TheoryPrep **(Due: 6/28/20)**

Assignment #4: Graph Theory

Name: Student name(s)

We've taken some problems from the course texts. Some problems are marked as (Challenge) or (Additional Problems). Do the rest first, and if you have time, return to these problems. It's **much more important** to have a rigorous understanding of the core problems and how to prove them than to simply finish all the additional problems.

Problem 1: Directed Graphs

Learning goal: These proofs will help you get familiar with directed graphs and common definitions we use with them. Our study of graphs will also help you reason about abstract algebraic objects–we're taking the discrete math skills that we learned on structures we had more experience with (ie. the numbers in the early weeks) and applying them to a new definition!

(a) Below are a list of CS classes with loose prerequisites

- CS 181: CS 50, Stat 110
- CS 121: CS 20
- CS 124: CS 50, CS 51, CS 121, Stat 110
- CS 51: CS 50
- CS 61: CS 50

Draw a directed graph of all of these course (including those with no prerequisites such as CS 50). The edges should go from prerequisites to the courses they are prerequisites for. Is this a DAG? Interpret this–if there are cycles why does that make sense, and if there are no cycles, why would that not make sense?

Assuming you can take an unlimited number of courses per semester, how many semesters would it take you to complete all the classes in the list? Assume you must take prerequisites for any course before it.

Solution:

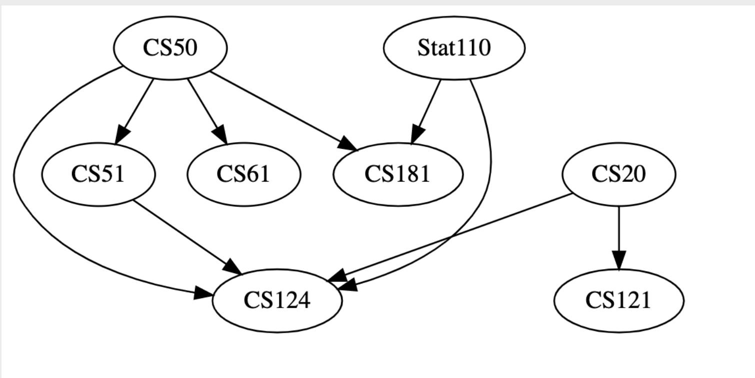

Figure 1: The DAG

This is a DAG. It would not make sense for there to be a cycle because then there would be an infinte chain of prerequisites, such that you couldn't take any course!

Semester 1: CS50, CS20, Stat110. Semester 2: CS51, CS61, CS181, CS121. Semester

3: CS124. We know this is the minimal, since it's not possible to take CS124 any earlier than the third semester (since you must take CS51 in a preceding semester and CS50 in a preceding semester before that).

(b) Let S be a set of finite size and $f : S \to S$ a function. Let G be the graph with edges $E = \{(s, f(s))|s \in S\}$.

1. What are the possible in-degrees and out-degrees of the vertices in G?
2. Suppose f is a surjection. What are the possible in-degrees of the vertices in G?

Solution:

1. The out degree of every vertex is 1, since a function maps an input to only one output. The in-degrees of the vertices must also then sum to $|S|$, but subject to that constraint, for a given vertex, its in-degree could be everywhere from 0 (if nothing maps to it) to $|S|$.
2. If f is a surjection, then it is also a bijection (because its domain and range are finite), so the in-degree of every vertex is 1.

(c) Prove that every odd-length closed walk contains a vertex with an odd-length cycle (as a reminder we assume simple graphs with no self-loops). However, this does not preclude a singular vertex from being in an odd-length, closed walk but not in an odd length cycle. Give an example of such a vertex.

Solution:
We prove the first statement by induction–we induct on the number of vertices in the odd-length cycle. No self loops means the base case is $n = 3$. In this case, we see an odd-length closed walk of 3 edges. The first and last vertex must be the same, so since there are no self-loops, all vertices (excepte the first and last) must be different, so this is an odd-length cycle.

Now for the inductive step. Suppose we have an odd-length closed walk going from u to u. If it is not a cycle, this is only because it repeats some vertex, call it v. We can write the walk as

$$u, u_1, u_2, ..., u_n, v, u_{n+1}, ..., u_m, v, u_{m+1}, ..., u_k, u \tag{0.1}$$

Now we break this into two walks. $u, u_1, u_2, ..., u_n, v, u_{m+1}, ..., u_k, u$ and $v, u_{n+1}, ..., u_m, v$. Note that these both have length shorter than the original walk, and since two numbers that sum to an odd number must be odd and even, one is a shorter, odd-length closed walk. By the inductive hypothesis, this contains a vertex with an odd-length cycle.

