Algorithms and Data Structures
Spring 2018
Exercises for Units 10 and 11

1. (a) How, if at all, can mergesort be implemented so that it is stable?
   (b) How, if at all, can quicksort be implemented so that it is stable?

2. For the purpose of this question a string is a pair $(k, A[0..k-1])$ where $k \geq i$ is an integer denoting the length of the string and $A[0..k-1]$ is an array of k characters, which you may assume to be $\{0, \ldots, b-1\}$ with b constant.

   (a) Develop an algorithm that sorts a given set S of n strings in time $O(n+m)$, where m is the sum of the lengths of the strings in S.

   Strings are to be compared lexicographically, with $A[0]$ being the most significant position.

   (b) What running time can you prove if b is not a constant?

3. Give a family of examples that show that the number of rotations that can happen during a deletion from an AVL-tree is proportional to the height of the tree.

4. Let T' and T'' be AVL-trees that store a sets S' and S'' of items, respectively. Assume that all the items in S' have keys smaller than all the items in S''.

   Describe an efficient method for computing an AVL-tree T for $S = S' \cup S''$. Your algorithm may destroy T' and T''.

   What is the running time of your algorithm?