This code uses time complexity of O(nm) the length of 2 input strings

It starts with looping on both strings comparing each character and comparing its combination with either the corresponding character or with the ‘-‘ or with choosing the other character with the ‘-‘and choose the highest score between both

Then we input the chosen value in an empty constructed matrix for dynamic programming then we backtrack to get out the values in the output strings by looping on the dp Matrix

If both i and j are greater than 0 (meaning there are characters left in both sequences), and the score for the current position in the matrix (dp[i][j]) is equal to the score obtained by aligning the characters charX and charY then insert both

If only i is greater than 0 (meaning there are characters left in sequence x), and the score for the current position in the matrix (dp[i][j]) is equal to the score obtained by aligning charY with a gap in sequence x , then insert ‘-‘ in x

If only j is greater than 0 (meaning there are characters left in sequence y), and the score for the current position in the matrix (dp[i][j]) is equal to the score obtained by aligning charX with a gap in sequence y , then insert ‘-‘ in y