Longest Common subsequence (LCS)

The problem

Given two strings, X and Y, the *longest common subsequence* (LCS) problem is the problem of finding the longest sequence of matching letters in X and Y. Matching letters may be separated by 0 or more non-matching letters. The following recursive function computes the length of the LCS for the first *i* letters of X and *j* letters of Y.

$$lengthLCS[i, j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ lengthLCS[i-1, j-1] + 1 & \text{if } i > 0, j > 0 \text{ and } x_i = y_j \\ \max(lengthLCS[i, j-1], lengthLCS[i-1, j]) & \text{if } i > 0, j > 0 \text{ and } x_i \neq y_j \end{cases}$$

In the equation, the notation

x_i means the *ith* letter of X and
y_j means the *jth* letter of Y.

Example

Given the two strings:

X = < a, b, c, d > and
Y = < x, a, y, b, d>

The length of the LCS is *lengthLCS*[4,5] = 3.