Algorithms and Complexity

Sorting Algorithms

Lecturer: Dr. Alaa Ahmed Abbood
Lecture 5.
Class 2nd .
Time: 8:30-10:30
Department: Businesses Information Technology (BIT)

Outline

Internal Sorting Algorithms

➢1- Bubble Sort

2- Insertion Sort

3- Selection Sort

4- Homework

➢1-Bubble Sort

- ❖ Bubble sort is a comparison based sorting algorithm wherein comparing adjacent elements is a primitive operation. In each pass, it compares the adjacent elements in the array and exchanges those that are not in order. Basically, each pass through the array places the next largest value in its proper place, hence the number of comparisons reduces by one at each pass

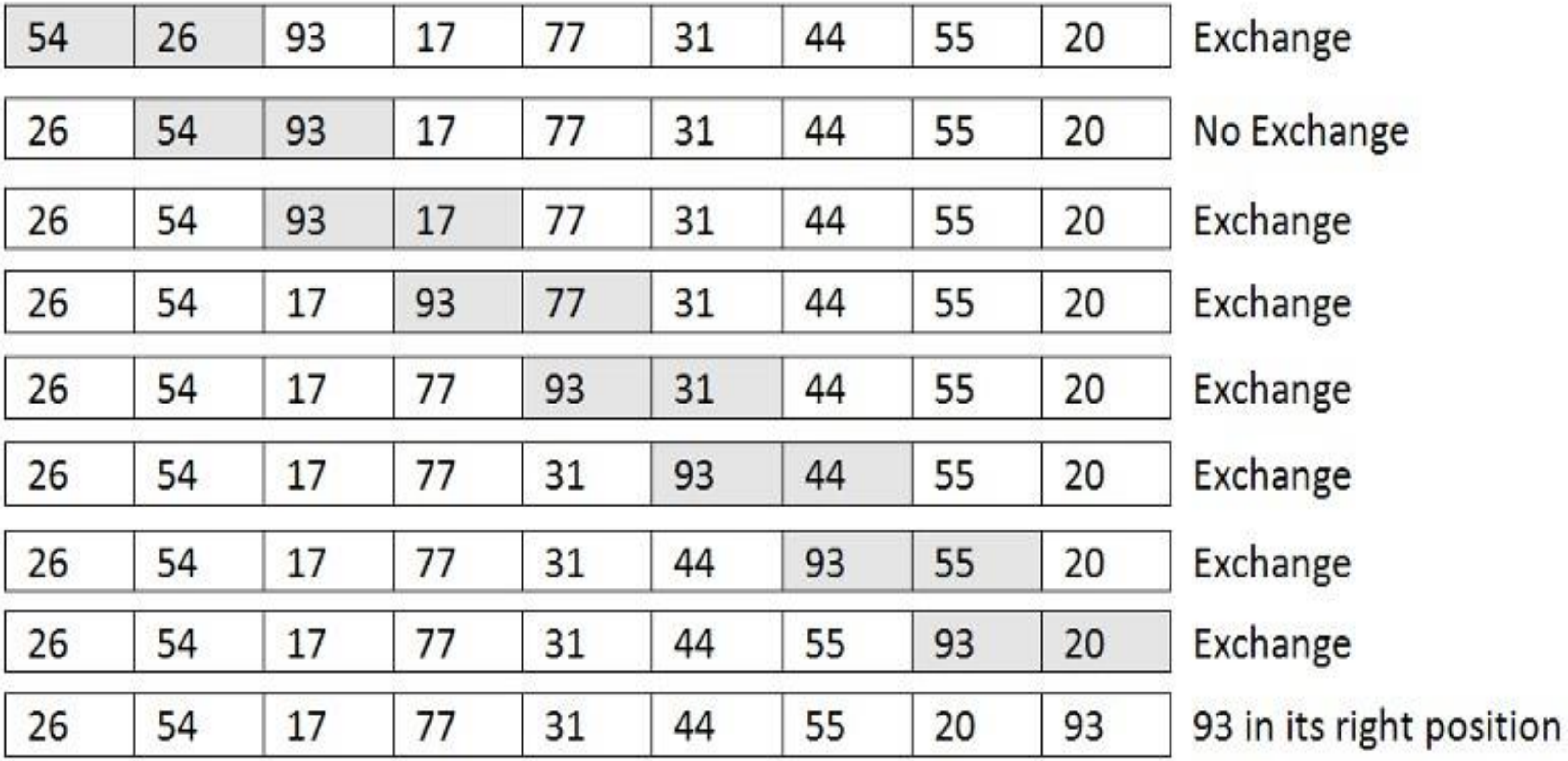
Example: Bubble Sort

6 5 3 1 8 7 2 4

Trace of Bubble Sort Algorithm

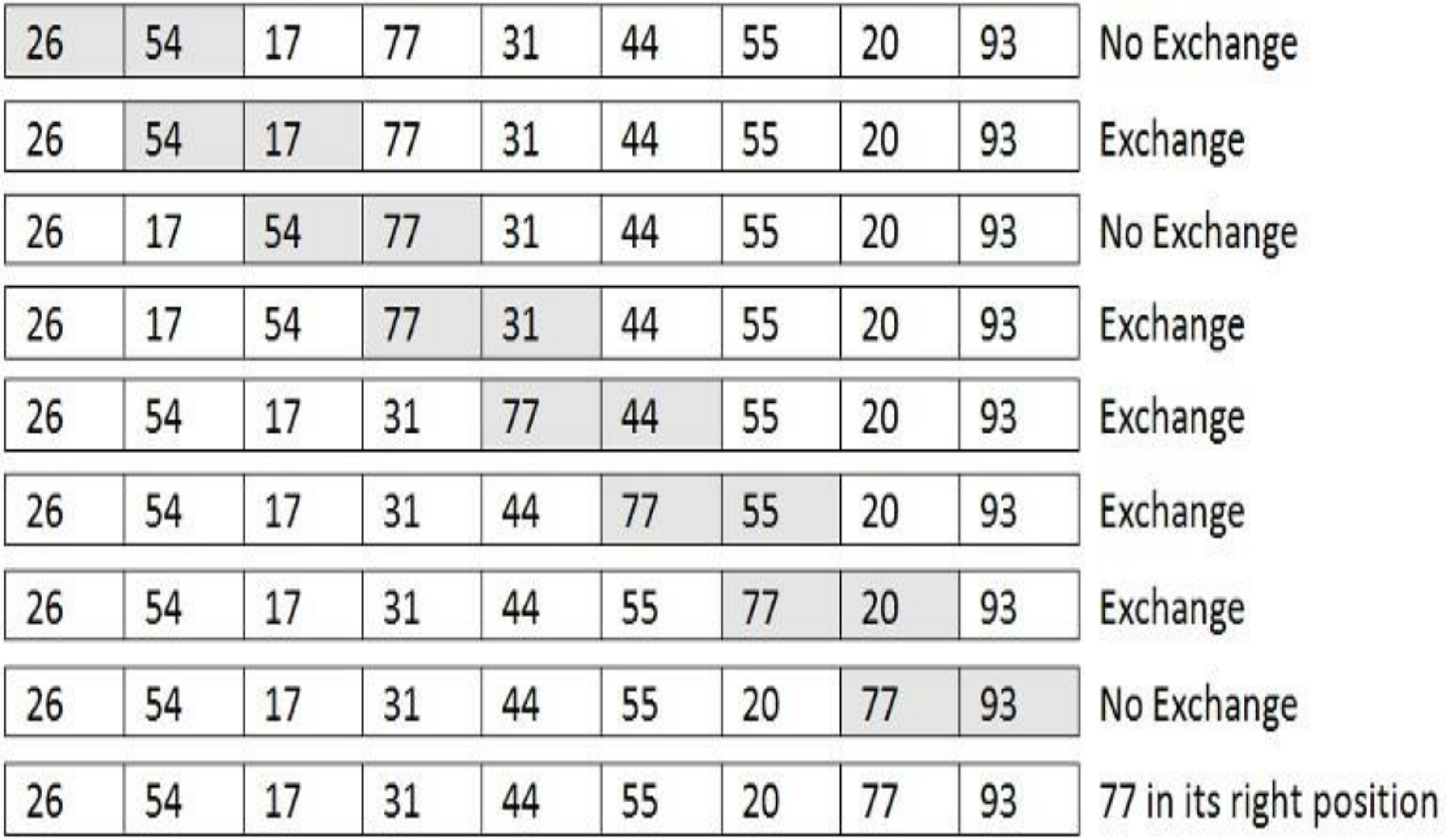
Pass 1

Input: a[9] = {54, 26, 93, 17, 77, 31, 44, 55,20}



54	26	93	17	77	31	44	55	20	Exchange
26	54	93	17	77	31	44	55	20	No Exchange
26	54	93	17	77	31	44	55	20	Exchange
26	54	17	93	77	31	44	55	20	Exchange
26	54	17	77	93	31	44	55	20	Exchange
26	54	17	77	31	93	44	55	20	Exchange
26	54	17	77	31	44	93	55	20	Exchange
26	54	17	77	31	44	55	93	20	Exchange
26	54	17	77	31	44	55	20	93	93 in its right position

