Design and Analysis of Algorithms

Dina El-Manakhly, Ph. D.

dina_almnakhly@science.suez.edu.eg

Some standard algorithms that follow Divide and Conquer algorithm

- Binary Search
- Merge Sort
- Quick Sort
- Closest Pair of Points
- Strassen's Algorithm (matrix multiplication)
- Finding maximum and minimum

Quick Sort Algorithm

❑ Definition

Problem Definition: Given an array A=(a_1, a_2,…, a_n) of n elements. Sorting the array is rearrangement the elements of the array such that $a_i \le a_{i+1}$, $1 \le i \le n-1$.

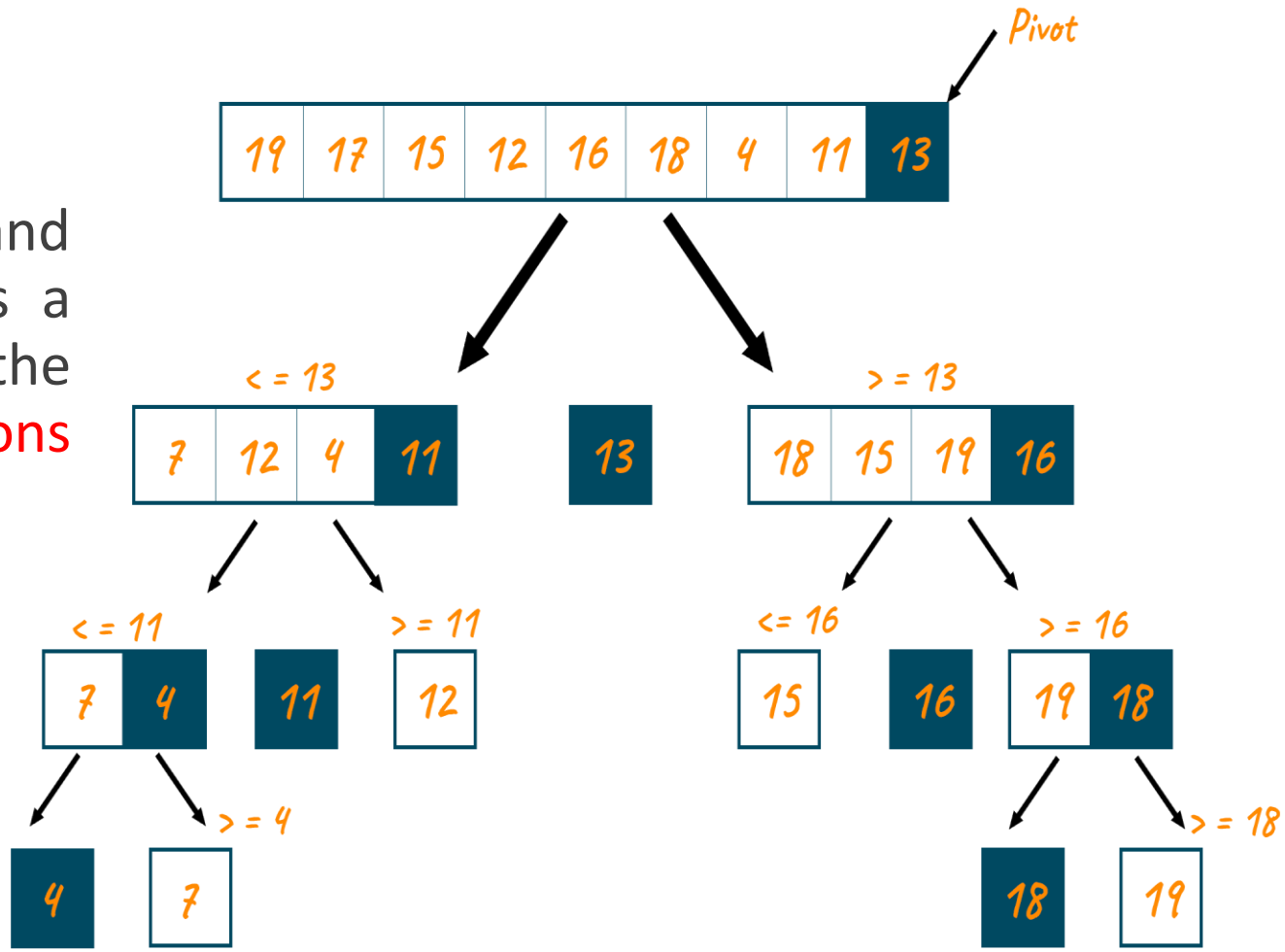
Example 1: Given A=(20,4,10,17,6,7,2)

