CMPT-405/705

January 20, 2016

ASSIGNMENT 2

Problem 1

You are at a conference and there are n presentations and you would like to attend quite a few of the presentations. Every presentation has a start time and a fnishing time, $s_1, \ldots, s_n$ and $f_1, \ldots, f_n$ respectively, with $s_i < f_i$ for all i. Unfortunately, the times at which some of the presentations take place overlap, so you can not attend all of them. We will add values $v_1, \ldots, v_n$ to the n presentations, corresponding to your desire to attend them.

You would like to figure out which presentations to attend such that the sum of their values is maximized. In $O(n \log n)$ time, use a dynamic programming algorithm to find a set of presentations with maximal total value such that none of their times overlap

Problem 2

The edit distance of two strings s and t is the the minimum number of single character operations (insert, delete, or substitution) needed to convert s into t. Let m and n be the length of strings s an t.

Design an $O(nm)$ time and $O(nm)$ space algorithm to calculate the edit distance between s and t.

Problem 3

The LONGEST COMMON SUBSTRING PROBLEM (LCS) is defined a follows. We are given two series $X = x_1, \ldots, x_m$ and $Y = y_1, \ldots, y_n$. The longest common substring $Z = z_1, \ldots, z_l$ is the maximum string that is a substring of X and Y. In more detail, l is the maximum value such that there exist $i_1 < i_2 < \cdots < i_l$ and $j_1, \ldots, j_l$ with $z_1 = x_{i_1} = y_{j_1}, \ldots, z_l = x_{i_l} = y_{j_l}$. Develop an algorithm following the approach of dynamic programming.

Problem 4

"Solve" the Knapsack problem using dynamic programming, assuming that all items have integer weights and values. The algorithm should have a runtime of $O(nW)$ where n is the number of items and W is the size of the knapsack.