CSE 5050: Basics of Algorithmic Thinking Handout

Sample Algorithm Description and Proof of Correctness

In the video lectures of Module 1, we saw two algorithm descriptions. One (insertion sort) was expressed as pseudocode and the other (Gale-Shapley) was structured like pseudocode but written largely using English. Here, we provide an example of an algorithm description that is written completely using just plain English. When deciding which of these three formats to use for describing an algorithm, you should choose the one that is easiest to write while being precise and concise.

Question

Consider a straight road with n houses scattered sparsely along it. Your goal is to place cell phone base stations at certain points along the road so that every house is within 4 miles of one of the base stations. Give an efficient algorithm for this problem that uses as few base stations as possible.

Solution

Algorithm description in plain English: Assume the road goes west-east and we start at the west end. For any point on the road, define its *position* to be the number of miles it is from the western end. We place the first base station at the easternmost (largest) position s_1 with the property that all houses between 0 and s_1 will be covered by s_1. We now remove all house covered by s_1 and repeat until all houses are covered. In general, having placed $\{s_1, \dots, s_i\}$ we place base station $i+1$ at the largest position s_{i+1} with the property that all houses between s_i and s_{i+1} will be covered by s_i and s_{i+1}.

Proof of Correctness: Let $S = \{s_1, \dots, s_k\}$ denote the full set of base station positions returned by our greedy algorithm. Let $T = \{t_1, \dots, t_m\}$ denote the set of base station positions in an optimal solution. We'll assume that these positions are sorted in increasing order (i.e., west to east). We will show that $k = m$.

We'll do this by showing that our greedy algorithm "stays ahead" of the optimal solution T. Specifically, we claim that $s_i \geq t_i$ for each i, and prove this by induction. The claim is true for $i = 1$ since we place the base station as far as possible to the east. Assume that it is true for some value $i \geq 1$. This means that our algorithm's first i stations $\{s_1, \dots, s_i\}$ cover all the houses covered by the first i stations $\{t_1, \dots, t_i\}$. As a result, if we add t_{i+1} to $\{s_1, \dots, s_i\}$ we will not leave any house between s_i and t_{i+1} uncovered. However, since our greedy algorithm chooses s_{i+1} to be as large as possible while still covering all houses between s_i and s_{i+1}, we must have $s_{i+1} \geq t_{i+1}$. This proves the claim by induction.

Finally, if $k > m$, then $\{s_1, \dots, s_m\}$ fails to cover all houses. But $s_m > t_m$ and so $\{t_1, \dots, t_m\}$ would also fail to cover all houses, a contradiction. Thus, we must have $k = m$, as was to be shown.

(While we haven't yet studied how to perform worst case complexity analysis, we provide such an analysis below for completeness.)

Complexity Analysis: We can sort the positions of the n houses west to east in $O(n \log n)$ time. Once we have this west-to-east ordering, our greedy algorithm simply goes through the list of houses in order and therefore just takes time linear in the number of houses. The total time complexity is thus $O(n \log n)$.