# Homework 6

CS 4364/5364
Spring 2022

Due: ~~6~~ 13 April 2022

Because of the reliance of the particular assignments in this class on mathematical notation, and the fact that all assignments will be submitted electronically, students are encouraged to use LaTeX to formalize their responses. **For those enrolled in the graduate section the use of latex is *required.*** This assignment (like all others) will be posted on the course `github`[1] as source code as well as in PDF form on the course website. Graduate students will need to include the `.tex` files as well as a PDF, this is optional but encouraged for undergraduates.

**Question (~~25~~ 35 points):** Give an algorithm, a proof of correctness, and proof of running time to solve the following problem in $O(kn^2c)$-time.

Given integers $k \geq 1$ and $c \geq 1$, as well as two strings $S$ and $T$ both of length $n \gg k$, determine a similarity between them that is calculated as the sum of:

- $c+1$ times the maximum number of shared distinct exact $k$-mers
- $c$ times the maximum number of shared distinct $k$-mers with one change that have not been accounted for already
- $c-1$ times the maximum number of shared distinct $k$-mers with two changes that have not been accounted for already
- ...
- 1 times the maximum number of shared distinct $k$-mers with $c$ changes that have not been accounted for already

normalized by the total number of distinct $k$-mers in both sequences. Here changes are replacements only and not indels.

As an example for the values of $k = 3, c = 2$ and the strings `AACTGT` and `TGTAAA` the similarity is $\frac{3}{4}$. The distinct 3-mers from the two strings are `AAC,ACT,CTG,TGT` and `TGT,GTA,TAA,AAA`. There is one 3-mer that matches exactly (`TGT`). There is one pair of 3-mers that match with one change (`AAC` and `AAA`). There is one pair of 3-mers that match with two changes (`CTG` and `GTA`) The remaining pair would need 3 changes (i.e. more than $c$ changes). Thus the similarity is

$$\frac{(1 \times 3) + (1 \times 2) + (1 \times 1)}{8} = \frac{3}{4}.$$

[1] `github.com/deblasiolab/CS4364-documents`

Hint: to find the maximum number of matching $k$-mers with a fixed number of changes, you may want to look into Hopcroft-Karp which finds a maximal-cardinality matching in bipartite graphs[2]

[2] `https://en.wikipedia.org/wiki/Hopcroft-Karp_algorithm`