We give an example below. In the graph below, 6 is in an odd-length closed walk of $6, 1, 4, 2, 3, 0, 2, 4, 5, 6$, but only in an even-length cycle.

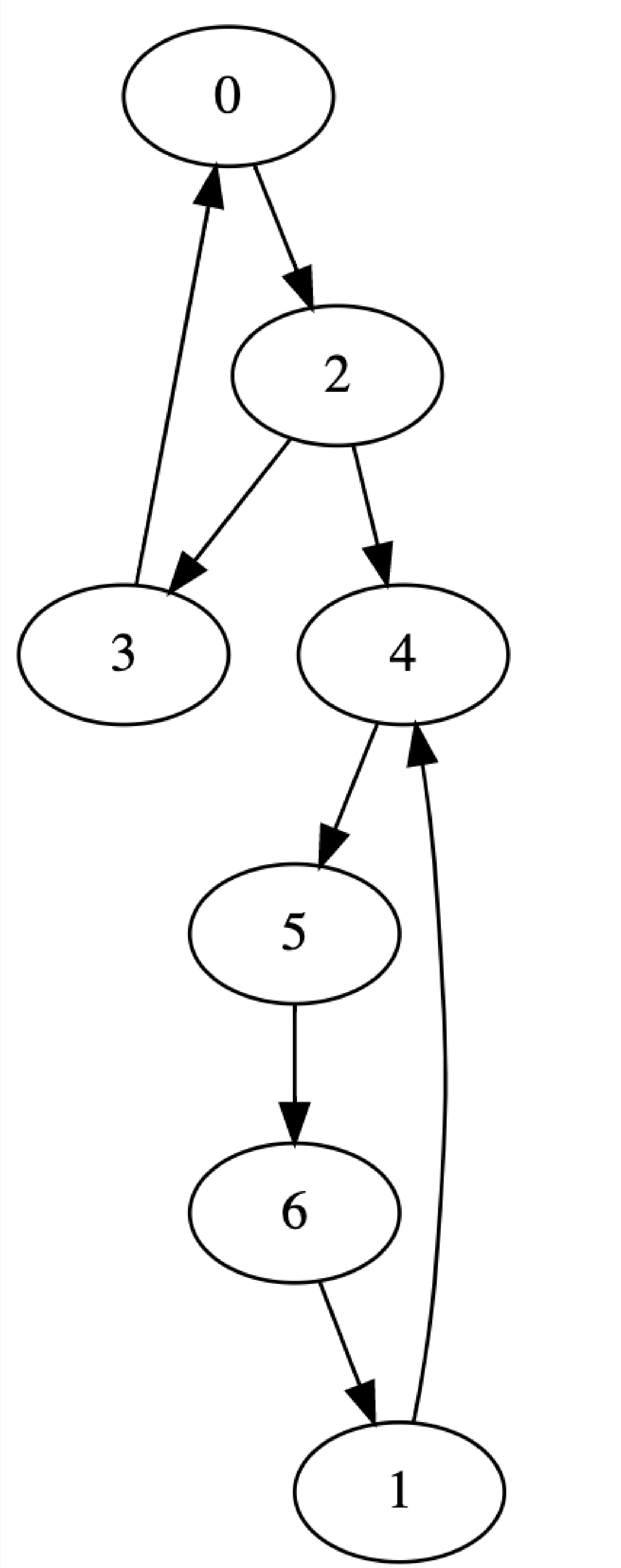

Figure 2: A counterexample

(d) The triangle inequality says that for any three vertices

$$dist(u, v) \leq dist(u, w) + dist(w, v) \tag{0.2}$$

The equality holds if and only if w is on a shortest path between u and v. Prove this iff statement. You do not need to prove the inequality since this is proven in the textbook

Solution:
First to prove the forwards direction. Suppose that $dist(u, v) = dist(u, w) + dist(w, v)$. Let S be the shortest path from u to w and T be the shortest path from w to v. Clearly, $|S| = dist(u, w), |T| = dist(w, v)$, so $dist(u, v) = |S| + |T|$, so $S \wedge T$ (the concatenation of S into T) is a shortest path from u to v containing w.

Now to prove the backwards direction. If w is on a shortest path from u to v, let n denote the length of that path. $dist(u, v) = n + m$, where n is the lenght of a possible path from u to w and m is the length of a possible path from w to v. Thus, $dist(u, w) \leq n, dist(w, v) \leq m$. Therefore, $dist(u, v) \leq dist(u, w) + dist(w, v)$, so $dist(u, v) = dist(u, w) + dist(w, v)$.

