University of Houston

COSC 3320: Algorithms and Data Structures
Spring 2017

Homework 7

Due April 27, at the start of class

1. Given two strings X and Y, a third string Z is a *common superstring* of X and Y if X and Y are both subsequences of Z. (Example: if $X =$ `sos` and $Y =$ `soft`, then $Z =$ `sosft` is a common superstring of X and Y.) Design a dynamic programming algorithm which, given as input two strings X and Y, returns the length of the shortest common superstring (SCS) of X and Y. Specifically, you have to write a recurrence relation $\ell(i,j) = |SCS(X_i, Y_j)|$ that defines the length of a shortest common superstring of X_i and Y_j, and the pseudocode. The algorithm, which has to return $\ell(n,m)$, must run in time $\Theta(n \cdot m)$, where $n = |X|$ and $m = |Y|$. (Hint: use an approach similar to the one used to compute the length of a LCS of two strings.)

2. Let G be an undirected graph with n vertices and m edges. Argue that
   (a) If G is connected, then $m \geq n - 1$.
   (b) If G is a tree, then $m = n - 1$.

3. Consider the following simple graph, represented by its adjacency matrix.

	a	b	c	d	e	f	g
a	0	0	1	0	1	1	0
b	0	0	0	0	0	0	1
c	1	0	0	1	0	0	0
d	0	0	1	0	1	0	1
e	1	0	0	1	0	1	0
f	1	0	0	0	1	0	1
g	0	1	0	1	0	1	0

   (a) Draw the graph.
   (b) Run the DFS algorithm starting from vertex a, and draw the final DFS tree.
   (c) Run the BFS algorithm starting from vertex a, and draw the final BFS tree.

4. Let $G = (V, E)$ be a (possibly not connected) graph with n vertices and m edges. Design and analyze an algorithm that returns, if it exists, a vertex $v \in V$ such that at least $n/2$ different vertices are reachable, via a path, from v. (Hint: Use the BFS algorithm.)

5. Consider the following weighted graph, represented by its adjacency matrix, where x_i is

the i-th digit of your 7-digit UH ID.

	a	b	c	d	e	f	g
a	-	x_3	-	-	-	x_5	1
b	x_3	-	10	-	9	-	4
c	-	10	-	x_1	-	-	x_6
d	-	-	x_1	-	6	-	x_7
e	-	9	-	6	-	5	x_2
f	x_5	-	-	-	5	-	2
g	1	4	x_6	x_7	x_2	2	-

(a) List the edges of the minimum spanning tree in the order they are added by Kruskal's algorithm.

(b) List the edges of the minimum spanning tree in the order they are added by Prim's algorithm starting from vertex a.