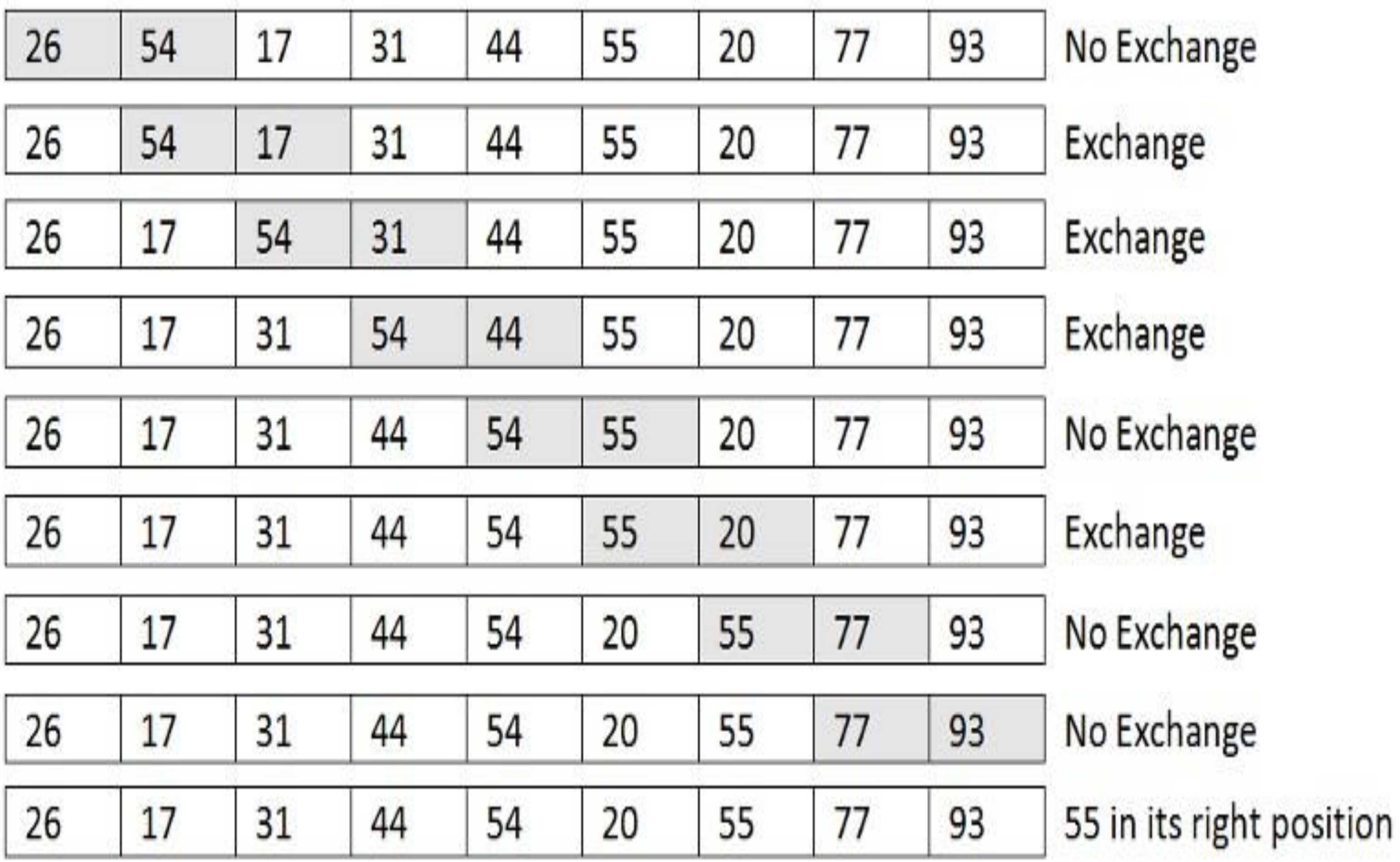
Cont



26	54	17	77	31	44	55	20	93	No Exchange
26	54	17	77	31	44	55	20	93	Exchange
26	17	54	77	31	44	55	20	93	No Exchange
26	17	54	77	31	44	55	20	93	Exchange
26	54	17	31	77	44	55	20	93	Exchange
26	54	17	31	44	77	55	20	93	Exchange
26	54	17	31	44	55	77	20	93	Exchange
26	54	17	31	44	55	20	77	93	No Exchange
26	54	17	31	44	55	20	77	93	77 in its right position

Cont..