Problem 2: Undirected Graphs

Learning goal: Similar overall learning goals of learning how to use our proof strategies on a new object we may not have intuition for. Also, you'll learn some common concepts that we use when working with undirected graphs.

(a) Suppose a simple, connected graph has vertex degrees that sum to 20. What is the greatest and least number of possible vertices in the graph? (Understanding definitions)

Solution:
The greatest number of possible vertices in the graph is 11 (vertex degree 20 implies 10 edges–we know this since a tree has the fewest number of edges while remaining connected and has one fewer edges than vertices). The least number of possible vertices is 5 since a fully connected graph would have 10 edges and therefore vertex degree of 20.

(b) Which of the following graph properties are preserved under isomorphism?

1. There is a cycle including all the vertices.
2. The vertices are numbered 1 through 7.
3. There are two degree 8 vertices.
4. No matter which edge is removed, there exists a path between all vertices.
5. The vertices are a set.
6. Two edges are of equal length (when drawn on the paper).
7. The graph can be drawn in a way such that all edges are the same length (on the paper).
8. The OR of two properties preserved under isomorphism.
9. The NOT of a property preserved under isomorphism.

Solution:

1. Preserved
2. Not Preserved
3. Preserved
4. Preserved
5. Preserved (always true)
6. Not Preserved
7. Preserved (note the **can be**)
8. Preserved
9. Preserved

(c) A simple graph is regular if every vertex has the same degree. Call a graph balanced if it is bipartite, regular, and has the same number of left and right vertices (very balanced as the name suggest). Prove that if graph G is balanced, its edges can be partitioned into perfect matchings (a matching that covers all vertices–intuitively, a pairing of all elements). As an example, suppose that graph G has $2k$ vertices, each of degree j. Then we can partition the kj edges into j sets of size k, which are each perfect matchings.

Solution:
We'll use the theorem we proved in the video (if students are stuck, remind them of this). For a balanced graph, let's call the degree of every vertex (they are the same since the graph is regular) it's thickness–mathematically this is given by $\frac{2|E|}{|V|}$ ($2|E|$ is the sum of the degrees of vertices, divided by number of vertices gives vertex degree).

First the base case, and the graph has thickness 0. It therefore has no edges, so we can partition it into perfect matchings.

Second, suppose that the graph has thickness $n+1$. The theorem that we proved in the video said that there was a perfect matching (note the theorem discusses matchings covering L, but since we have the same number of left and right vertices, ever matching will be perfect), if $\forall S \subseteq L. |S| \leq |N(S)|$. I will prove that this is true. Consider a set S of vertices on the left. If $|S| \leq n+1$, then a singular lefthand vertex has $n+1$ outgoing edges, so must have at least $n+1$ neighbors, so thus $|S| \leq |N(S)|$. If $|S| > n+1$, suppose towards a contradiction that $|N(S)| < |S|$. The number of outgoing edges from the points in S is $|S|(n+1)$, these edges must all terminate in $N(S)$, so at least one vertex in $N(S)$ must have degree $\lceil \frac{|S|(n+1)}{|N(S)|} \rceil > n+1$ (this is called the averaging principle–if the average of $a_1, ..., a_n$ is m, then at least one of those quantities must be at least m. In this case, we know that this must be $\lceil m \rceil$ since degrees are integers). This is a contradiction since every vertex has degree $n+1$. Thus, we see that $\forall S \subseteq L. |S| \leq |N(S)|$, so there is a perfect matching.

Let M be the perfect matching. Consider the graph $(V, E-M)$, so in particular, we're getting rid of the edges in the matching. I claim the resulting graph is still balanced. In particular, removing the perfect matching decreases every vertex degree by exactly 1, so it is still regular, and the other properties depend only on the vertices, which remain unchanged. We see the thickness decrease by exactly one since we are removing exactly $|M| = \frac{|V|}{2}$ edges, so $\frac{2|E-M|}{|V|} = \frac{2|E|}{|V|} - \frac{2|M|}{|V|} = \frac{2|E|}{|V|} - 1$. Thus, by the inductive hypothesis, the remaining edges can also be partitioned into perfect matchings, giving a partition for the entire set of edges.

