Name: _____

COT 6417
Algorithms on Strings and Sequences
Fall 2020
Mid-term exam

There are five questions.
Note: You can assume that the alphabet Σ is of constant size.

Question 1 (20 points)
Strings A and B are said to be cyclic shifts of each other if $A = \alpha\beta$ and $B = \beta\alpha$, for non-empty strings α and β. Provide an algorithm that determines whether two strings S_1 and S_2, of lengths m and n respectively, are cyclic shifts of each other. The algorithm should run in $O(n+m)$ time.

Question 2 (20 points)

Given a set $S = \{s_1, s_2, .., s_Z\}$ of Z distinct strings, the problem is to identify each string in S that is a substring of some other string in S. Provide an algorithm that runs in $O(n)$ time, where n is the sum of the lengths of all strings in S.

Question 3 (20 points)

Given text T (of length m) and a number k, the problem is to determine whether there exists a string α of length k such that $\alpha\alpha$ is a substring of T. Provide an algorithm that solves this problem in $O(m)$ time.

Question 4 (20 points)

Given a string S (of length m) and a number k, the problem is to find, if it exists, the *longest* substring of S that occurs *exactly* k times in S. Provide an algorithm that solves this problem in $O(m)$ time.

Question 5 (20 points)

Let S be a string of length n whose characters are drawn from an alphabet Σ (of constant size). We are interested in queries of the form $Q(i,j,c)$, where $Q(i,j,c)$ returns the number of times character $c \in \Sigma$ occurs in the substring $S[i,j]$. Provide an algorithm that takes $O(1)$ time to answer a query $Q(i,j,c)$, and which uses no more than $O(n)$ bits to store any auxiliary information.