Programming Assignment

Q1) Given two strings s and t, t is a substring of s if t is contained as a contiguous collection of symbols in s (as a result, t must be no longer than s).

The position of a symbol in a string is the total number of symbols found to its left, including itself (e.g., the positions of all occurrences of 'U' in "AUGCUUCAGAAAGGUCUUACG" are 2, 5, 6, 15, 17, and 18). The symbol at position i of s is denoted by $s[i]$

A substring of s can be represented as $s[j:k]$, where j and k represent the starting and ending positions of the substring in s; for example, if s = "AUGCUUCAGAAAGGUCUUACG", then $s[2:5]$ = "UGCU".

The location of a substring $s[j:k]$ is its beginning position j; note that t will have multiple locations in s if it occurs more than once as a substring of s (see the Sample below).

Given: Two DNA strings s and t (each of length at most 1 kbp).

Return: All locations of t as a substring of s.

Q2) Assume that an alphabet A has a predetermined order; that is, we write the alphabet as a permutation A=(a1,a2,…,ak), where a1<a2<…<ak. For instance, the English alphabet is organized as (A,B,…,Z)

Given two strings s and t having the same length n, we say that s precedes t in the lexicographic order (and write s<Lext) if the first symbol s[j] that doesn't match t[j] satisfies sj<tj in A.

Given: A collection of at most 10 symbols defining an ordered alphabet, and a positive integer n (n≤10).

Return: All strings of length n that can be formed from the alphabet, ordered lexicographically (use the standard order of symbols in the English alphabet).