CSEN703: Analysis and Design of Algorithms

Assignment 2

Author: Maram Osama
52-4968
8001779

To solve this problem, we use dynamic programming, where we use tabulation to store the highest-scoring ‘paths’ so we can later construct our strings accordingly.
First, the tabulation step: if our first string is of length ‘n’, and second string is of length ‘m’, then we create a m+1 x n+1 matrix (called max_score) initially of zeroes. Then we populate it by the max values of aligning the strings. For the first row, we can’t compare our alignment with previous row(s), so we populate it with scores of aligning our first string with a string of dashes ‘-’. For the first column, we can’t compare our alignment with previous column(s), so we populate it with scores of aligning our second string with a string of dashes ‘-’. Thus we’ve done the base case.
Then, we do the rest of the matrix. How we choose the score is simple, assuming we have a perfect alignment until max_score[i-1][j-1], we can do one of three things: align two characters of string (thus we move diagonally), align a character from first string with a gap, align a character from second string with a gap. We use the scoring matrix to calculate score of each possibility, and we take the maximum one and put it as the new value for max_score[i][j]. This step has time complexity (m x n) as we loop over each row (m) and within each row we loop over the columns (n).
Then we need to construct the string that gives us this maximum score. We compare the value in max_score[i][j] and see where this value has come from (did it come from aligning two characters, or a character with a gap, or a gap with a character) and then write the string accordingly. At the end, if one string still has characters, we check for it and write those characters and align them with gaps.
The alignment for the first example for s1= ‘ATGCC’ and s2= ‘TACGCA’ gave us this alignment: -ATGCC-

TACG-CA Which is different from the one in example in assignment, but it has a better score which is 0 (zero), in comparison to the example which has a score of -2.1. Same for second example, my solution has a score of +5, but example= -7.4