Goal A=(2,4,6,7,10,17,20)

Quick Sort Algorithm

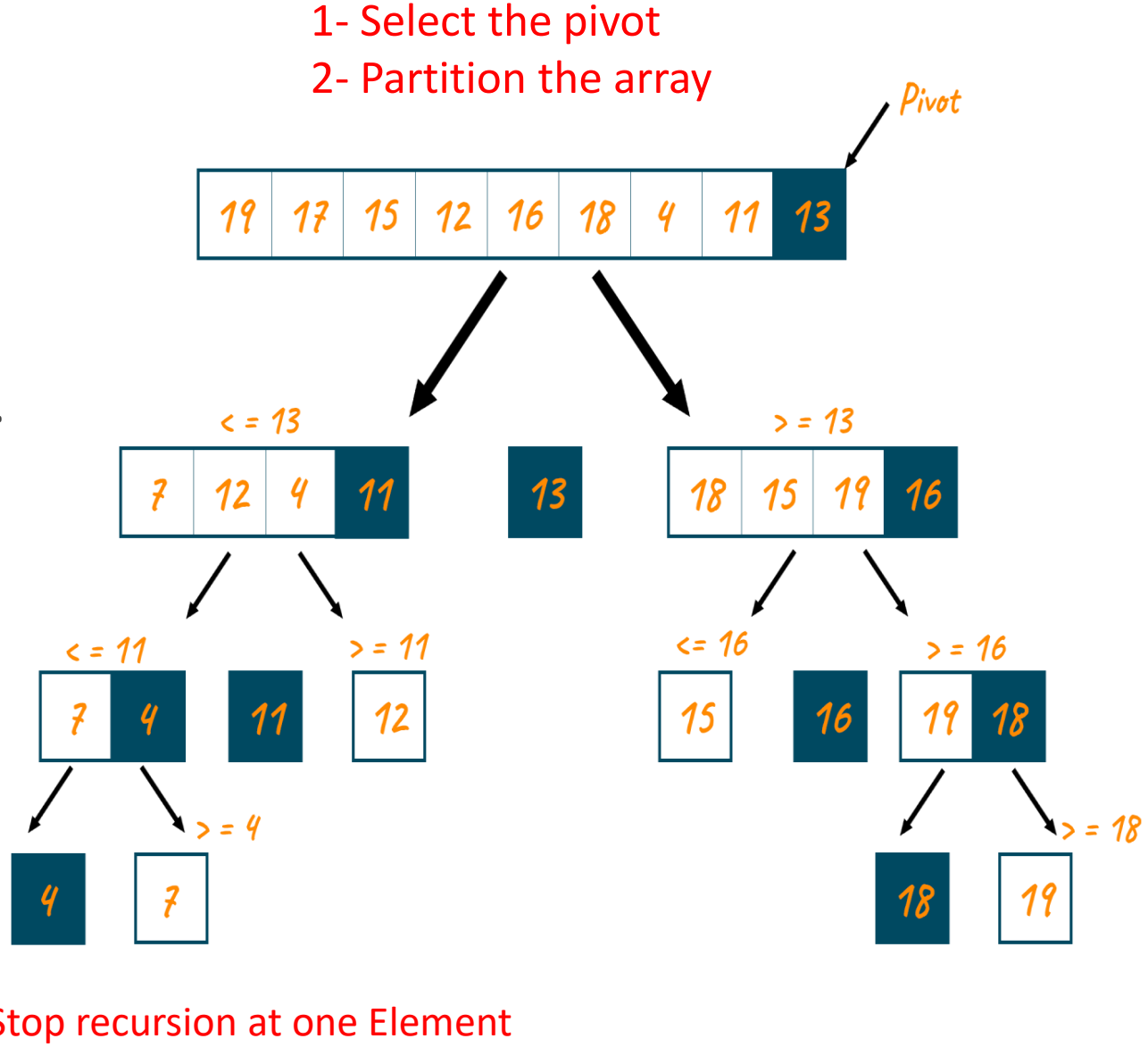
- Like Merge Sort, **QuickSort** is a Divide and Conquer algorithm. It picks an element as a pivot and partitions the given array around the picked pivot. There are many different versions of quickSort that pick pivot in different ways:

  - Pick the first element as a pivot.
  - Pick the last element as a pivot
  - Pick a random element as a pivot.
  - Pick median as the pivot.



