CSCI 270 - Spring 2023 - HW 7

Due: March 8, 2023

1. [20 points] Solve Kleinberg and Tardos, Chapter 6, Exercise 5.
   (a) Define (in plain English) subproblems to be solved. (4 pts)

   (b) Write a recurrence relation for the subproblems (6 pts)

   (c) Using the recurrence formula in part b, write pseudocode to find the maximum total quality among all segmentation possibilities. (5 pts)

   (d) Make sure you specify
      i. base cases and their values (2 pts)

      ii. where the final answer can be found (1 pt)

   (e) What is the complexity of your solution? (2 pts)

2. [20 points] You are given an integer array $a[1], \ldots, a[n]$, find the contiguous subarray (containing at least one number) which has the largest sum and only returns its sum. The optimal subarray is not required to return or compute. Taking $a = [5, 4, -1, 7, 8]$ as an example: the subarray [5] is considered as a valid subarray with sum 5, though it only has one single element; the subarray $[5, 4, -1, 7, 8]$ achieves the largest sum 23; on the other hand, $[5, 4, 7, 8]$ is not a valid subarray as the numbers 4 and 7 are not contiguous.
   (a) Define (in plain English) subproblems to be solved. (4 pts)

(b) Write a recurrence relation for the subproblems. (6 pts)

(c) Using the recurrence formula in part b, write pseudocode to find the subarray (containing at least one number) which has the largest sum. (5 pts)

(d) Make sure you specify

i. base cases and their values (2 pts)

ii. where the final answer can be found (1 pt)

(e) What is the complexity of your solution? (2 pts)

3. [20 points] You are given an array of positive numbers $a[1], \ldots, a[n]$. For a subsequence $a[i_1], a[i_2], \ldots, a[i_t]$ of array a (that is, $i_1 < i_2 < \cdots < i_t$): if it is an increasing sequence of numbers, that is, $a[i_1], a[i_2], \ldots, a[i_t]$, its happiness score is given by

$$\sum_{k=1}^{t} k \times a[i_k]$$

Otherwise, the happiness score of this array is zero.
For example, for the input $a = [22, 44, 33, 66, 55]$, the increasing subsequence $[22, 44, 55]$ has happiness score $(1) \times (22) + (2) \times (44) + (3) \times (55) = 275$; the increasing subsequence $[22, 33, 55]$ has happiness score $(1) \times (22) + (2) \times (33) + (3) \times (55) = 253$; the subsequence $[33, 66, 55]$ has happiness score 0 as this sequence is not increasing. Please design an efficient algorithm to **only** return the highest happiness score over all the subsequences.

(a) Define (in plain English) subproblems to be solved. (4 pts)

(b) Write a recurrence relation for the subproblems (6 pts)

(c) Using the recurrence formula in part b, write pseudocode to find the highest happiness score over all the subsequences. (5 pts)

(d) Make sure you specify

i. base cases and their values (2 pts)

ii. where the final answer can be found (1 pt)

(e) What is the complexity of your solution? (2 pts)

4. [20 points] You've started a hobby of retail investing into stocks using a mobile app, RogerGood. You magically gained the power to see N days into the future and you can see the prices of one particular stock. Given an array of prices of this particular stock, where $prices[i]$ is the price of a given stock on the i^{th} day, find the maximum profit you can achieve through various buy/sell actions. RogerGood also has a fixed fee per transaction. You may complete as many transactions as you like, but you need to pay the transaction fee for each transaction (only pay once per pair of buy and sell). Assume you can own at most one unit of stock.

   (a) Define (in plain English) subproblems to be solved. (4 pts)

   (b) Write a recurrence relation for the subproblems (6 pts)

   (c) Using the recurrence formula in part b, write pseudocode to solve the problem. (5 pts)

   (d) Make sure you specify

      i. base cases and their values (2 pts)

      ii. where the final answer can be found (1 pt)

   (e) What is the complexity of your solution? (2 pts)