

Sorting is the operation of ordering a list of items in a well-defined order. A lot of sorting algorithms have been developed over the years such as: Selection Sort, Bobble Sort and Insertion Sort which discusses in this lab.

### Insertion Sort Algorithm

In the Insertion Sort array is divided into two parts: sorted and unsorted. At the beginning, sorted part contains first element of the array and unsorted one contains the rest. At every step, algorithm takes element in the unsorted part and insert it to the right place of the sorted one. When unsorted part becomes empty, algorithm stops. As the follow:



become



This approach shift element only to the final correct position instead of swapping approach. it is the most commonly used.

#### Algorithm InsertionSort( $A$ ):

**Input:** An array  $A$  of  $n$  comparable elements

**Output:** The array  $A$  with elements rearranged in non-decreasing order

**for**  $i \leftarrow 1$  to  $n - 1$  **do**

    {Insert  $A[i]$  at its proper location in  $A[0], A[1], \dots, A[i - 1]$ }

$cur \leftarrow A[i]$

$j \leftarrow i - 1$

**while**  $j \geq 0$  and  $a[j] > cur$  **do**

$A[j + 1] \leftarrow A[j]$

$j \leftarrow j - 1$

$A[j + 1] \leftarrow cur$  { $cur$  is now in the right place}

As can be seen the insertion sort algorithm uses two loops to sort the array, the outer one responsible for the index of array, and the inner one for comparison operation. If the array is already sorted the inner loop does one comparison which is increase the efficiency.