

Due at the start of class Wednesday, July 9, 2003.

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

Problem 2. Do Exercise B.5-5 on page 1091 of CLRS. (Exercise 5.5-6 on page 97 of CLR.)

Problem 3.

- (a) Give an if and only if condition for a *directed* graph to have an Euler cycle.
- (b) Give high-level pseudo code for an efficient algorithm (time $\Theta(|V| + |E|)$) to find an Eulerian cycle in a directed graph. Assume the graph is stored using an adjacency list. Make sure you explicitly show how to splice two intersecting cycles together.

Problem 4. Do Exercise 24.3-1 on page 600 of CLRS. (Exercise 25.2-1 on page 531 of CLR.)

Problem 5. Give a simple example of a directed graph with negative weight edges, but no negative weight cycles, for which Dijkstra's algorithm produces incorrect answers.