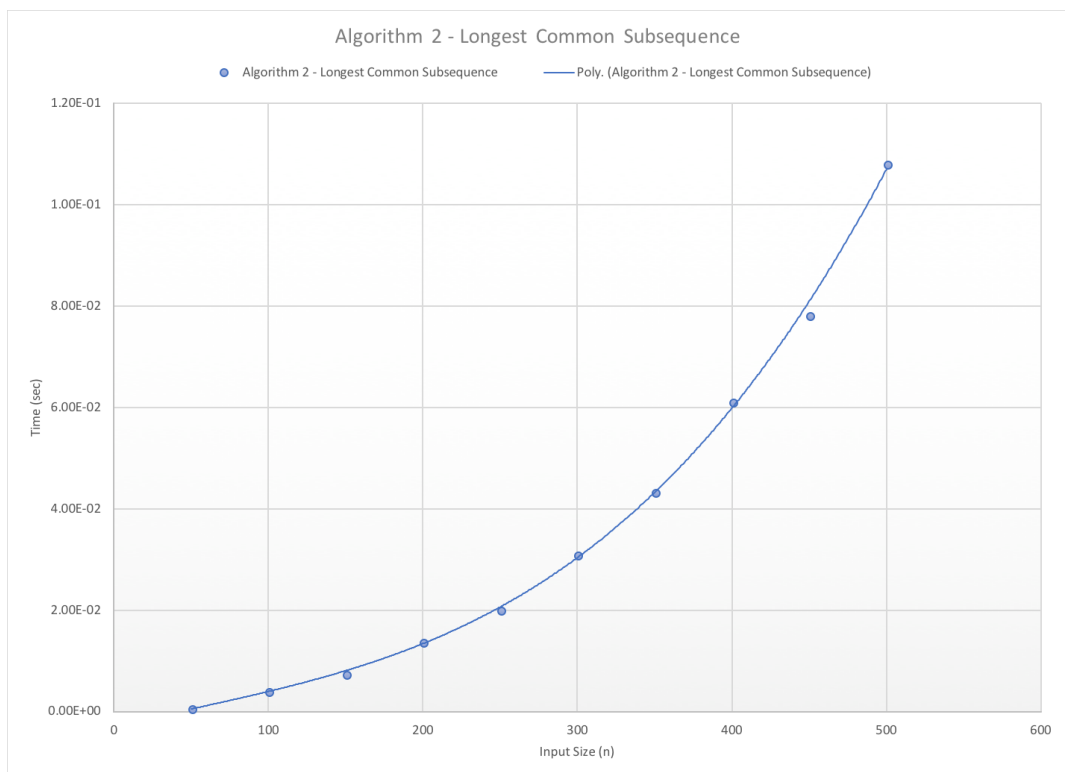
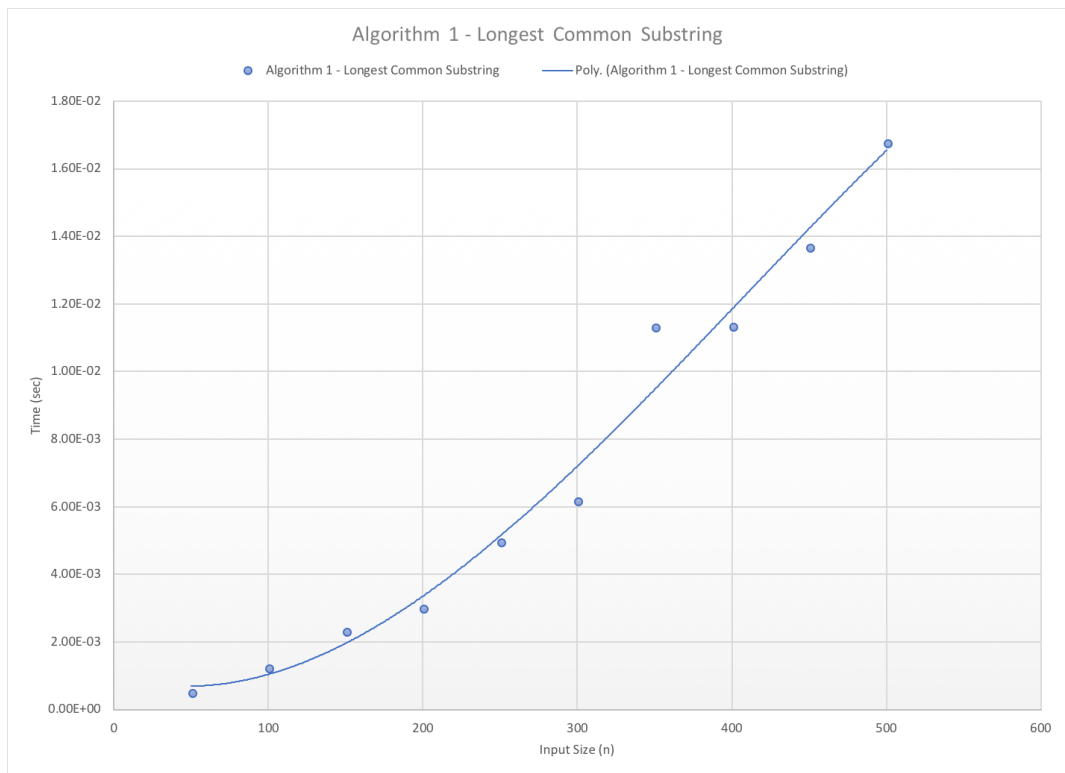


CPSC 335

Project 4: Dynamic Programming

Devontae Reid devontae.reid@csu.fullerton.edu

Scatterplots



Questions

- a. Consider the two algorithms you've implemented for the substring problem (the exhaustive search algorithm in project 3, and the dynamic programming algorithm in this project). How do their performance compare; is one significantly faster than the other, and which one; or are they roughly equivalent? If there is a speed difference, is it enough to make the difference between the algorithm being practical to use, and impractical?

Between the two algorithms, I found out that dynamic programming accomplished the same solution at exhaustive search but did it extremely faster. The speed difference is a huge difference because in the same time it takes the exhaustive search to find a solution for two strings of length less than 20. The dynamic program can do a string of length 500.

- b. Answer the same questions, but for the subsequence problem.

The second algorithm was similar to the first when I compared it to its counterpart from project 3. The exhaustive program would take about an hour to compare two words of length 500 while it doesn't even take a second for dynamic program to find the longest common subsequence.

- c. Did you find implementing this algorithm to be easy, difficult, or in between? What was the hardest part, and why? How does the difficulty level compare to the exhaustive search algorithms from project 3?

Yes, I found out that the dynamic programming algorithm was tremendously easier to implement. There was not anything hard about the two algorithms can finish them in less than 30 mins. The exhaustive search took about 4 hours to implement because majority of the time was spent researching.