HW 7

Due on **Wednesday, October 19, 2022 at 9am CST**

CS/ECE 374A: Intro. Algorithms & Models of Computation, Fall 2022 Version: **1.1**

Submission instructions as in previous homeworks.

13 (100 PTS.) Break into intervals.

The input is a sequence $\alpha_1, \ldots, \alpha_n$ of n real numbers, and a parameter $k \leq n$. For integers $i \leq j$, their *interval* is the set $[\![i : j]\!] = \{i, i+1, \ldots, j\}$. The *benefit* of such an interval is $\varphi(i,j) = \sum_{t=i}^{j} (\alpha_t + \alpha_i)^2$. Describe an efficient *dynamic programming* algorithm that computes the maximum price solution when one breaks the sequence into (*exactly*) k intervals $I_1, \ldots, I_k$. Specifically, compute the intervals $I_1, \ldots, I_k$, such that:

(a) $\cup_i I_i = [\![1 : n]\!]$,
(b) all the k intervals are disjoint,
(c) all the intervals $I_1, \ldots, I_k$ are non-empty,
(d) and $\sum_i \varphi(I_i)$ is ***maximized***.

In addition, your algorithm also need to output the optimal solution itself (i.e., the k intervals forming the optimal solution).

As usual, your algorithm should be as fast as possible, and try and minimize (within reason) the space used.

(See top of previous homework to see exact instructions of what you have to provide – in particular, you have to provide a recursive formulation of this problem, and a dynamic [non-recursive] program to solve this problem.)

14 (100 PTS.) Convert into graph problems.

14.A. (50 PTS.) For problem 11 (from the previous homework), describe an efficient reduction of the problem to a graph problem. Formally, describe an efficient algorithm that solves it by constructing an appropriate DAG (i.e., directed acyclic graph) G, and then solving the original problem by solving a standard graph problem on G. Assuming that this standard problem can be solved in linear time in the size of the graph (i.e., number of vertices plus the number of edges), what is the running time of your algorithm? (As usual your algorithm should be as fast as possible, but we do not care about the space requirement here.)

14.B. (50 PTS.) Same as previous part, but for problem 13 above (here, you just need to compute the optimal solution value, there is no need to output the solution itself).