1. Let $G = (V, E)$ be a directed graph.

   (a) Assuming that G is represented by an adjacency matrix $A[1..n, 1..n]$, give a $\Theta(n^2)$-time algorithm to compute the adjacency list representation of G. (Represent the addition of an element v to a list l using pseudo code by $l \leftarrow l \cup \{v\}$.)

   (b) Assuming that G is represented by an adjacency list $\mathrm{Adj}[1..n]$, give a $\Theta(n^2)$-time algorithm to compute the adjacency matrix of G.

2.

   (a) Use breadth-first-search to determine if an undirected graph $G = (V, E)$ is 2-colorable, and if so 2-color it. Write the pseudo code.

   (b) Use depth-first-search to determine if an undirected graph $G = (V, E)$ is 2-colorable, and if so 2-color it. Write the pseudocode.

3. A cycle can be represented by a list of numbers from the set $\{1, \ldots, n\}$. The last number in the list connects back to the first. The cycle is not necessarily simple, so a number may be on the cycle more than once. Given two such cycles that share at least one number, we would like to splice the two cycles together to form one large cycle. To keep things simple, in the following problems, you may assume auxilliary memory initialized however you like (independent of the input).

   (a) Assume that a cycle is represented by an array. Describe an efficient algorithm to splice two cycles, in arrays A and B, into one cycle, in array C. Give the pseudo code *and* briefly state in English how your algorithm works. (There may be more than one way to splice the two cycles together. Any legitimate splicing is fine.) Analyze its efficiency.

   (b) Assume that a cycle is represented by a circular linked list with a pointer `next` to the next vertex in the list. Describe an efficient algorithm to splice the cycles together into one cycle. The actual splicing should be constant time. The two cycles have pointers a and b into them, and the final cycle should have a pointer c into it. Give the pseudo code *and* briefly state in English how your algorithm works. Analyze its efficiency.