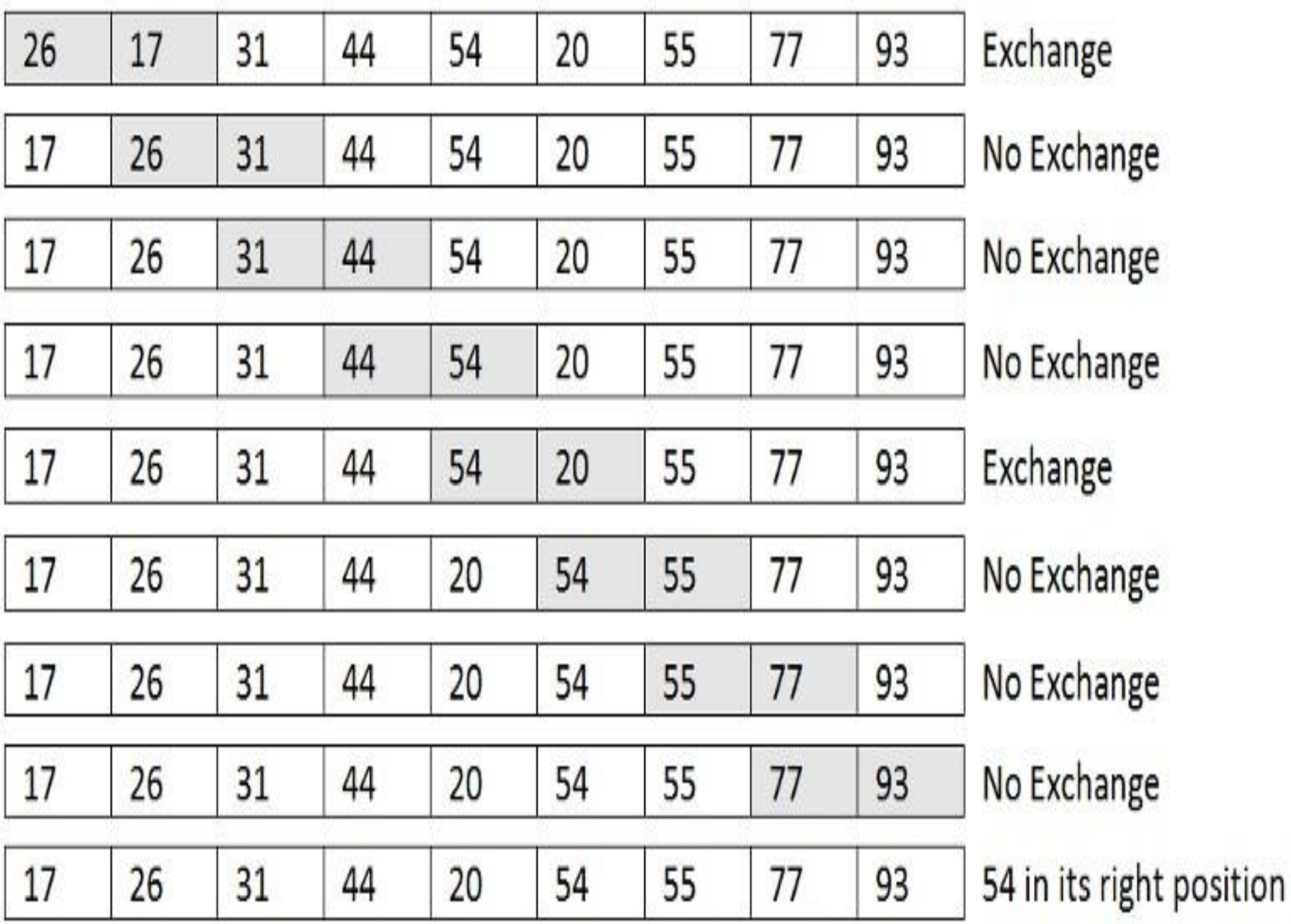
Pass 3



26	54	17	31	44	55	20	77	93	No Exchange
26	54	17	31	44	55	20	77	93	Exchange
26	17	54	31	44	55	20	77	93	Exchange
26	17	31	54	44	55	20	77	93	Exchange
26	17	31	44	54	55	20	77	93	No Exchange
26	17	31	44	54	55	20	77	93	Exchange
26	17	31	44	54	20	55	77	93	No Exchange
26	17	31	44	54	20	55	77	93	No Exchange
26	17	31	44	54	20	55	77	93	55 in its right position

Cont..

