Purdue University
CS 381

Hambrusch
Spring 2015

Assignment 4

Due: Tuesday, March 3, 2015 (hand in before class)

1) (15 pts.) Consider the following variation of the sequence alignment problem presented in class. Given are two strings $X = x_1x_2 \dots x_n$ and $Y = y_1y_2 \dots y_m$. In the *asymmetric gap alignment* problem the cost of aligning X and Y is defined by the following operations:

- If a character in X is not matched with a symbol in the Y, it occurs an out-gap penalty cost of 3.
- If a character in Y is not matched with a symbol in the X, it occurs an in-gap penalty cost of 1.
- If x_i is matched with y_j and $x_i = y_j$, there is no cost. If $x_i \neq y_j$, the mismatch cost is 2.

The cost of an alignment is the sum of all gap penalties and mismatch costs. For example, for $X = abba$ and $Y = ababab$ an alignment cost of 2 can be achieved.

1. State and explain the recurrence relation for $OPT(i, j)$ which represents the minimum cost of aligning $x_1x_2 \dots x_i$ and $y_1y_2 \dots y_j$ using the above rules.
2. State and explain time and space bounds for computing the entry $OPT(n, m)$ (i.e., only the best alignment cost) using the recurrence relation.
3. Describe how to generate the actual alignment by producing the pairs of characters matched. Characters not matched do not have to be listed. What additional data structures need to be maintained? What is the time needed to generate the alignment?

2 (15 pts.) Bonnie and Clyde are shipping some of their valuable possessions. Each one has one duffel bag; Bonnie's has a capacity of W_B and Clyde's has a capacity of W_C. Altogether they have n valuable possessions and each possession has a weight of w_i and a value of v_i, $1 \leq i \leq n$. Bonnie identifies the possessions she is willing to pack in her duffel bag and Clyde does the same. Hence, every possession has a type t_i which is either B, C, or BC.

Describe and analyze an efficient DP algorithm deciding the items Bonnie and Clyde should pack so that (i) items packed in each duffel bag do not exceed its capacity, (ii) each one only packs possessions they are willing to pack, and (iii) the sum of the values of the packed possessions is maximized. Clearly state and explain the recurrence relation underlying the DP algorithm.

3) (20 pts.) Transylvania has built a new interstate connecting the towns of Brasov and Cluj-Napoca. The interstate has n exit/entry ramps. The goal is to build g gas stations at the ramps. The authorities want to have at most d exits between two consecutive ramps having a gas station. The cost of operating a gas station at ramp i is c_i, $1 \leq i \leq n$.

1. State and explain the conditions under which there exists a solution placing g gas stations so that there are most d exits between two consecutive ramps having a gas station. For example, for $n = 11$, $d = 2$, and $g = 2$, there exists no solution. For $g = 3$ there exists a unique solution and for $g = 4$ there exist multiple solutions.

2. Describe and analyze an efficient DP algorithm to determine where to locate the g gas stations so that (i) the total operating cost is a minimum and (ii) a traveler encounters at most d exits without a gas station before an exit has one. Follow the four steps of a DP algorithm described in class. The algorithm takes as input an array C of size n representing the costs of operating gas stations and two integers, g and d defined above.

   Hint: One approach is to compute entries $OPT(i, r, q)$ representing the optimum solution for ramps 1 to i having placed r gas stations and the closest gas station is distance q from ramp i ($q = 0$ means ramp i has a gas station).