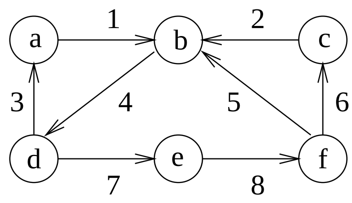
Analysis of Algorithms: Assignment 8

Due date: March 22 (Thursday)

Problem 1 (5 points)
Using Figures 23.1 and 23.2 (page 466) in the textbook as a model, give the adjacency-list and adjacency-matrix representation of the following weighted graph; note that your representation should include the edge weights. Assume that weights represent network capacities, and the absence of an edge between two vertices corresponds to a zero capacity.



Problem 2 (5 points)
Write efficient algorithms for converting (a) an adjacency-list representation of a graph into an adjacency matrix and (b) an adjacency matrix into adjacency lists. Give the time complexity of your algorithms.

Problem 3 (bonus)
This problem is optional; if you solve it, then you will get one bonus point toward your **final grade** *for the course. You cannot submit this bonus problem after the deadline.*

Consider a directed graph with n vertices, represented by an adjacency matrix $M[1..n, 1..n]$. A vertex is called a *sink* if it has $(n-1)$ incoming edges and no outgoing edges; note that the graph can have at most one sink. Give an algorithm that finds the sink vertex and returns its number; if there is no sink, return 0. The running time of your algorithm must be $O(n)$.