Algorithms

Homework 2

Due January 22, by midnight

Instructions:
Include pseudo-code for your algorithms, a run-time analysis and a proof of correctness.
Also, read these instructions carefully: [http://courses.cs.washington.edu/courses/cse417/16wi/grading.html](http://courses.cs.washington.edu/courses/cse417/16wi/grading.html)

Turn-in:

- Turn in problems 1 and 2 here: [https://canvas.uw.edu/courses/1021503/assignments/3138258](https://canvas.uw.edu/courses/1021503/assignments/3138258)
- Turn in problem 3 here: [https://canvas.uw.edu/courses/1021503/assignments/3140572](https://canvas.uw.edu/courses/1021503/assignments/3140572)

Problems:

1. Problem 2.27 on page 83 of [DPV].

2. You are interested in analyzing some hard-to-obtain data from two separate databases. Each database contains \( n \) numerical values – so there are \( 2n \) values total – and you may assume that no two values are the same. You’d like to determine the median of this set of \( 2n \) values, which we will define here to be the \( n^{th} \) smallest value.

   However, the only way you can access these values is through queries to the databases. In a single query, you can specify a value \( k \) to one of the two databases, and the chosen database will return the \( k^{th} \) smallest value it contains. Since queries are expensive, you’d like to compute the median using as few queries as possible.

   Give an algorithm that finds the median value using at most \( O(\log n) \) queries.

3. Recall the problem of finding the number of inversions: You are given a sequence of \( n \) numbers \( a_1, \ldots, a_n \), which we are assume are all distinct, and we define an inversion to be a pair \( i < j \) such that \( a_i > a_j \).

   We motivated the problem of counting inversions as a good measure of how different two orderings are. However, one might feel that this measure is too sensitivie. Let’s call a pair a significant inversion if \( i < j \) and \( a_i > 2a_j \). Give an \( O(n \log n) \) algorithm to count the number of significant inversions between two orderings.