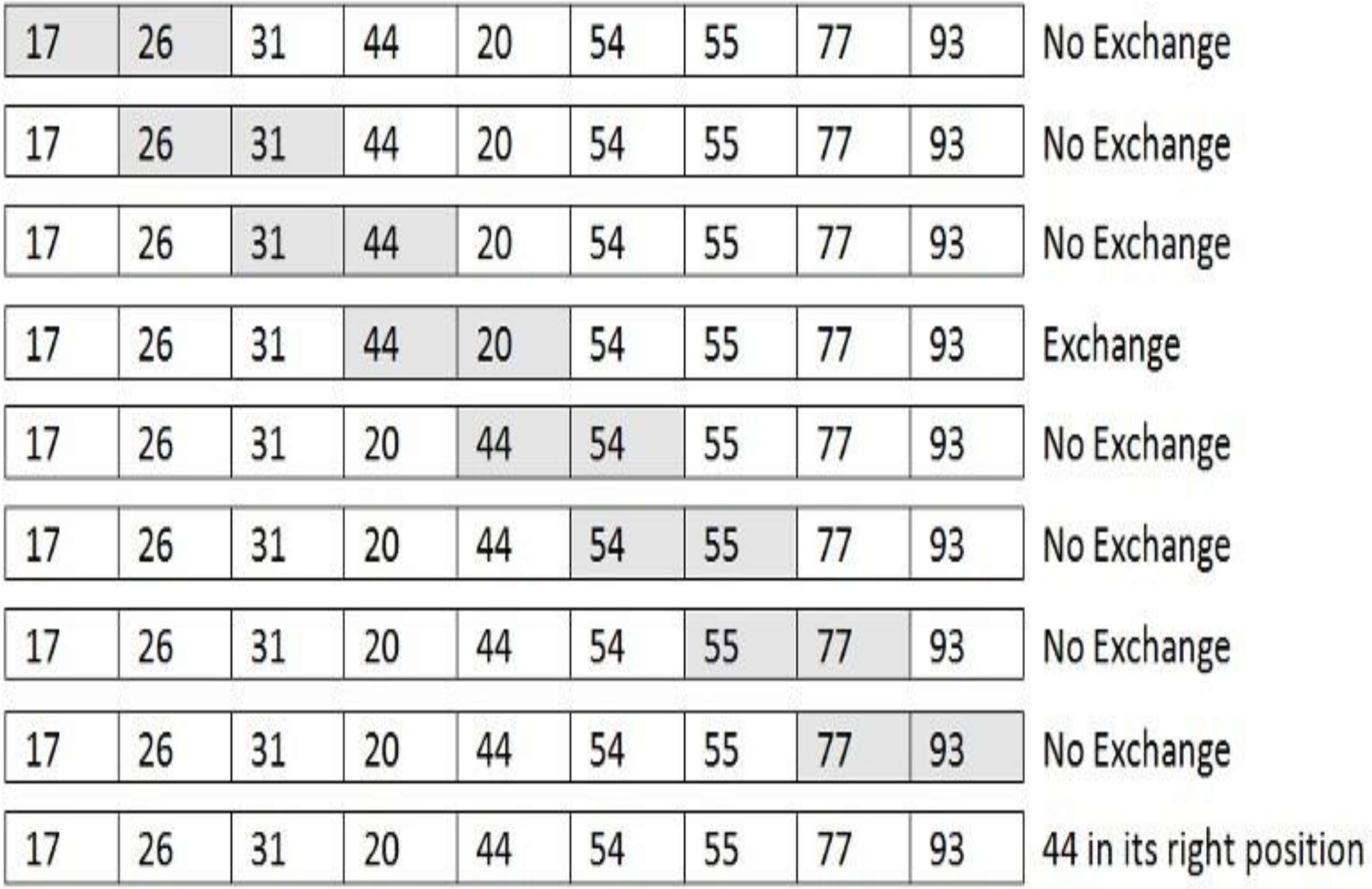
Pass 4



26	17	31	44	54	20	55	77	93	Exchange
17	26	31	44	54	20	55	77	93	No Exchange
17	26	31	44	54	20	55	77	93	No Exchange
17	26	31	44	54	20	55	77	93	No Exchange
17	26	31	44	54	20	55	77	93	Exchange
17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	54 in its right position

Cont..

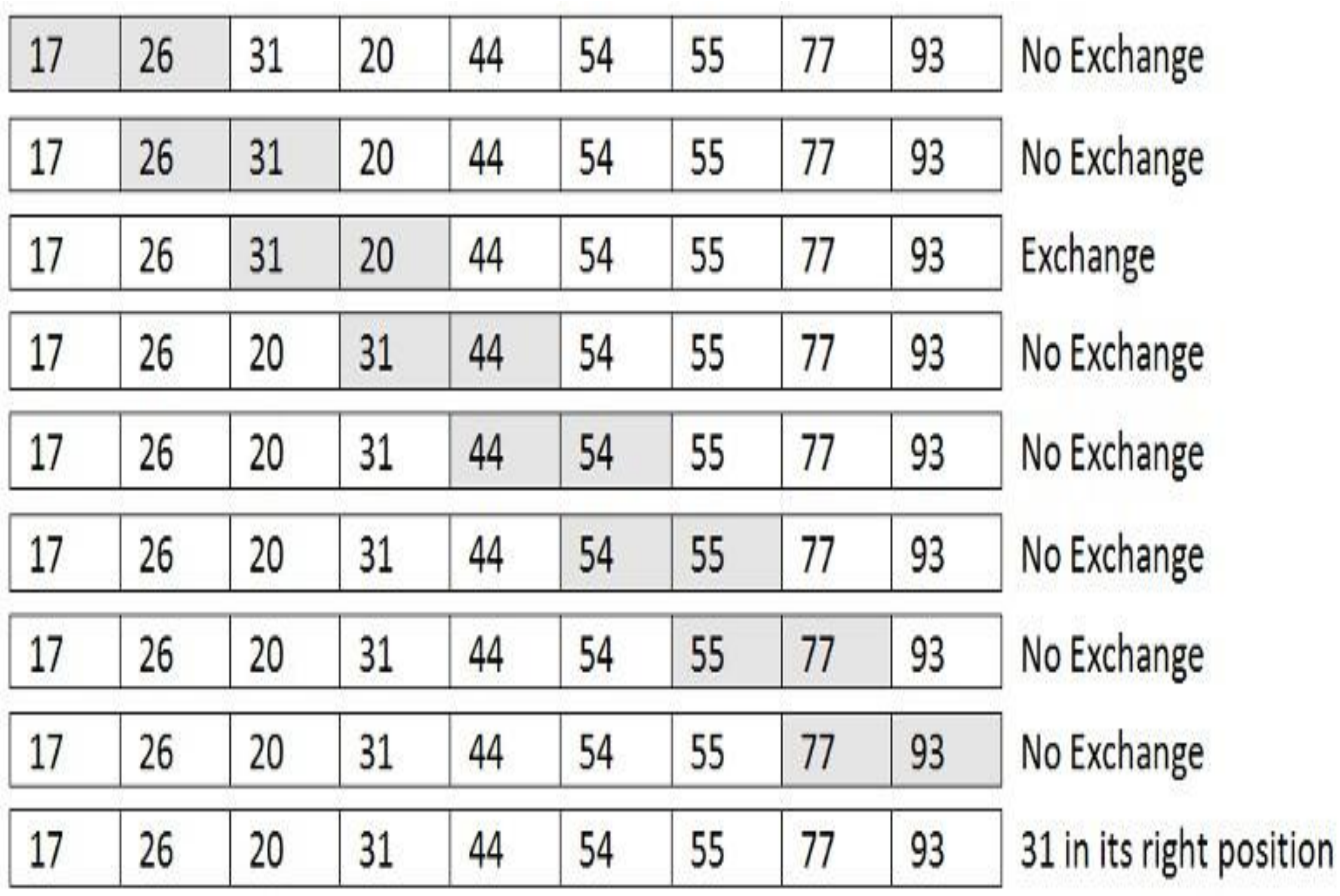
Pass 5



17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	No Exchange
17	26	31	44	20	54	55	77	93	Exchange
17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	44 in its right position

Cont..

