Bryce Egley
Brian Huang
CS325

Implementation 2 report

Pseudo code

Read in the text
Pass the two sequences, and the cost table into the findMin function.
Run the find distance function with the two string, their lengths and the cost table.
Create a matrix with the dimensions of [length1 + 1][length2 + 1]
For i in length1 + 1
For j in length2 + 1
Fill in the top row of the matrix using the cost table.
Fill the first column of the matrix using the cost table
If the two string at [i-1][j-1] are equal
Matrix at [i][j] = matrix [i-1][j-1] + cost
Else
matrix [i][j] = cost + minimum(insertion, remove, substitution)

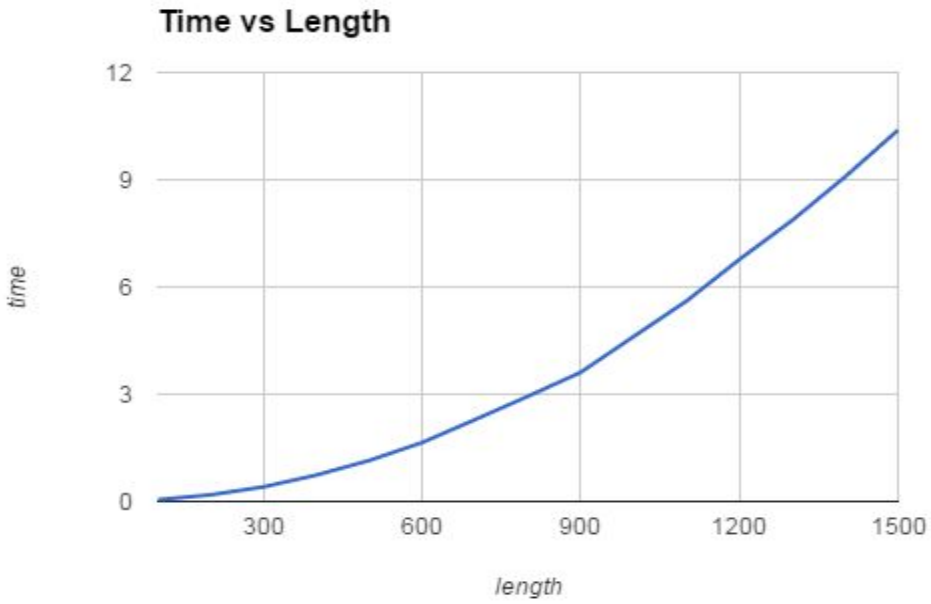
Return the value at matrix[i][j] as the cost.

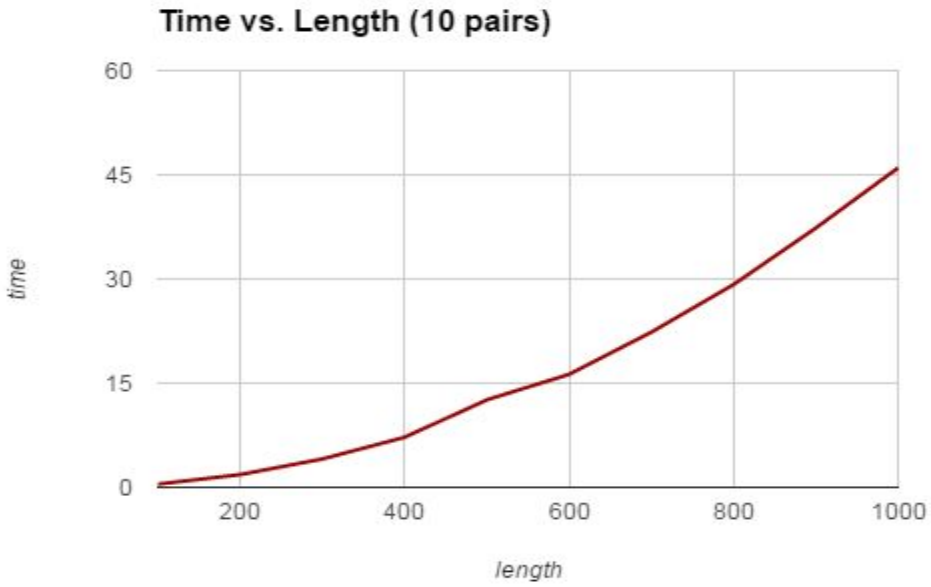
Asymptotic Analysis of run time

Our code uses two for loops to find the shortest distance, and then one for to run through the pairs of sequences. The number of sequences can be negligible depending on how many there are. Usually the dna sequences are much longer than the number of sequences so the length of the sequence is what really determines the runtime. Since we used two for loops the expected run time can be $O(n^2)$

Reporting and plotting the runtime

To measure the run time for our program we used a generator file that we created. The generator file creates a pair of sequences for our program to compare. The graphs below show the runtime of our program with varying length and size. We tested two different number of pairs and we varied the length of the dna sequences to test the runtime. The graphs match our results, they both look like they have a polynomial runtime.





Interpretation and Discussion

The growth curve does match our expectation based on their theoretical bounds. The run time we calculated from our algorithm was $O(n^2)$ and the graph follows that as shown above. When we run 10 pairs the graph still appears to follow an $O(n^2)$ curve.

The solution our code outputs is consistently either off by a small margin or gets the correct answer. We believe there is an error somewhere in our logic but we were not able to figure out what is causing it. I suspect it is the way we gave the findDistance method the strings and it might not be looking at the first or last character.