Quick Sort Algorithm

- The key process in **quickSort** is a partition(). The target of partitions is, given an array and an element x of an array as the pivot, put x at its correct position in a sorted array and put all smaller elements (smaller than x) before x, and put all greater elements (greater than x) after x.



Partition Algorithm

Partitioning the array

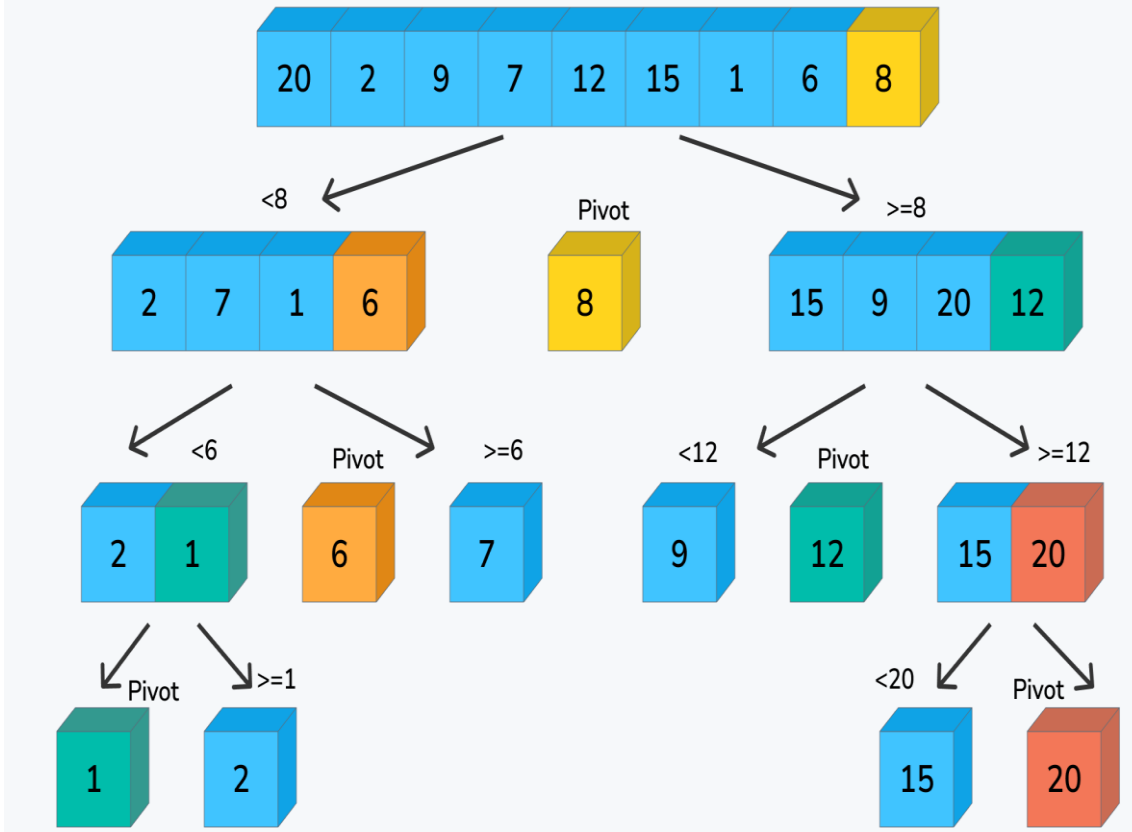
The key to the algorithm is the PARTITION procedure, which rearranges the subarray $A[p..r]$ in place.

```
PARTITION(A, p, r)
1  x = A[r]
2  i = p − 1
3  for j = p to r − 1
4      if A[j] ≤ x
5          i = i + 1
6          exchange A[i] with A[j]
7  exchange A[i + 1] with A[r]
8  return i + 1
```

https://www.youtube.com/watch?v=MZaf_9IZCrc

```
int partition(int arr[], int p, int r)
{


}
```




```
void quickSort(int arr[], int p, int r)
{
  if (p<r)
  {
    int pivot_index = partition(arr, p, r);
    quickSort(arr, p, pivot_index - 1); // recursive call on the left of pivot
    quickSort(arr, pivot_index + 1, r); // recursive call on the right of pivot
  }
}
```