(CS1.301) Algorithm Analysis & Design (Monsoon 2024)

Practice Problem Set 4

Instructions:

- The following problem set is not graded and is for practice.
- Some of the problems will be covered in the tutorial.

Question 1

Given two sequences A, B, count number of unique ways in sequence A, to form a subsequence that is identical to the sequence B.

Subsequence : A subsequence of a string is a new string which is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (ie, “ACE” is a subsequence of “ABCDE” while “AEC” is not).

Take a look at the following example:

A = "rabbbit"

B = "rabbit"

Output: 3

Explanation: These are the possible removals of characters:

=> A = "ra_bbit"

=> A = "rab_bit"

=> A = "rabb_it"

Note: "_" marks the removed character.

Devise an algorithm which takes two strings A and B and returns the number of distinct subsequences of B in A.

Question 2

Edit distance between two strings is the minimum number of letter insertions, letter deletions, and letter substitutions required to transform one string into the other. For example, the edit distance between FOOD and MONEY is at most four, and distance between the strings ALGORITHM and ALTRUISTIC is at most 6.

For any two input strings $A[1..m]$ and $B[1..n]$, we can formulate the edit distance problem recursively as follows: For any indices i and j, let $\mathtt{Edit}(i, j)$ denote the edit distance between the prefixes $A[1..i]$ and $B[1..j]$. We need to compute $\mathtt{Edit}(m, n)$. Please complete the arguments here on how $\mathtt{Edit}(i, j)$ can be computed using memoization.

Question 3

Given two integer arrays $array_1$ and $array_2$, write the integers of $array_1$ and $array_2$ (in the order they are given) on two separate horizontal lines. A straight line connecting two numbers $array_1[i]$ and $array_2[j]$ is called a connecting line if:

(a) $array_1[i] == array_2[j]$

(b) The line drawn does not intersect any other connecting (non-horizontal) line.

Note that a connecting line cannot intersect even at the endpoints (i.e., each number can only belong to one connecting line).

Give an algorithm that takes two integer arrays, $array_1$ and $array_2$ and returns the maximum number of connecting lines that can be drawn between the two arrays.

Question 4

There is an $m \times n$ grid. A robot is initially located at the top-left corner (i.e., $grid[0][0]$). The robot tries to move to the bottom-right corner (i.e., $grid[m-1][n-1]$). An obstacle in the path of the robot is marked by a '1', i.e, if $grid[i][j]$ is 1, the robot cannot move through that space. The robot can move through grid cells marked with a 0. The robot can only move either down or right at any point in time.

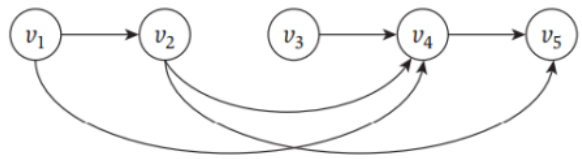
Given two integers m and n, write an algorithm that returns the number of possible unique paths that the robot can take to reach the bottom-right corner.

Question 5

Let G = (V, E) be a directed graph with nodes $v_1, \ldots, v_n$. We say that G is an ordered graph if it has the following properties.

(i) Each edge goes from a node with a lower index to a node with a higher index. That is, every directed edge has the form (v_i, v_j) with $i < j$.

(ii) Each node except v_n has at least one edge leaving it. That is, for every node $V_i(i = 1, 2, \ldots, n-1)$, there is at least one edge of the form (v_i, v_j). The length of a path is the number of edges in it.



Give an efficient algorithm that takes an ordered graph G and returns the length of the longest path that begins at v_1 and ends at v_n. For example int the graph above, $(v_1 \to v_2 \to v_4 \to v_5)$ is the longest path of length 3.