# Homework Assignment #7

## Note

This assignment is due 2:10PM Tuesday, May 15, 2018. Please write or type your answers on A4 (or similar size) paper. Drop your homework by the due time in Yih-Kuen Tsay's mail box on the first floor of Management College Building 2. Late submission will be penalized by 20% for each working day overdue. You may discuss the problems with others, but copying answers is strictly forbidden.

## Problems

There are five problems in this assignment, each accounting for 20 points.

1. (6.47) Modify the KMP string matching algorithm to find the largest prefix of $B$ that matches a substring of $A$. In other words, you do not need to match all of $B$ inside $A$; instead, you want to find the largest match (but it has to start with $b_1$).

2. Consider the *next* table as in the KMP algorithm for string $B[1..9] = abaababaa$.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| $a$ | $b$ | $a$ | $a$ | $b$ | $a$ | $b$ | $a$ | $a$ |
| $-1$ | 0 | 0 | 1 | 1 | 2 | 3 | 2 | 3 |

   Suppose that, during an execution of the KMP algorithm, $B[6]$ (which is an $a$) is being compared with a letter in $A$, say $A[i]$, which is not an $a$ and so the matching fails. The algorithm will next try to compare $B[next[6]+1]$, i.e., $B[3]$ which is also an $a$, with $A[i]$. The matching is bound to fail for the same reason. This comparison could have been avoided, as we know from $B$ itself that $B[6]$ equals $B[3]$ and, if $B[6]$ does not match $A[i]$, then $B[3]$ certainly will not, either. $B[5]$, $B[8]$, and $B[9]$ all have the same problem, but $B[7]$ does not.

   Please adapt the computation of the *next* table so that such wasted comparisons can be avoided.

3. (6.17) Given two strings $A = ababa$ and $B = bbaaba$, compute the minimal cost matrix
   $C[0..5, 0..6]$ for changing the first string character by character to the second one. Aside from giving the cost matrix, please show the details of how the entry $C[5,6]$ is computed.

4. (6.40) Design an algorithm that, given a set of integers $S = \{x_1, x_2, \ldots, x_n\}$, finds a nonempty subset $R \subseteq S$, such that

$$\sum_{x_i \in R} x_i \equiv 0 \pmod{n}.$$

Before presenting your algorithm, please argue why such a nonempty subset must exist.

5. (6.62) You are asked to design a schedule for a round-robin tennis tournament. There are $n = 2^k$ ($k \geq 1$) players. Each player must play every other player, and each player must paly one match per round for $n-1$ rounds. Denote the players by $P_1, P_2, \ldots, P_n$. Output the schedule for each player. (Hint: use divide and conquer in the following way. First, divide the players into two equal groups and let them play within the groups for the first $\frac{n}{2} - 1$ rounds. Then, design the games between the groups for the other $\frac{n}{2}$ rounds.)