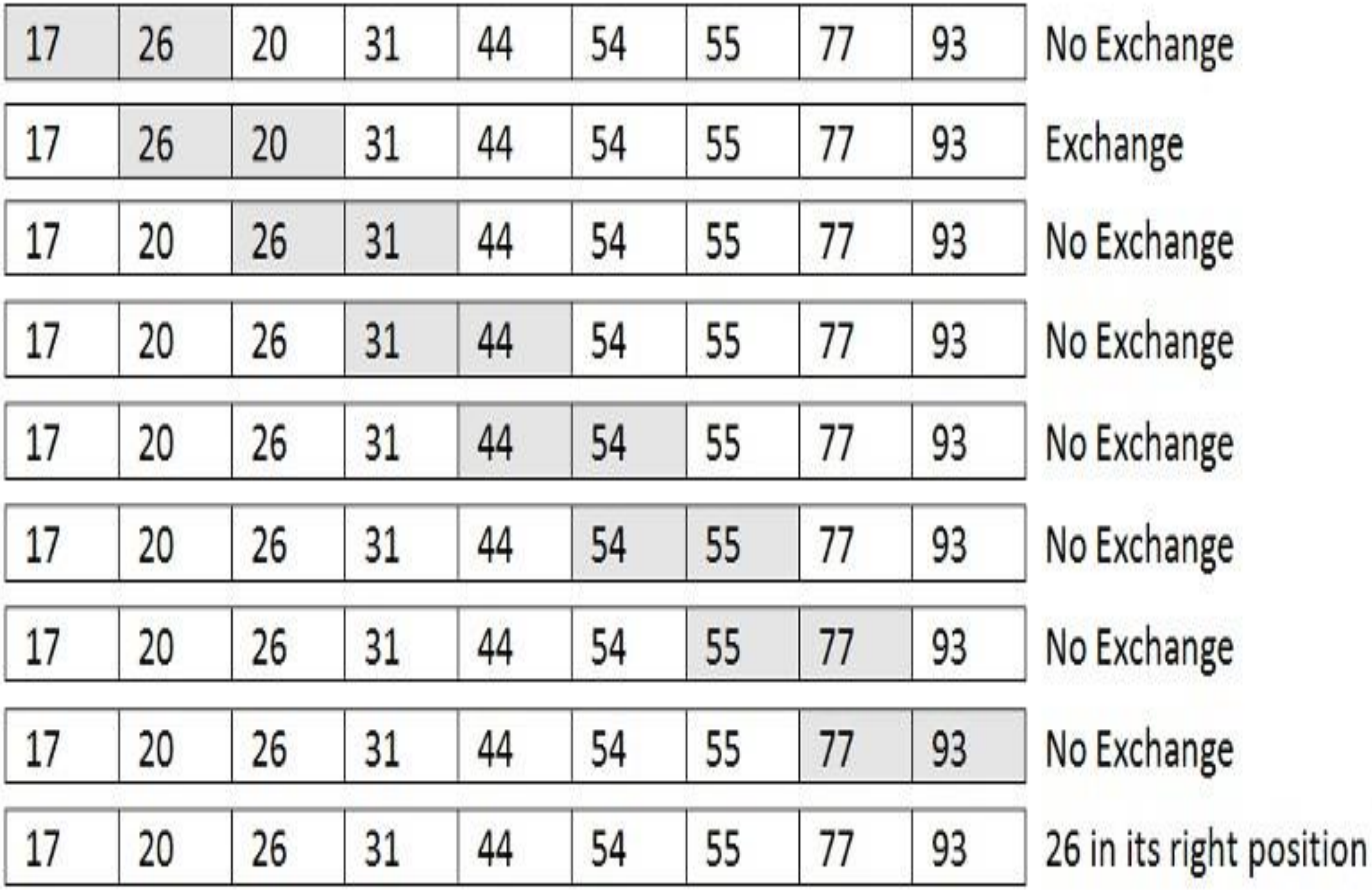
Pass 6



17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	No Exchange
17	26	31	20	44	54	55	77	93	Exchange
17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	31 in its right position

Cont..

Pass 7



17	26	20	31	44	54	55	77	93	No Exchange
17	26	20	31	44	54	55	77	93	Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	No Exchange
17	20	26	31	44	54	55	77	93	26 in its right position

Bubble Sort Algorithm

```
ALGORITHM  $BubbleSort(A[0..n-1])$
    //Sorts a given array by bubble sort
    //Input: An array $A[0..n-1]$ of orderable elements
    //Output: Array $A[0..n-1]$ sorted in non decreasing order
    for $i \leftarrow 0$ to $n-2$ do
        for $j \leftarrow 0$ to $n-2-i$ do
            if $A[j+1] < A[j]$
                swap $A[j]$ and $A[j+1]$
```

