CSIT 5500 Advanced Algorithms
2020 Spring Semester
Written Assignment 2
Handed out: March 18
Due: 21:00 on April 1
Please submit a soft copy via the canvas system by the due date and time shown above. Late assignments will not be graded.

1. (10 points) (Problem R-11.7 in "Data Structures and Algorithms in C++" by Goodrich, Tamasia, and Mount, Wiley & Sons)

   Compute the next($\cdot$) table for the pattern

   cgtacgttcgtac

   that supports the KMP string matching algorithm. Show your steps.

2. (10 points) Construct the suffix array for the string

   minimize

   by following the construction algorithm in the lecture notes which consists of $O(\log n)$ stages. For each stage $j = 0, 1, 2, \ldots$, show the following:

   - The ordered pairs of substrings to be processed, as well as their corresponding ordered pairs of ranks.
   - The output table R for this stage j: $R[i]$ gives the rank of the substring that starts at position i and has length 2^j.

3. (10 points) (Problem 6.4 in "Introduction to Algorithms" by Cormen, Leiseron and Stein, MIT Press) You are given a string of n characters $s[1..n]$, which you believe to be a corrupted text document in which all punctuation marks and blanks have vanished (so that it looks something "itwasthebestoftimes..."). You wish to reconstruct the document using a dictionary, which is available in the form of a boolean function dict($\cdot$): for any string w,

$$\text{dict}(w) = \begin{cases} \text{true} & \text{if } w \text{ is a valid word} \\ \text{false} & \text{otherwise} \end{cases}$$

   Give a dynamic programming algorithm that determines whether the string $s[1..n]$ can be reconstituted as a sequence of valid words. The running time should be at most $O(n^2)$, assuming calls to dict take unit time. Define and explain your recurrence relation and boundary conditions.