Joanna Klukowska
joannakl@cs.nyu.edu

DNHI Homework 6
Searching and Sorting

Problem 1

Write a method that provides recursive implementation for selection sort.

Problem 2

Given the following array of integers, show what the array will look like after each iteration of the outer loop of insertion sort. Indicate where the "sorted" part ends after each iteration. The elements should be sorted from smallest to largest.

index	0	1	2	3	4	5	6	7	8	9
item	7	32	12	3	0	64	23	12	3	45

Problem 3

Apply quick sort to the following integer array. The array should be sorted from smallest to largest. The index of the pivot is selected to be `(first+last)/2` where first and last are the indexes of the `first` and `last` elements in the considered partition (do not use the median of three pivot selection that we discussed in class). The pivot should be moved to the last position before the partitioning step. After the partition it should be moved back to its proper position. You should show the following steps:

- content of the array after pivot is moved to the last position,
- content of the array after completing partitioning and moving the pivot to its proper position (indicate the position of the pivot in bold and/or in color).

index	0	1	2	3	4	5	6	7	8	9	10
value	12	13	27	65	43	59	45	67	7	5	55

Problem 4

Apply merge sort to the following integer array. The array should be sorted from smallest to largest. Show all the steps of splitting the array and then merging the pieces.

index	0	1	2	3	4	5	6	7	8	9	10
value	2	76	27	62	43	59	45	87	13	5	99

Problem 5

Write an implementation of the generic method that given two sorted arrays, merges them into a single array and returns the array containing all elements in sorted order. The original arrays passed as a parameter should not be modified.
What do you need to assume about the generic type?

Problem 6

Write an implementation of the generic method that given a sorted **`ArrayList<E>`** object removes all repeated elements (leaving only one copy of each element). Make sure your method is efficient - it should run in linear time (not quadratic)!
What do you need to assume about the generic type?

Joanna Klukowska
joannakl@cs.nyu.edu

Problem 7

Given an array of n elements, specify an algorithm for finding mode (the element which apprears most often). What is the complexity of your solution? Can you think of a different way of achieving the same task? What is teh complexity of the other solution?

Extra Challenge [1]

Write a sort method that sorts an array of strings so that all of the anagrams are next to one another.

[1] Not for the quiz, but maybe useful for a job interview.