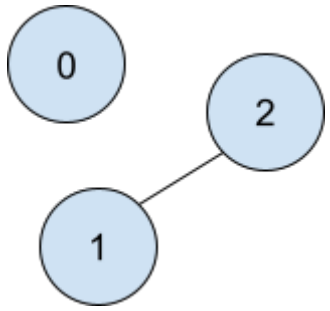
Rafid
Shaon
CSC 4520
Nov 08 2022

HW4: Graphs, Traversals

Reference

Example Undirected Graph

Visual	Adjacency List	Adjacency Matrix

	[[], [2], [1]]	[
		  [0, 0, 0]
		  [0, 0, 1]
		  [0, 1, 0]
		]

Questions start On next page ↴

Q1. Given an undirected graph with 10 vertices but 45 edges, which of the following will be true? Select all that apply.

A) Every node will have at least one neighbor
B) The graph will have duplicate labels
C) An Adjacency List would be more space-efficient to represent this graph vs. a Matrix
D) The graph is GUARANTEED to have a cycle in it
E) None of the above

A and D are true.

Q2. There are N cities, with M roads, where each road connects a pair of cities. You are given city **x**. We want to see if every city has a direct road leading to **x**. Return true if this condition is met, and false otherwise.

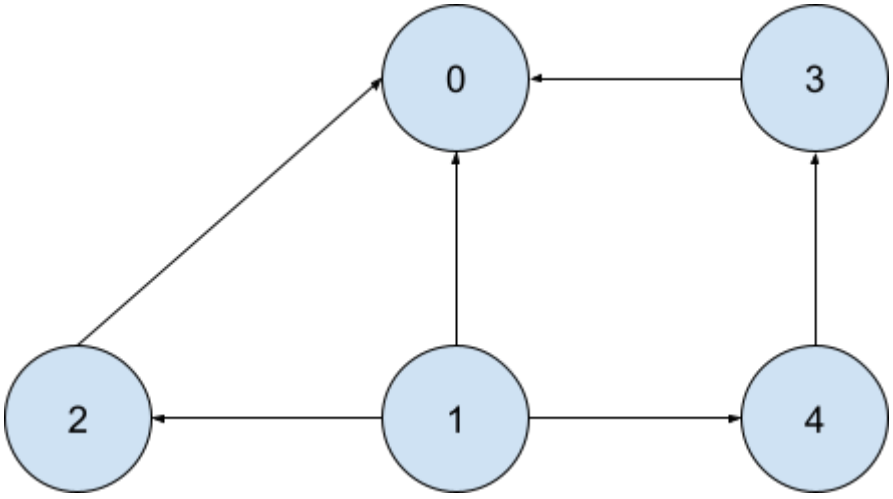
A) If we represent this as a graph problem, what are the nodes and edges?
The nodes are the cities. Each node in a graph serves as a point of intersection for edges. The edges are the roads. Each edge in a graph connects points between nodes.

B) Is this a directed or undirected graph? – Please consider the situation between two cities in the real world. The question is not about two places in a city.
This is an undirected graph; its edges do not have arrows pointing towards the node. In the real world, the roads between two cities usually have cars traversing in two directions. These two-directional roads act as edges with no pointing arrows.

C) Given an adjacency matrix for this graph, describe using words how you would find the answer to this problem. You do not have to write code.
Since the cities will be connected by roads, I would have to traverse through the connected roads. If two cities are connected via roads, the matrix representation is 1; if two cities are not connected at all, the matrix representation is 1. I would need to find the connection for every city shown in the graph and insert which cities connect or not inside the adjacency matrix.

D) Given an adjacency list for this graph, describe using words how you would find the answer to this problem. You do not have to write code.
Similar to the adjacency matrix, I would need to traverse through the roads connecting the cities. Assume there are Cities A, B, C, and D. City A is connected with City B and C, City B is connected with City A and D, City C is connected with City A, and City D is connected with City B. For every city, I would need to write down all of the cities it is connected to. City A: [B, C], City B: [A, D], and so on.

Q3. Given the graph below, answer the following questions.



A) Represent this graph as an adjacency list.

List: [[0], [0, 2, 4], [0], [0], [3]]

0: [0]
1: [0, 2, 4]
2: [0]
3: [0]
4: [3]

B) Represent this graph as an adjacency matrix.

matrix[*u*][*v*] = **1** if an edge exists from node *u* to node *v*

matrix[*u*][*v*] = **0** if an edge does not exist from node *u* to node *v*

	0	**1**	**2**	**3**	**4**
0	0	0	0	0	0
1	1	0	1	0	1
2	1	0	0	0	0
3	1	0	0	0	0
4	0	0	0	1	0

C) What is the ordering of nodes If we run Graph DFS starting on node 1? Assume we visit the smallest neighbour first.

When running Graph DFS starting on node 1, the ordering of nodes is

[1, 0, 2, 4, 3] == 1 → 0 → 2 → 4 → 3

D) Write your Java code(submit a .java file) to implement the DFS for graph traversal using the adjacency matrix (either recursive or iterative).
For the test case, you can directly use the above example. And you should call the DFS function several times with different starting points to show the different traversal orders.

DFS(graph, 0); // one possible output likes 0
DFS(graph, 1); // one possible output likes 1 0 2 4 3
DFS(graph, 2); // …
DFS(graph, 3); // …
DFS(graph, 4); // …

My Code: Graph.java & Q3D.java
* these files will be submitted into the Dropbox