Time Complexity

worst case: $O(N^2)$
Best case: $O(N)$
Average case: $O(N^2)$

Outline

Internal Sorting Algorithms

1- Bubble Sort

➢ 2- Insertion Sort

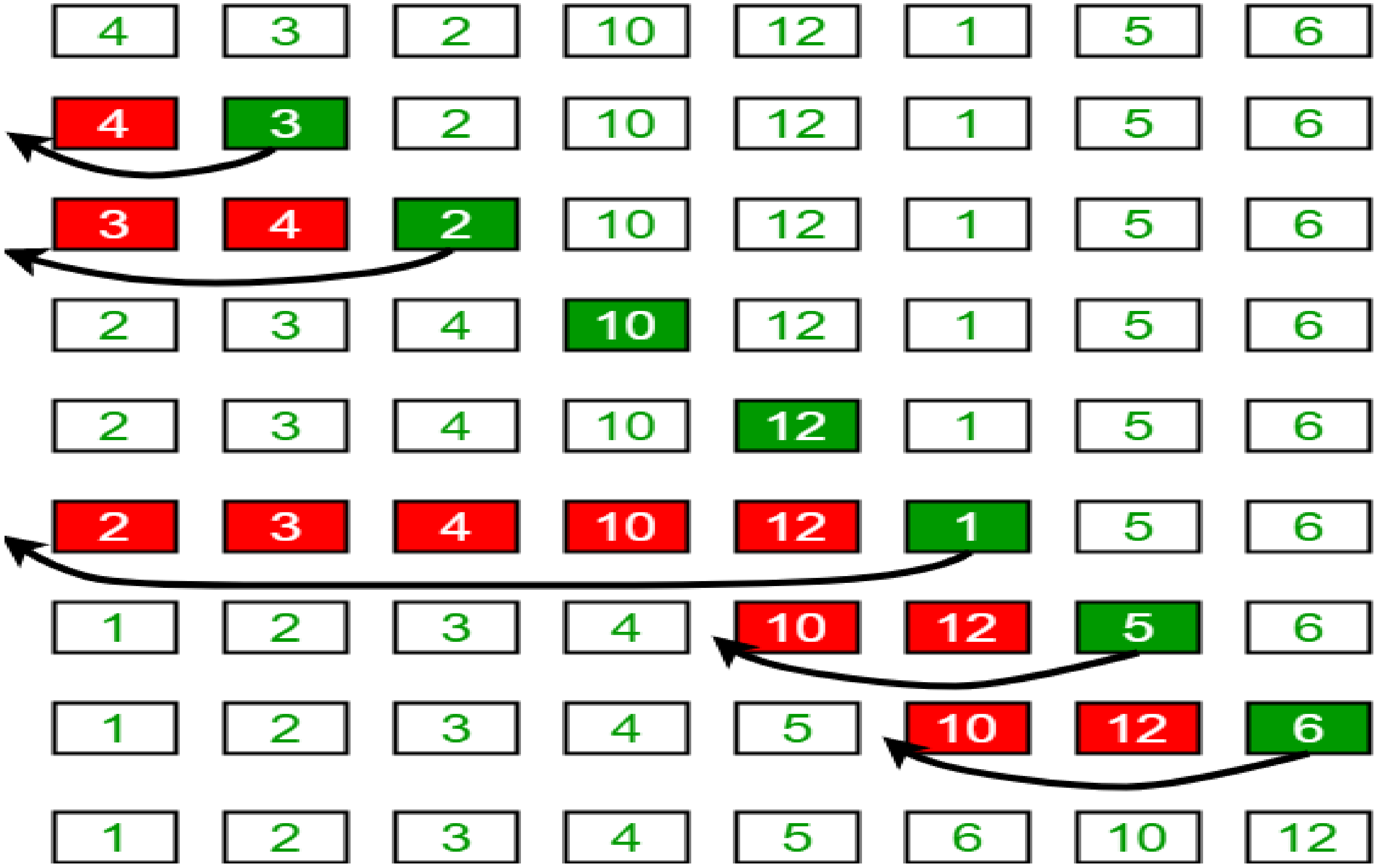
3- Selection Sort

4- Homework

GIF Example of Insertion Sort

Cont..

Insertion Sort Execution Example



4	3	2	10	12	1	5	6
4	3	2	10	12	1	5	6
3	4	2	10	12	1	5	6
2	3	4	10	12	1	5	6
2	3	4	10	12	1	5	6
2	3	4	10	12	1	5	6
1	2	3	4	10	12	5	6
1	2	3	4	5	10	12	6
1	2	3	4	5	6	10	12

Insertion Sort Algorithm

```
ALGORITHM   InsertionSort($A[0..n-1]$)
            //Sorts a given array by insertion sort
//Input: An array $A[0..n-1]$ of $n$ orderable elements
//Output: Array $A[0..n-1]$ sorted in nondecreasing order
 for $i \leftarrow 1$ to $n-1$ do
$v \leftarrow A[i]$
$j \leftarrow i-1$
while $j \geq 0$ and $A[j] > v$ do
            $A[j+1] \leftarrow A[j]$
$j \leftarrow j-1$
$A[j+1] \leftarrow v$
```

