Homework 5

CS 4390/5390
Fall 2019

Due: 2 December 2019
(1 point of extra credit if you turn it in on the original due date, 27 November)

This homework is worth 4 points out of the total 25 points of homework in the class.

1. **(2 Point)** Given a **BTW index** $(L[1...n], C[1...\sigma])$ of string $S[1...n]$, design an algorithm to find the length of the longest repeat (longest substring that occurs at least twice). Hint this will be recursive and will run in $O(\sigma n)$ time.

2. **(2 point)** Given a set of reads $R = (r_1, r_2, r_3, ..., r_\ell)$, a k-mer conversion function $f(x) = y$ such that assigns an integer $y \in [1...\sigma^k]$ to each $x \in \Sigma^k$, and a k-mer count array $C[0...\sigma^k]$ where $C[y]$ contains the number of times $f^{-1}(y)$ occurs in R. An erroneous k-mer is one that occurs at least once and less than 5 times. Design an algorithm that outputs a modified set of reads $R' = (r'_1, r'_2, r'_3, ..., r'_\ell)$ that replaces any errors such that the number of k-mers in R' that are erroneous is lowered (it may not be eliminated). You can assume that any window of $2k$ bases will only have 1 error, i.e. there will never be conflicts where two point mutations in the same k-mer. In the case that a character could be replaced with two different characters and satisfy this condition, prefer the one that has more total occurrences across the k overlapping windows.

   For example, assume $k = 3$ and the following read segment corrections would be made given these k-mer frequencies:

   `...ACTTG...` $\longrightarrow$ `...ACCTG...`

x	$C[f(x)]$
ACA	100
ACC	50
ACT	3
ACG	9
ATG	2
CAT	4
CCT	12
CTT	4
CTG	7
CGT	0
TTG	5
GTG	3