



Computer Programming (b) - E1124

(Spring 2021-2022)

Lecture 2



Applications of Arrays (Searching and Sorting)

INSTRUCTOR

Dr / Ayman Soliman

➤ Contents

- Objectives
- List Processing
- Sequential Search
- Sorting a List: Bubble Sort
- Sorting a List: Selection Sort
- Sorting a List: Insertion Sort



➤ Objectives

- Learn how to implement the sequential search algorithm
- Explore how to sort an array using the bubble sort, selection sort, and insertion sort algorithms

➤ **List Processing**

➤ **List**: a set of values of the same type

□ Basic list operations:

- a) Search for a given item
- b) Sort the list
- c) Insert an item in the list
- d) Delete an item from the list

➤ Searching

- To search a list, you need
 - a) The list (array) containing the list
 - b) List length
 - c) Item to be found
- After the search is completed
 - d) If found,
 - ✓ Report “success”
 - ✓ Location where the item was found
 - e) If not found, report “failure”

➤ Sequential Search

- Sequential search: search a list for an item
- Compare search item with other elements until either
 - Item is found
 - List has no more elements left
- Average number of comparisons made by the sequential search equals **half** the list size
- Good only for very short lists

➤ Sequential Search (cont.)

```
int seqSearch(const int list[], int listLength, int searchItem)
{
    int loc;
    bool found = false;

    for (loc = 0; loc < listLength; loc++)
        if (list[loc] == searchItem)
        {
            found = true;
            break;
        }

    if (found)
        return loc;
    else
        return -1;
}
```

➤ **Sorting a List: Bubble Sort**

➤ Suppose $\text{list}[0] \dots \text{list}[n - 1]$ is a list of n elements, indexed 0 to $n - 1$

➤ Bubble sort algorithm:

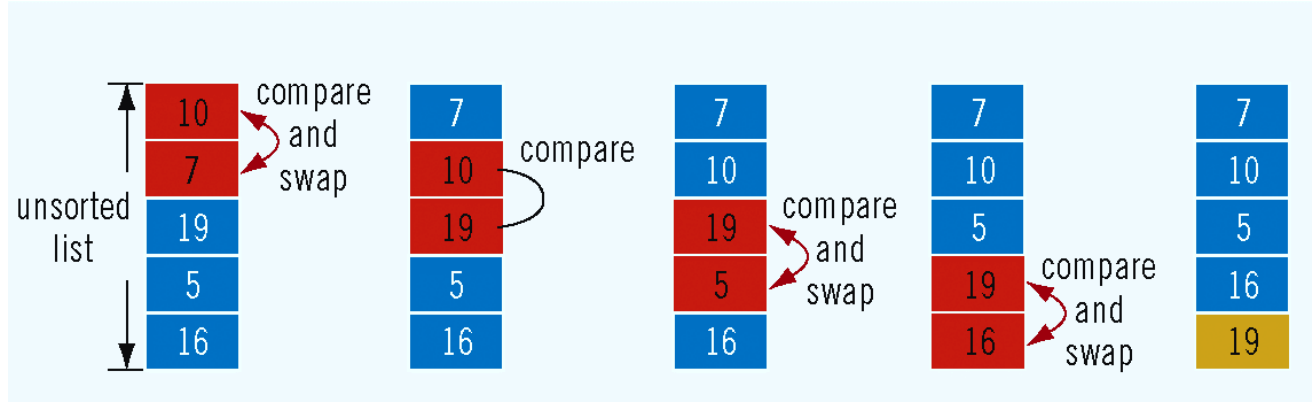
❑ In a series of $n - 1$ iterations, compare successive elements, $\text{list}[\text{index}]$ and $\text{list}[\text{index} + 1]$

❑ If $\text{list}[\text{index}]$ is greater than $\text{list}[\text{index} + 1]$, then swap them