Time Complexity

worst case: $O(N^2)$

Best case: $O(N)$

Average case: $O(N^2)$

Outline

Internal Sorting Algorithms

1- Bubble Sort

2- Insertion Sort

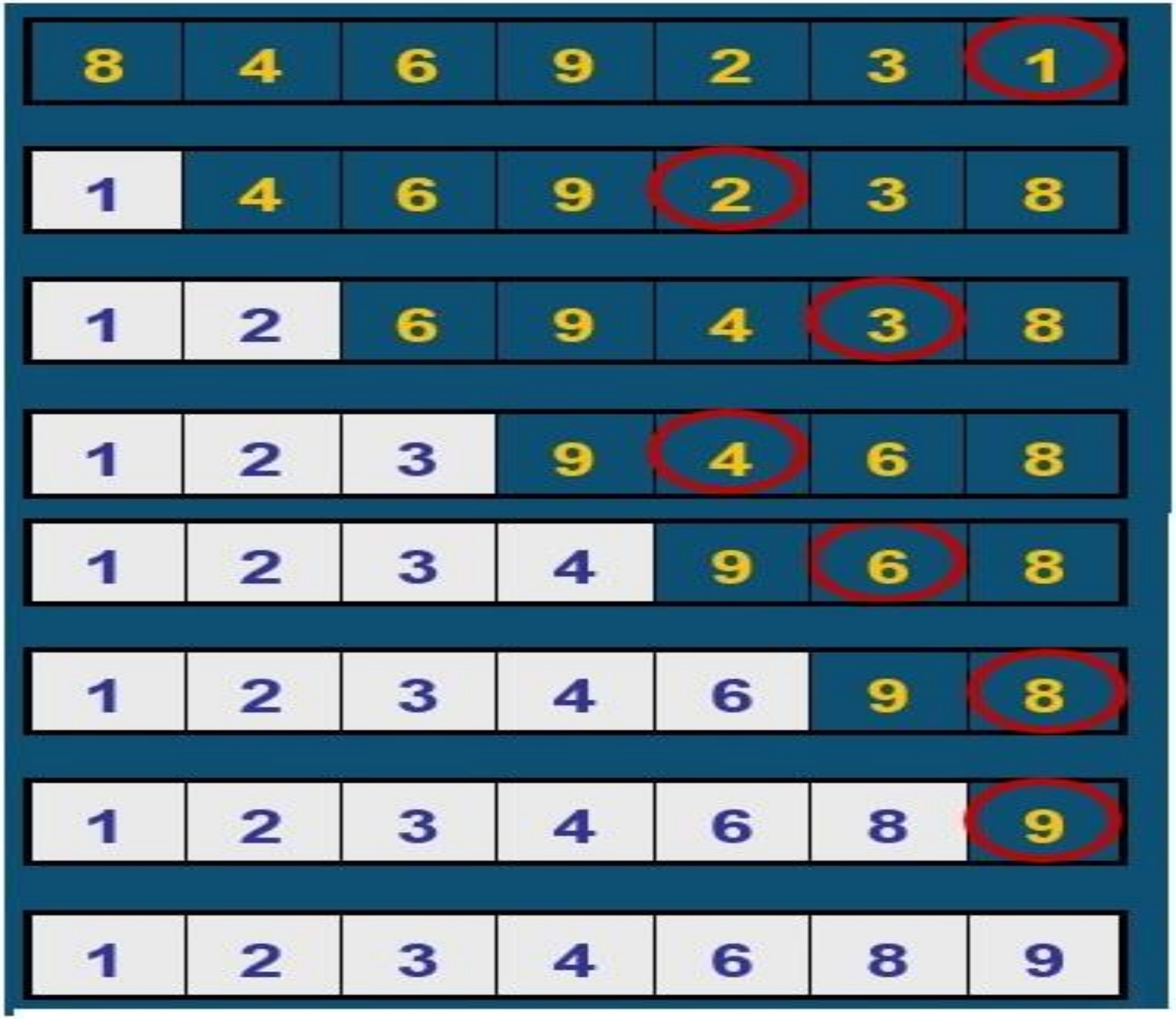
➢ 3- Selection Sort

4- Homework

Introduction to Selection sort

- We start selection sort by scanning the entire given list to find its smallest element and exchange it with the first element, putting the smallest element in its final position in the sorted list. Then we scan the list, starting with the second element, to find the smallest among the last $n - 1$ elements and exchange it with the second element, putting the second smallest element in its final position, After $n - 1$ passes, the list is sorted.

Example



8	4	6	9	2	3	1
1	4	6	9	2	3	8
1	2	6	9	4	3	8
1	2	3	9	4	6	8
1	2	3	4	9	6	8
1	2	3	4	6	9	8
1	2	3	4	6	8	9
1	2	3	4	6	8	9

Selection Sort Algorithm

```
ALGORITHM SelectionSort($A[0..n-1]$)
        //Sorts a given array by selection sort
//Input: An array $A[0..n-1]$ of orderable elements
//Output: Array $A[0..n-1]$ sorted in non-decreasing order
 for $i \leftarrow 0$ to $n-1$do
$min \leftarrow i$
for $j \leftarrow i+1$ to $n-1$ do
if $A[j] < A[min]$  $min \leftarrow j$
 swap $A[i]$ and $A[min]$
```

Time Complexity

worst case: $O(N^2)$

Best case: $O(N^2)$

Average case: $O(N^2)$

Outline

Internal Sorting Algorithms

1- Bubble Sort

2- Insertion Sort

3- Selection Sort

➢ 4- Homework

Homework

H.W// Write a complete analysis for Selection, Bubble and Insertion sort algorithms

THANK YOU