

# CSCI 3110 Tutorial 8

Reviewed July 12, 2019

Consider the following game. A “dealer” produces a sequence  $s_1 \dots s_n$  of “cards”, face up, where each card  $s_i$  has a value  $v_i$ . Then two players take turns picking a card from the sequence, but can only pick the first or the last card of the (remaining) sequence. The goal is to collect cards of largest total value. (For example, you can think of the cards as bills of different denominations.) Consider the score from the perspective of the first player, that is, the sum of the cards the first player has chosen minus the sum of the cards the second player has chosen. Assume  $n$  is even.

- (a) Show a sequence of cards such that it is not optimal for the first player to start by picking up the available card of larger value. That is, the natural greedy strategy is suboptimal. Why does the greedy algorithm fail on your sequence and what is the optimal strategy on your sequence?
- (b) Describe an  $O(n^2)$  algorithm to compute an optimal strategy for the first player. Given the initial sequence, your algorithm should precompute in  $O(n^2)$  time some information, and then the first player should be able to make each move optimally in  $O(1)$  time by looking up the precomputed information. Assume that the second player will always make an optimal move: after every move the first player makes the second player will make the move that reduces the final score by as much as possible.

Describe the problem, give a recurrence for the subproblems, and then explain the dynamic programming order. You do not need to provide pseudocode but explain how Player 1 chooses what move to make from your precomputed information.