

## Einsendeaufgabe 3

1- Consider the following Pseudo-Code for the bubblesort algorithm:

```

Bubble-Sort II(A)
1  m ← n - 1
2  do
3      b ← false
4      for i ← 1 to m
5          do if A[i] > A[i + 1]
6              then Vertausche(A[i];A[i + 1])
7                  b ← true
8                  k ← i
9      m ← k - 1
10 while b
11 return A

```

Show how the algorithm works with the array (2, 4, 11, 12, 7, 15, 19) . In particular show how the values of m, i and k are changing, which elements are compared and which elements are exchanged, see script p. 50 example 4.7.

2- Implement this algorithm in a class called `MyArrays` . Do the necessary changes as in Java (and other programming languages) arrays begin with the index 0 and have a reference type. You should implement a method with the following signature so that the number of **comparisons** is returned:

```
public static int bubbleSort(int[] ar)
```

Test your implementation in a class called `MyArraysTest` with an empty array, an array is one element only and arrays with more than 1 element.

Please do follow the algorithm given here. In the literature you may find versions that are slightly different and they may give different results for question 3 below.

3- Test your method with the file `QuickSort.txt` Copyright *Tim Roughgarden*, Stanford University.

For that used the 2 static methods in the class called `UtilitiesArraysFiles` . You can also write your own methods if you wish. One method reads the numbers from the textfile in an array, and the other writes the numbers of the array in a textfile whose name is given as a parameter. Theses methods have the following headers:

```
public static int[] readInArray(String filename)
```

```
public static void writeArrayToFile(int [] ir, String filename).
```

How many comparisons do you get? Does this number fit the theoretical result  $O(n^2)$  (in fact  $n(n-1)/2$ , see script p. 50) in the worst case or  $O(n)$  in the best case?

Send your source code and your answer in a .zip file called `NamenESA3.zip`