**Assignment 4 is due May 14, 17:30.**

**Problem 1 (100 points)** Consider a variation of the Longest Common Subsequence (LCS) problem where $k$ ($k > 2$) sequences are given as input and the goal is to find their LCS.

(a) How would you modify the recursive formula and the top-down LCS algorithm that we have studied in class for two sequences, to find an LCS for $k$ sequences? Please explain.

(b) What is the asymptotic time complexity of your algorithm? Please discuss: is it expected or a surprise?

A pdf copy of your own solution should be submitted at SUCourse.

**BONUS (100 points)** Suppose that a travel agent asks you to design and develop an algorithm that computes a shortest itinerary $I_c$ from *Istanbul* to every city/town $c$ in a given set $C$ of cities/towns in Turkey. For instance, if $C = \{Ankara, Izmir\}$, then the goal is to compute a shortest itinerary from *Istanbul* to *Ankara*, and a shortest itinerary from *Istanbul* to *Izmir*.

The travel agent has an additional request. For every city/town $c$ in $C$, there is a set $S_c$ of cities/towns that the tourists are usually interested in visiting on their way to $c$. The travel agent desires all the cities/towns in $S_c$ to be included exactly once in the itinerary $I_c$ computed by your algorithm. For instance, for $c$=*Ankara* and $S_{Ankara}=\emptyset$, according to the travel distances between cities/towns provided to you by the travel agent, a shortest itinerary from *Istanbul* to *Ankara* may be

$$I_{Ankara}=\langle Istanbul, Izmit, Bolu, Ankara\rangle.$$

If $S_{Ankara}=\{Bolu, Duzce, Zonguldak\}$ then,

$$I_{Ankara}=\langle Istanbul, Izmit, Duzce, Zonguldak, Bolu, Ankara\rangle.$$

If $S_{Ankara}=\{Kirikkale\}$ then

$$I_{Ankara}=\langle Istanbul, Izmit, Bolu, Kirikkale, Ankara\rangle.$$

(a) Define the problem described above, precisely as a computational problem: Input? Output?

(b) Prove that this problem is NP-complete: Membership? Hardness?

A pdf copy of your own solution should be submitted at SUCourse.