Analysis Assignment 2 Report

By: Mohamed Ihab Seifedlin Zakaria 7000399 T2

First the initialization seqA, seqB the input sequence of strings normally then the delta(scoring matrix) is allowed to be flexible 2D array that is defined as whatever the user wants, it is independent of the actual code.

```
delta = {
   'A': {'A': 1,   'G': -0.8, 'T': -0.2, 'C': -2.3, '-': -0.6},
   'G': {'A': -0.8, 'G': 1,   'T': -1.1, 'C': -0.7, '-': -1.5},
   'T': {'A': -0.2, 'G': -1.1, 'T': 1,   'C': -0.5, '-': -0.9},
   'C': {'A': -2.3, 'G': -0.7, 'T': -0.5, 'C': 1,   '-': -1},
   '-': {'A': -0.6, 'G': -1.5, 'T': -0.9, 'C': -1,  '-': 0}
      }
```

So, the scoring matrix can be changed to incorporate different/new types of genes as long as the relationship is written in a valid way and to fetch any relation we would use delta[‘geneA’][‘geneB’].

sequenceAlignment function:

m,n get the length of input string sequences.

resultMatrix is initialized as a 2D array with size (m+1)*(n+1) with all of its cells initialized as zeros while statusMatrix is the same size except its cells are initialized as empty strings.

Dynamic nested for loop where we get the maximum value of the 3 possible combinations at that moment where we get the current x and y in the delta(match) or current x and - (gapY) or - and current y(gapX), the value is calculated using the delta(scoringMatrix) + the corresponding resultMatrix using the indices of the match/gapX/gapY. This MAX value is stored in the corresponding resultMatrix and the status is stored in the corresponding statusMatrix.

Afterwards we know the status and our next step is to get the aligned sequence Strings, we get them using concatenation in the newly initialized strings alignedX, alignedY depending on the status of the current index in the while loop and we only concatenate x and y at the same time as to not have one string be longer and not match the other string.

Now assuming that is all said and done, our next and final step is to calculate the alignmentScore to know how close the sequences are to alignment.\

Note: I am not using the resultMatrix to calculate the alignmentScore.

Using a simple for loop we match the current character in alignedX, alignedY,

```
alignmentScore = 0
for z in range(len(alignedX)):
     alignmentScore += delta[alignedX[z]][alignedY[z]]
```

Now that matches using delta[‘geneA’][‘geneB’] as I’ve said at the beginning of the initialization. Finally, we “return alignedX, alignedY, alignmentScore” as a tuple to be accessed after the function is executed.

```
output = sequenceAlignment(seqA, seqB, delta)
print("Sequence X:", output[0])              print("Sequence Y:", output[1])              print("Alignment Score:", output[2])
```

We access the results from the output as a tuple/array and display them.

Final Comment: My code has a lot of comments that could help explaining if something is unclear in the report(about both the syntax and the logic of the ideas I implemented)