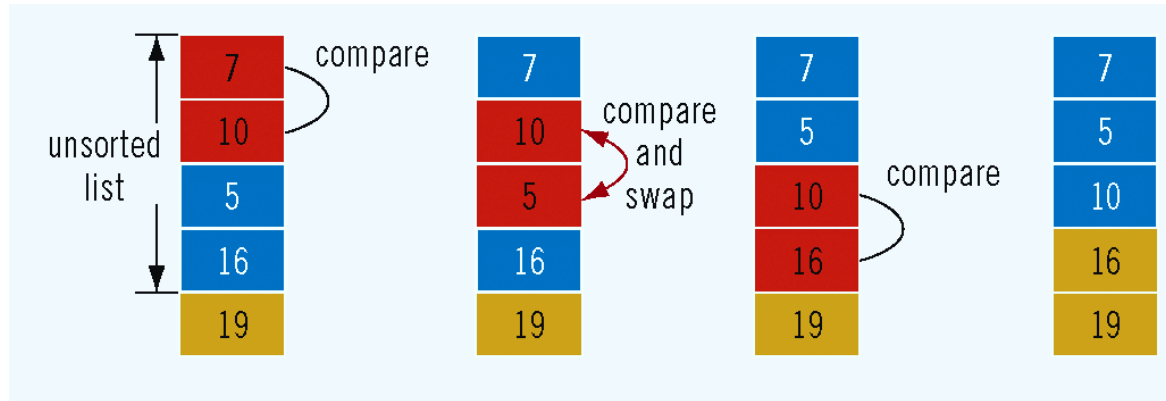
➤ Example

```
list
list[0] 10
list[1] 7
list[2] 19
list[3] 5
list[4] 16
```

List of five elements

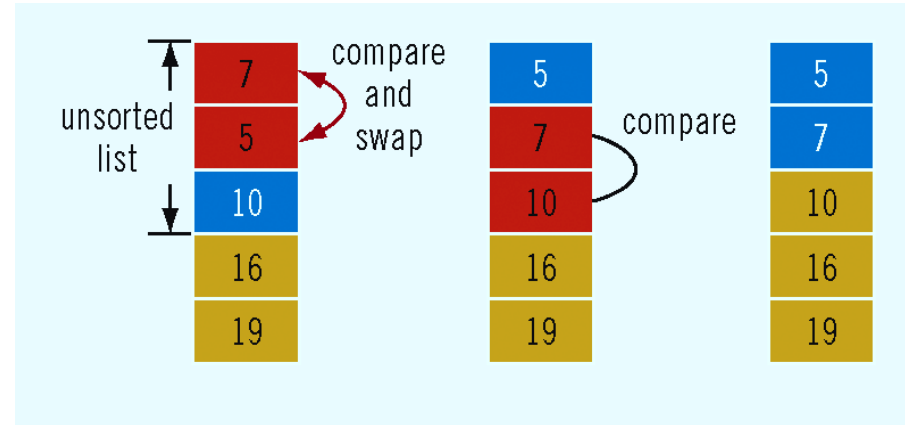


First iteration

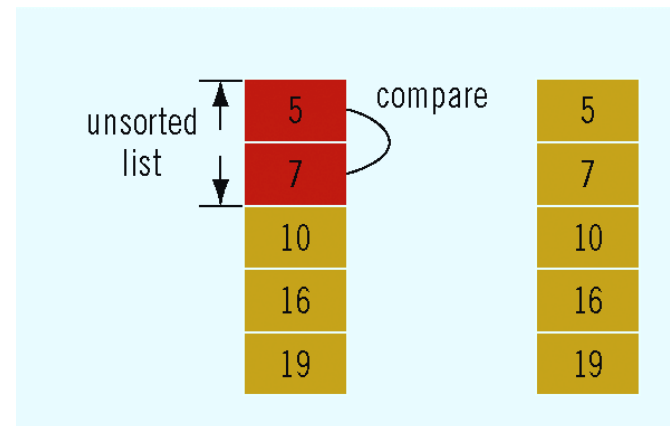


Second iteration

➤ Example (cont.)



Third iteration



Fourth iteration

➤ Bubble Sort Code

```
void bubbleSort(int list[], int length)
{
    int temp;
    int iteration;
    int index;
    for (iteration = 1; iteration < length; iteration++)
    {
        for (index = 0; index < length - iteration; index++)
            if (list[index] > list[index + 1])
            {
                temp = list[index];
                list[index] = list[index + 1];
                list[index + 1] = temp;
            }
    }
}
```

Thank
you

