

CSCI 115 Lab 1
DUE: 2/1/15 (Sunday) at 11:59 P.M. on Mulan

For this lab assignment, you will be implementing a directed graph data structure in two ways: (1) adjacency list and (2) adjacency matrix.

You need to implement the following six functions for both implementations:

`initialize_graph`: this function creates a new directed graph

`insert_edge`: inserts a directed edge ($V1 \rightarrow V2$) into the graph

`delete_edge`: deletes an edge from the graph

`list_all_edges`: lists all of the edges in the graph

`list_all_neighbors`: lists all of the neighbors for a particular vertex*

`no_incoming_edges`: lists all of the vertices with no incoming edges

*neighbor here means all of the vertices a vertex can reach in one step

INSTRUCTIONS: Download the following three files from blackboard: (1) `adj_list.cpp` (adjacency list implementation), (2) `adj_matrix.cpp` (adjacency matrix implementation), and (3) `graph.h` (header file with the class definitions).

The driver program (i.e., the code in the main function) has been provided to you. The code for the driver program is exactly the same in the `adj_list.cpp` and `adj_matrix.cpp` files.

The class definitions (i.e., member functions and data variables) have been provided to you in the `graph.h` file. Your implementation should only use the graph class that has been provided. No other classes are needed.

Please also take a look at the “Lab 1 sample runs” file to get an idea of how the output in the console window should look in your programs.

When you are finished, submit the three files (`adj_list.cpp`, `adj_matrix.cpp`, and `graph.h`) on Mulan. Make sure to also **thoroughly** test both implementations!

Adjacency List

An adjacency list is an array (or vector in our case) of pointers that each point to a linked list (doubly linked list in our case).

Adjacency Matrix

An adjacency matrix is a two dimensional array (our vector in our case).

Here is a figure from page 384 of the textbook, which shows a directed graph along with its adjacency list and adjacency matrix implementations.

384

Chap. 11 Graphs

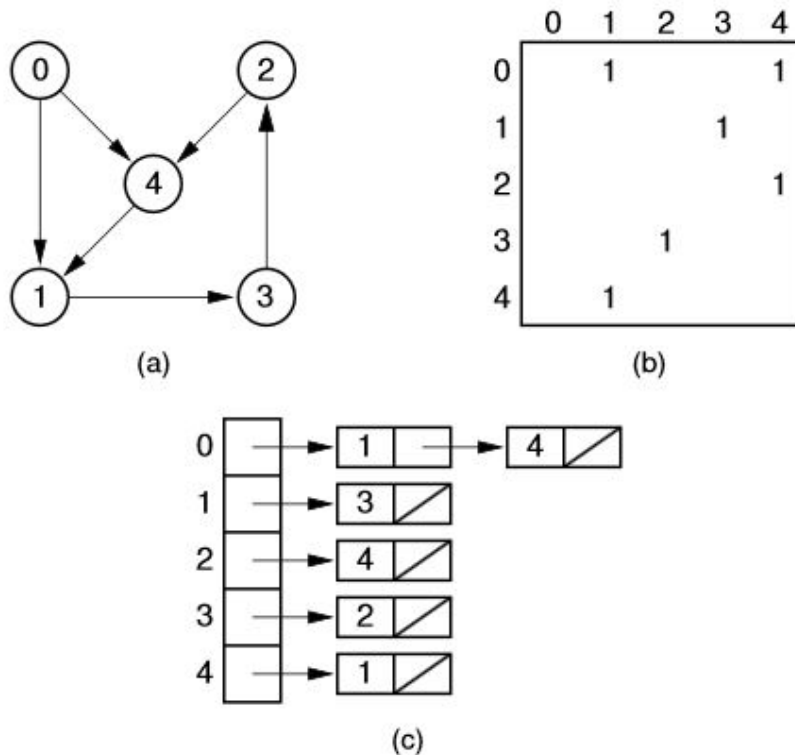


Figure 11.3 Two graph representations. (a) A directed graph. (b) The adjacency matrix for the graph of (a). (c) The adjacency list for the graph of (a).