There are more sorting-related problems in TRAKLA2.

1. We have seen the procedure for merging two sorted lists into one in time $O(n)$. Give an algorithm for merging $k$ sorted lists in time $O(n \log k)$. **Hint:** priority queue.

2. How does the `PARTITION` procedure work, if the array under consideration
   (a) is initially in an increasing order
   (b) is initially in a decreasing order
   (c) contain only copies of the same key value?

3. Suppose `PARTITION` always puts $\alpha k$ elements into the smaller part and $(1 - \alpha)k$ elements into the larger part, where $k$ is the size of the array and $0 < \alpha < 1/2$ is a constant. What is roughly the (a) largest (b) smallest depth of a leaf in the recursion tree? (Ignore the issue of integer rounding.)