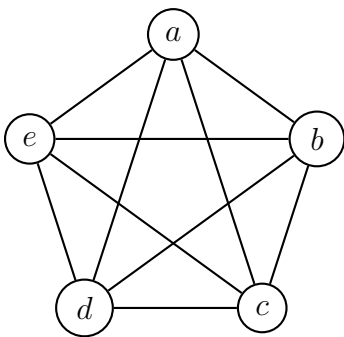CS 5970/6970: Graph Algorithms
August 25, 2023

Homework 1

Make sure you have python3 (mac's ship with python2 on the PATH, but python2 is depricated) installed. Get the networkx package via pip or conda etc.
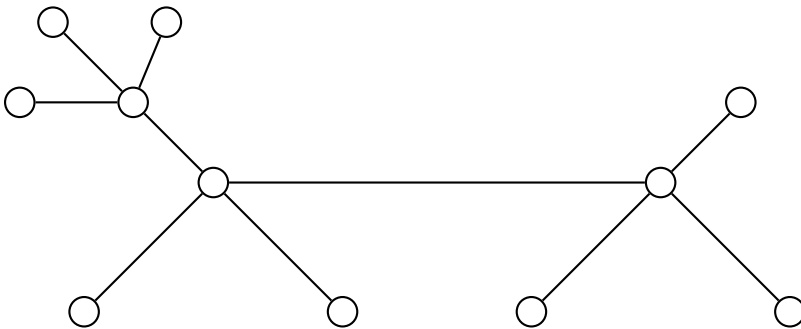
Question 1. List as many properties of the following graphs as you can and the categories of graphs each of the following graphs belong to. (5 points each)

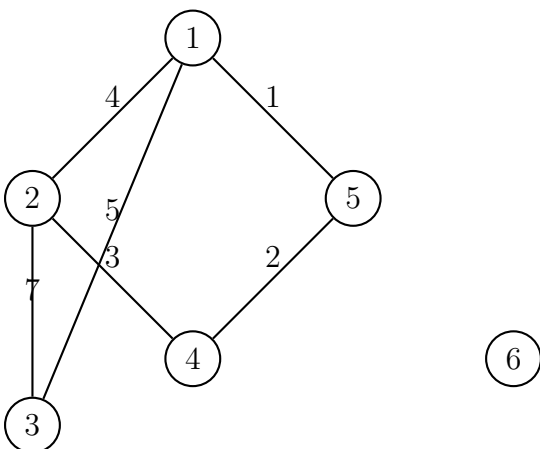**Complete, unweighted, undirected, connected**

5 vertices, 10 edges

Vertex degree: 4

Simple, unweighted, undirected, connected
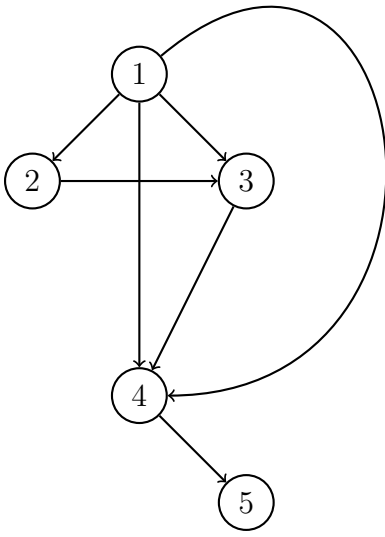
11 vertices, 10 edges

Vertex degree: 1 min/ 4 max

Weighted, undirected, disconnected

6 vertices, 6 edges

Vertex degree: 0 min/ 3 max
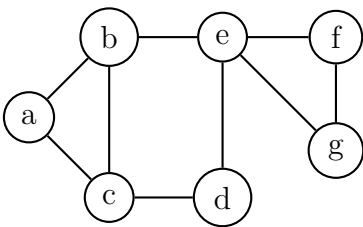
Unweighted, directed, connected

5 vertices, 7 edges

Multiedge 1 -> 4

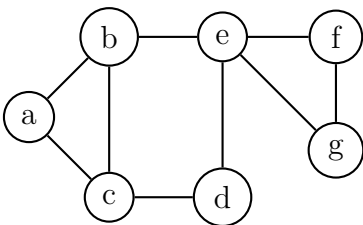**Question 2.** Create the adjacency matrix for the last graph from question 1 (5 points).

	1	2	3	4	5
1	0	1	1	1	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1
5	0	0	0	0	0

Question 3. On the following graph find a walk $b \to d$ that is not a trail. (5 points)
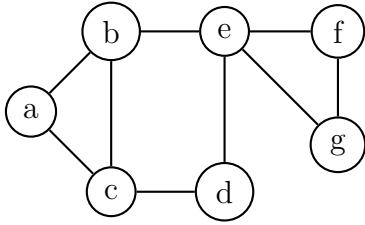
b->e, e->g, g->f, f->e, e->b, b->c, c->d

Question 4. On the following graph find a trail $b \to d$ that is not a path. (5 points)
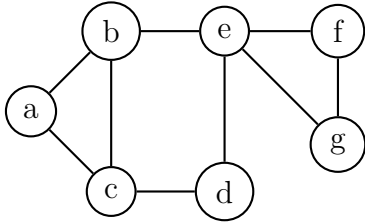
b->e, e->g, g->f, f->e, e->d,

Question 5. On the following graph find a path $b \to d$. (5 points)

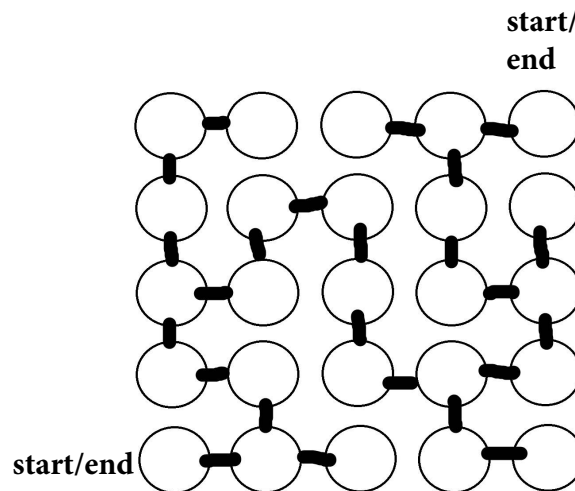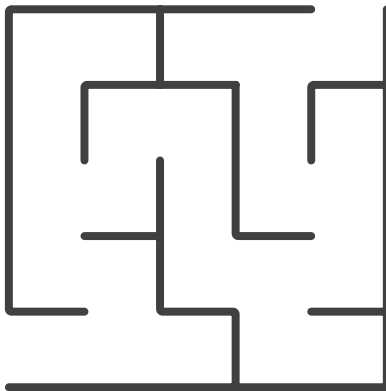**b -> c, c-> d**

Question 6. On the following graph find a trail $b \to d$ that is not a path. (5 points)

b->e, e->g, g->f, f->e, e->d,

Question 7. Given the following simple maze, convert the maze into a graph. (5 points)

Question 8. Implement depth first search in the maze function in homework1.py. (25 points)

Question 9. All possible chess games can be thought of as a tree of chess moves starting at the initial position with then all legal moves from each position as children of each other position. In the function checkmate, implement breadth first search on the given move tree and root node to find the fastest possible checkmate. The output should be the sequence of moves. Eg e2e4 e7e5 .. .. The graph stores a text representation of the game at every node as well as the move to get to that position and the text representation of the board at the previous position (the parent node). (25 points)