(d) Let $N(u) = \{v | \{u, v\} \in E\}$ be the set of neighbors of u in an undirected graph $G = (V, E)$. Let f be an isomorphism on G. We will prove that neighbors are preserved under isomorphism, or in other words

$$N(f(u)) = f(N(u)) \tag{0.3}$$

Solution:
We will prove that $h \in N(f(u)) \Leftrightarrow h \in f(N(u))$. Let E' be the set of edges under the isomorphism. We know $h \in N(f(u)) \Leftrightarrow \{h, f(u)\} \in E'$ by defintion. This is equivalent to $\{f^{-1}(h), u\} \in E$ (note f is a bijection so we can take the inverse) since f is an isomorphism, which by definition means $f^{-1}(h) \in N(u)$. That's equivalent to $h \in f(N(u))$ (forwards direction by applying f, backwards direction by applying f^{-1}).

Problem 3: Longer Proofs

Learning Goal: We want you to be familiar doing proofs that take multiple steps. Doing longer proofs also includes being comfortable trying to find the answer for extended periods of time–that's a natural part of the process!

(a) We say that an undirected graph is a tree if it is connected and acyclic. However, this is equivalent to many other conditions. In particular, prove that the below conditions are the same (this is an interesting result that there are so many simple characterizations of the exact same class of objects!)

1. G is a tree
2. G is acyclic, but adding any additional edge would form a cycle.
3. Any two vertices in G are connected by a unique path.
4. G is connected, but it would not be connected were any edge removed.

Solution:
We will prove a circle of implications, which implies equivalence (If $A \Rightarrow B \Rightarrow C \Rightarrow A$, then starting from any, you can get to any other, so they are equivalent).

(1$\Rightarrow$ 2) If G is a tree, then it is acyclic. We need to prove adding any additional edge would form a cycle. Suppose you add an edge $\{u, v\}$. Since the graph was previously connected, there was some path $u, ..., v$ connecting u to v, so now you have a cycle $u, ..., v, u$.

(2$\Rightarrow$3) Suppose G is acyclic, but adding any additional edge would form a cycle. Fix u, v arbitrary vertices in G. If they are not connected, then we could add the edge $\{u, v\}$ and not form a cycle (because since the graph was previous acyclic, any formed cycle must include $\{u, v\}$, so we could use the rest of the cycle as a path in the original graph between u, v), so there must exist a path between them. We now prove that path is unique. Suppose the path is not unique, so there is some path $u, a_1, ..., a_n, v$ and some path $u, b_1, ..., b_n, v$.

Consider the iterative walks of $a_i, ..., a_n, v, b_n, ..., b_1, u$, starting with $i = n$ and going down (if we get to one, we then consider $u, a_1, ..., a_n, v, b_n, ..., b_1, u$). I claim that at every step, we will either find a cycle, or $a_i, ..., a_n, v, b_n, ..., b_1, u$ has no repeated vertices. At each step, check to see if adding the new a_i is the same as any of the existing b_j (it cannot be the same as u or v since it was on a path with those vertices). If so, then $a_i, ..., a_n, v, b_n, ..., b_j$ is a cycle. If not, we see that $a_i, ..., a_n, v, b_n, ..., b_1, u$ has no repeated vertices. This process must terminate because then $u, a_1, ..., a_n, v, b_n, ..., b_1, u$ is a cycle, if none of the a_i repeat a b_j. If students are struggling, it's okay to have them skip this and assert that two different paths imply a cycle without proof.

(3$\Rightarrow$4) G is connected since any two vertices have a path between them. Consider removing any edge $\{u, v\}$, and suppose the graph is still connected, so there is some path $u, a_1, ..., a_n, v$. Therefore, in the original graph, there are two paths between u and v, namely u, v and $u, a_1, ..., a_n, v$. Thus, we see that the graph must not be connected if any edge is removed.

(4⇒1) G is connected, so we only need to prove it's acyclic. Suppose towards a contradiction that there is a cycle $u, a_1, ..., a_n, u$. I claim if we remove the edge $\{a_n, u\}$, the graph will still be connected. For any two vertices, use the same path as in the original graph, but if you need to use edge $\{u, a_n\}$, replace it with $a_n, ..., a_1, u$. This will give a walk between v_1, v_2. This contradicts that if you remove any edge then it will not be connected. Hence, the graph must be acyclic.

(b) We now define a concept called a binary tree. Binary trees are defined inductively as follows

- A single vertex, called a root, is a binary tree.
- A binary tree can be constructed by taking two binary trees, b, b' adding another vertex r, and connecting r to the roots of b and b'. r is the root of the new tree.

We say that the height of a binary tree is the number of edges between the root and the farthest leaf. Prove that a binary tree of height k has at most 2^k leaves.

