Advanced Algorithms 2014. Exercises 3-4

Exercise 3.
This is an open-end discussion exercise about an application.

Imagine a landscape in the mountains. (No, this was not yet the exercise. Read further :-) A set \( P \) of \( n \) points in this mountain landscape is marked as “important”. A point \( q \) is visible from a point \( p \) if there is no obstacle (e.g., some mountain) on the straight-line segment from \( p \) to \( q \). Trivially, every point \( p \) is visible from itself. We wish to place a minimum number \( k \) of guards on \( k \) points of \( P \), such that every point in \( P \) is visible from the point of at least one guard.

3.1. Discuss: How would you approach this problem, and what kind of complexity results can you provide? (Hardness results? Exact polynomial-time algorithm? Some approximation? Something else?) Be as comprehensive and precise as you can, and feel free to add own reflections.

3.2. We say that point \( p \) dominates point \( q \) if every point in \( P \) visible from \( q \) is also visible from \( p \). (For instance, if you go uphill from \( q \) to \( p \), then often the higher point \( p \) dominates \( q \), but not always.)

Discuss: Is it worthwhile to determine the dominated points first? In particular: How long does that take? Does it help in some way, when it comes to the original problem?

See reverse page.
Exercise 4 (optional).
The following problem, a bit simplified though, arises when we want to label places on a geographic map by their names. Every possible name tag occupies a given axis-parallel rectangle with a fixed height 1. (We take the font size as the length unit.) The problem is to select as many as possible name tags that are pairwise disjoint, to ensure that users of the map can read all names. This problem version does not allow to move or shrink name tags on the map; we may only use the given positions. Name tags that do not fit may still be put on the margin of the map.

Formal description of the problem: Let $R$ be a given set of axis-parallel rectangles. All have height 1, but the horizontal lengths are arbitrary. Select a maximum subset of rectangles from $R$ that are pairwise disjoint.

In the special case that all rectangles of $R$ intersect a fixed horizontal line, the problem is easy to solve to optimality, by a polynomial-time greedy algorithm $G$. You may assume that $G$ is already available as a black box, so we do not specify $G$ here. Now $G$ can be used to build an approximation algorithm for the general case. It starts as follows. (Drawing some pictures may help understand it.)

We select a set $L$ of horizontal lines that hit all rectangles of $R$, and whose vertical distances are larger than 1. Such a set $L$ is very easy to construct, think about the details. For each line $\ell \in L$, let $R(\ell)$ denote the subset of rectangles of $R$ hit by $\ell$. For each $R(\ell)$ we compute an optimal solution using $G$. Note that rectangles from $R(\ell)$ can still overlap rectangles that are hit by the two neighbors of $\ell$ in $L$, but they cannot overlap more distant rectangles.

Your task is: Finish the proposed algorithm such that it returns a correct solution that contains at least half the optimal number of pairwise disjoint rectangles. Do not forget to prove the approximation ratio $1/2$ and to argue why the algorithm runs in polynomial time.