CS3000: Algorithms & Data
Drew van der Poel

Recitation 5:

- Greedy Algorithms

June 7, 2021

Problem 1

A fencing tournament involving Blue and Red teams is matching n Blue players with n Red players. We would like to match the players n pairs so that each player is in exactly one pair. Let the height of the ith Blue player be b_i and the height of the ith red player be r_i. Ideally, we should pair each blue player with a red player whose heights match as closely as possible. Globally, we would like to compute a matching in which the sum, over all pairs, of the absolute differences of the heights of the players in the pair, is minimized.

Design a greedy polynomial-time algorithm for the problem. Prove the correctness of your algorithm. Analyze its worst-case running time.

(*Hint:* Start with two Blue players and two Red players. How would you match them? Continue to three Blue and three Red players, and identify a strategy.)

R	60	56
B	58	53

Idea: Pair the i^{th} tallest blue player w/ the i^{th} tallest red player

2 sorts of size n

R/t: $O(n \log n)$

Pf. (Exchange Argument)

Let O be some non-greedy solution,
Let i be the smallest value s.t. i^{th} blue player is not paired w/ the i^{th} red player.

$$O \quad \begin{matrix} B_1 & B_2 & B_3 & \cdots & B_i = B & B' \\ | & | & | & & & \\ R_1 & R_2 & R_3 & \cdots & R_i & R' \\ & & & & \shortparallel & \\ & & & & R & \end{matrix}$$

(WLOG $B \geq R$)

Know: $B \geq B'$, $R \geq R'$

heights

WTS: $|B - R| + |B' - R'| \leq |B - R'| + |R - B'|$

Case 1: $\underline{B \geq \boxed{R \geq B'} \geq R'} \rightarrow |B-R| + |B'-R'| \leq |B-R'| + |R-B'|$

$$\cancel{B} - R + B' - \cancel{R'} \leq \cancel{B} - \cancel{R'} + R - B'$$

$$2B' \leq 2R \checkmark$$

Case 2: $B \geq B' \geq R \geq R'$

$$|B-R| + |B'-R'| \leq |B-R'| + |R-B'|$$

$$\cancel{B}-\cancel{R} + \cancel{B'}-\cancel{R'} \leq \cancel{B}-\cancel{R'} + \cancel{B'}-\cancel{R}$$

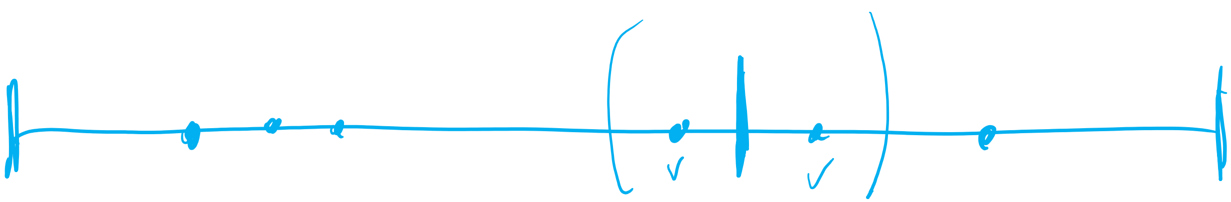
$$0 \leq 0$$

Case 3: $B \geq R \geq R' \geq B'$

Problem 2

A wireless communications company is planning to place cell phone base stations along a long, quiet country road, with houses scattered very sparsely along it. You can view the road as a line running left to right, with an leftmost endpoint, a rightmost endpoint, and points on the line indicating the locations of the houses. The company would like to install base stations so that every house is within two miles of one of the base stations.

Give an efficient algorithm that achieves this goal, using as few base stations as possible. Assume that base stations can be installed at any point on the line.



Idea: • place a station d miles to the right of 1st uncovered house
• repeat until all houses are covered

Inputs: $h_1, h_2, \ldots, h_n$ // locations of houses

recent = $h_1 + d$
solution = {recent}
For $i = 1, \ldots n$
    if $|h_i - \text{recent}| > d$:
        //Place station @ $h_i + d$
        recent = $h_i + d$
        solution.add(recent)

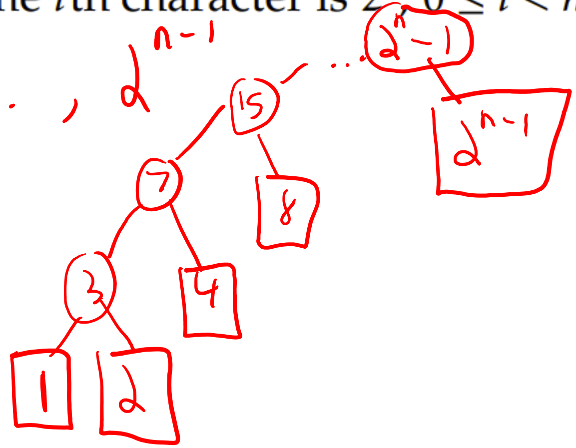
return solution

R/T: $O(n)$
(assuming houses are given in order)

Problem 3

(a) *(Recitation)* Determine the Huffman code for a set of n characters with the counts given as follows: the count of the ith character is 2^i, $0 \le i < n$.

$1, 2, 4, \ldots, 2^{n-1}$



(b) *(Recitation)* True or False: In any Huffman code for a character set with at least 2 elements, there will exist at least two characters that will have the same length code. Justify your answer.

True – 2 lowest freq characters will always have same length encoding

(c) *(Recitation)* Determine the Huffman code for a set of 8 characters in which the characters are about equally common; i.e., the count of the most common character is less than twice the count of the least common.

$C_1 \le C_2 \le \dots \le C_8$

C_i = count of char. i

$2C_1 > C_8$

$C_B \ge$

$C_A = C_1 + C_2 > C_8 \ge C_7 \dots \ge C_3$

$C_D \ge C_C \ge C_B \ge C_A$

$C_X = C_A + C_B > 2C_8 \ge C_D \ge C_C$