Sabancı University
Faculty of Engineering and Natural Sciences

CS301 – Algorithms

Homework 4

Due: November 26, 2020 @ 23.59
(Upload to SUCourse+ - **see late submission policy below**)

PLEASE NOTE:

- Provide only the requested information and nothing more. Unreadable, unintelligible and irrelevant answers will not be considered.
- You can collaborate with your `TA/INSTRUCTOR ONLY` and discuss the solutions of the problems. However you have to write down the solutions on your own.
- Plagiarism will not be tolerated.
- Submit your answers as a single PDF file
- Late submission is allowed for only 10 hours. Each hour late submission costs 10% of your grade. For example, if you submit 2 hours late, your grade will be multiplied by 0.8.

1. In this question we will solve Maximum Subsequence Sum (MSS) problem using dynamic programming. Given a sequence of integers, MSS problem is to calculate the sum of the subsequence of **non-adjacent** elements with the maximum sum. For example, if the sequence is $[-2, 1, 3, -4, 5]$, the answer would be 8 for the subsequence $[3, 5]$. Note that the empty subsequence is a subsequence of any sequence. We consider the sum of elements in an empty subsequence as 0. MSS problem for $[-1, -2, -3]$ has the answer 0, since the empty subsequence has the maximum sum among any subsequence we can consider for $[-1, -2, -3]$.

   (a) As you have seen in the lecture, we can solve LCS problem with dynamic programming approach, by giving a recurrence that formulates the construction of the optimal solution from the optimal solutions of subproblems. For our MSS problem, a similar recurrence can be given to solve the problem with dynamic programming.

   Let us introduce the function $MaximumSubseqSum(i)$ which has the following meaning. $MaximumSubseqSum(i)$ gives the maximum sum that we can obtain if we consider the indices from 0 to i. For example, when the sequence $[-2, 1, 3, -4, 5]$ is considered, $MaximumSubseqSum(0) = 0$, $MaximumSubseqSum(1) = 1$, $MaximumSubseqSum(2) = 3$, $MaximumSubseqSum(3) = 3$, and $MaximumSubseqSum(4) = 8$. Note that, if the last index in our sequence is k (we use 0 based indexing), then $MaximumSubseqSum(k)$ is the answer of the problem.

   Please fill–in the following recurrence with the appropriate values so that this recurrence gives the optimal solution for MSS problem for a sequence $A = [a_0, a_1, a_2, \ldots, a_n]$.

$$MaximumSubseqSum(i) = \begin{cases} \_\_\_\_\_\_ & \text{if } i < 0 \\ \_\_\_\_\_\_ & \text{if } i = 0 \text{ and } a_i > 0 \\ \_\_\_\_\_\_ & \text{if } i = 0 \text{ and } a_i \leq 0 \\ max(\_\_\_\_\_\_, \_\_\_\_\_\_) & \text{otherwise} \end{cases} \quad (1)$$

(b) By creating a dynamic programming table for the sequence $[1, -3, 4, 5, 2]$, calculate the sum of the subset of non-adjacent elements with the maximum sum. Show the final version of the table. (Hint: In this problem, table will be 1 dimensional)

2. Consider the following problem which is a slight modification of one of the problems of Homework 3. This time, it is allowed to carry partial amount of the items.

   Suppose that there are n items to be transferred from a warehouse A to another warehouse B. Each item i to be transferred has the weight of w_i and the value of v_i , $1 \leq i \leq n$. We will be paid 0.1% of the total value of the load that we transfer by our truck which can carry at most W tons, and we are allowed to make only one trip from warehouse A to warehouse B.

   (a) Suggest an efficient greedy algorithm to decide which items we should carry so that we make the most money.

   (b) What is the complexity of your algorithm?

   (c) Apply your algorithm to the following items and their value. Find best solution for transporting the items. (Assume W is 5 tons)

Item	Weight	Value
1	2	12
2	1	10
3	3	20
4	2	15