilt Sort

```
import java.util.Arrays;  
Arrays.sort(arr)  
 $O(n \log n)$ 
```

```
Arrays.sort(arr, si, ei)
```

## Inbuilt Sort

```
import java.util.Collections;  
Arrays.sort( arr, Collections.reverseOrder() )
```

```
Arrays.sort( arr, si, ei, Collections.reverseOrder() )
```

## Counting Sort

1, 4, 1, 3, 2, 4, 3, 7