CSC 403 ICE 7 Name Brian Bauman

Note: this is the handout from class – for your reference.

You only need to answer questions 1 & 2 on this page for this ICE assignment.

1. Choose two substantive lecture content questions (by anyone) – one before and one after the break. State the question and answer in your own words.
2. @ 1:17:45, you ask “What would the code look like for this function [degree()]? You pass in a graph and a vertex.”

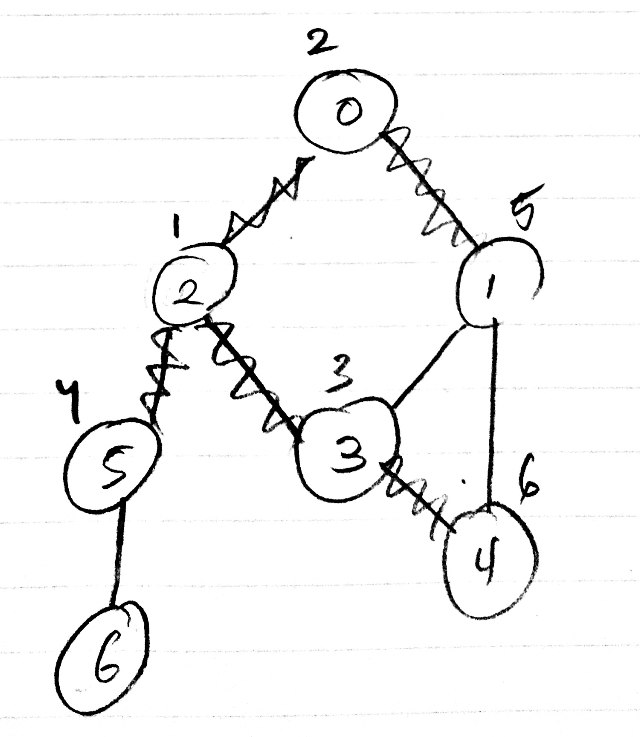
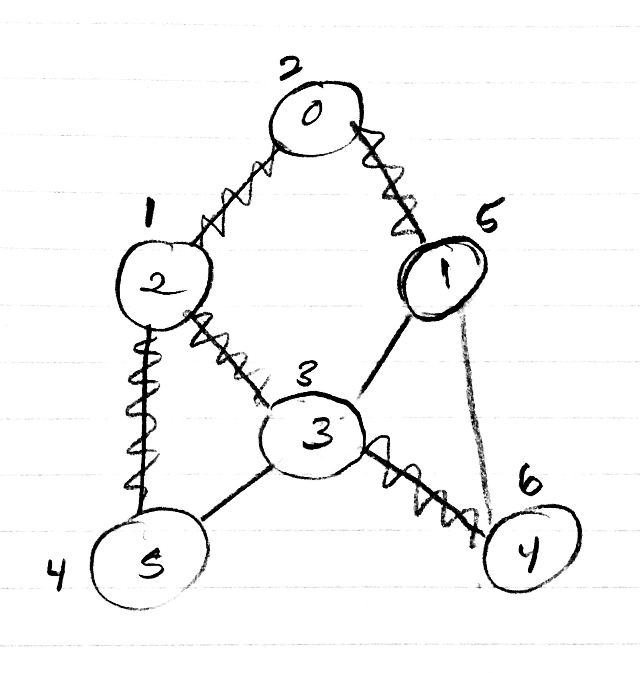
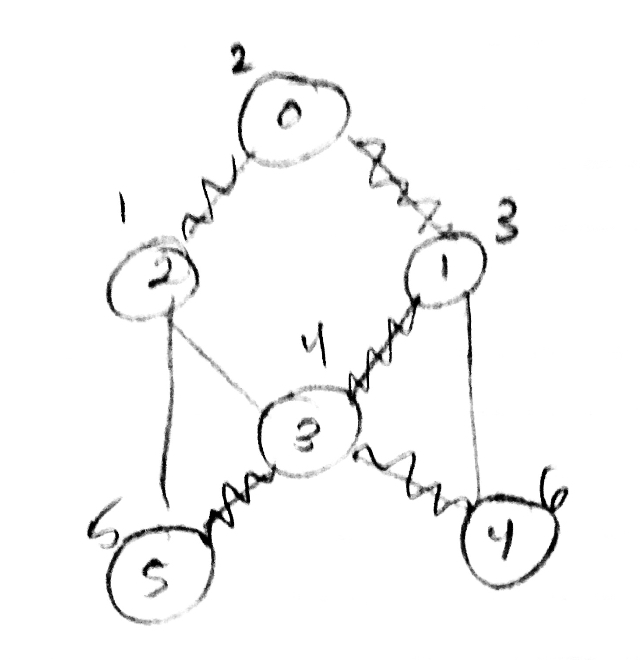
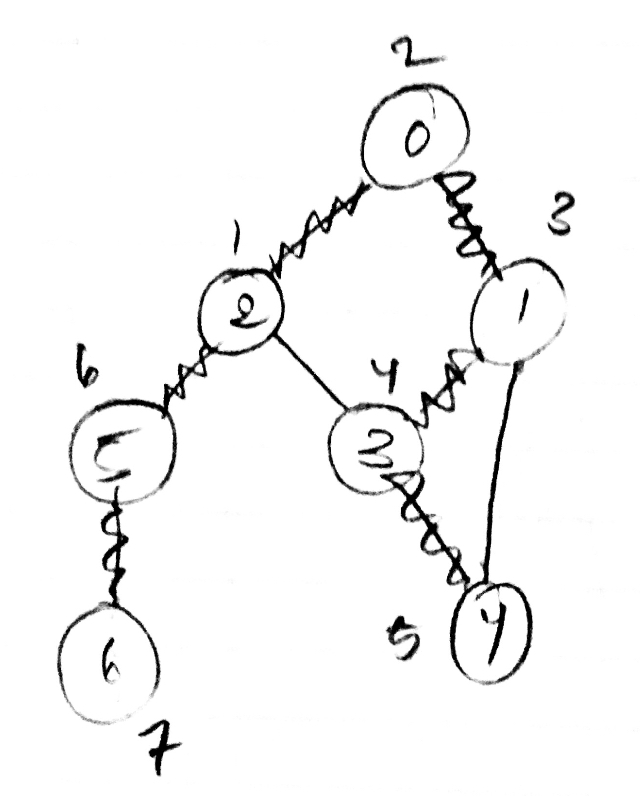
Basically, you want to use the existing API functionality to retrieve an iterator of adjacent vertices to the inputted vertex. You can count through this list of vertices to determine the degree of the original vertex.

b) @ 2:34:29, a student asks “Does shortest path combine depth-first and breadth-first search?”

Either method can be used to solve the problem.

1. Trace BFS and DFS starting at node 2 on the graphs given. Label each vertex by the order in which it was visited. Also highlight the edges that were used as part of the traversal.

BFS DFS



*count self-loops*

*compute the degree of* v

public static int degree(Graph G, int v)

{

}

*compute maximum degree*

public static int maxDegree(Graph G)

{

}

*compute average degree*

public static double averageDegree(Graph G)

{

}

*count self-loops*

public static int numberOfSelfLoops(Graph G)

{

// each edge counted twice