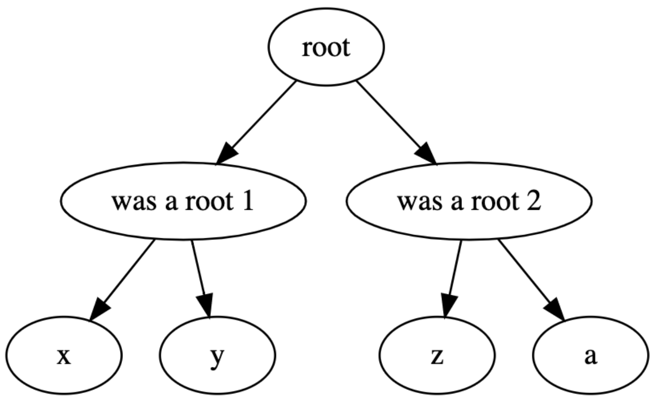

Figure 3: Example binary tree with height 2

Solution:
We prove this inductively, inducting on height. If the height is 0, then the binary tree is a single vertex, and therefore has $2^0 = 1$ leaves.

Suppose we have a binary tree T of height $n + 1$. It must have been constructed by merging together two binary trees b, b'. The maximum height of those binary trees is n, since if they had height $> n$, then the height of the resulting tree would have been $> n + 1$. Therefore, applying the inductive hypothesis, b, b' had at most 2^n leaves, so T has at most 2^{n+1} leaves.

(c) (Additional Problem) Suppose you are a road surveyor and you must test each of the roads in a a town, which you represented as a directed graph (because some of the streets are one way and you need to test both sides of bi-directional roads). However, you are not keen on rewalking roads and would like to end up in your starting spot, so your goal is to walk each road exactly once and come back to the same place–this corresponds to a closed walk that traverses each edge exactly once, an Euler Tour. In this problem, we'll prove a necessary and sufficient condition for a graph to have an Euler Tour. First let's figure out some properties of an Euler Tour

1. Show that if a graph has an Euler Tour, then the in-degree of every vertex is equal to its out-degree.

   This suggests a possible condition. We will prove "A weakly connected (which means there is a path between any two vertices if you replaced all the edges as undirected) digraph G has an Euler Tour if and only if it is every vertex's in-degree is equal to its out-degree." Note we've proved the forward direction of this theorem.

2. Now we will prove the other direction, so **assume for the remainder of the problem** that G is weakly connected and that every vertex has the same in and out-degree. We'll first prove an intermediary result that will help us. A trail is a walk that traverses no edge more than once. Suppose that a trail does not include every edge. Prove that there must be an edge that goes into or out of a vertex on the trail.
3. Let w be the longest trail in the graph. Prove that if w is closed, it is a Euler Tour (try using the result from the previous part).
4. Let v be the last vertex on w. Explain why all edges leaving v must be on w.
5. Prove that if w was not closed, then the in-degree of v would be greater than its out degree.
6. Conclude that if a weakly-connected graph has equal in and out-degree of every vertex, then it has an Euler tour.

Solution:

1. Let n_v be the number of times a vertex v is visited during the euler tour (for the starting node, count starting and ending there as one visit). Thus, the node v is entered n_v times and exited n_v times, each time on a different edge. Since the Euler tour traverses every edge exactly once, v has in-degree and out-degree n_v.
2. If the trail covers all the vertices, then any additional edge must come from one of the traversed vertices. If the trail does not cover all vertices, pick a vertex v on the trail and a vertex u not on the trail. Since the graph is weakly connected, treating all edges as undirected, we have a path $u, a_1, ..., a_n, v$. We see that there is some a_i such that i is the smallest possible number such that a_i is on the trail. Thus, a_{i-1}, a_i share an edge, and a_{i-1} is not on the trail, so this edge is not on the trail but has an endpoint on it.
3. Suppose w is not an Euler Tour, so by the previous part, there is an edge that goes into our out of the trail, say at vertex v. Then since w is closed, we can create a trail of length one greater by prepending or postpending this edge. This is a contradiction. Thus, w is an Euler Tour.
4. If this is not the case, add an edge leaving v onto w to make a longer trail, which is a contradiction with the maximality of the length of w.
5. If w is not closed, v is not the starting vertex for the trail. Let n be the number of times that v is visited, aside from when the trail ends there. Since each entry into and exit from v are on different edges, this implies that there are at least $n + 1$ edges into v, and by part 4, all edges leaving v must be on w, so the out-degree of v is n. This implies that the in-degree of v is greater than it's out-degree.
6. Part 5 implies that if w is not closed, then v has greater in-degree than out-degree, so w i not closed. Therefore it is an Euler Tour.

Problem 4: Review

Learning goal: The goal for this section is to reflect on how you can improve your proof skills!

(a) Select the problem from a previous week that you think you would learn the most from redoing. Copy the problem and your instructor's feedback below, rewrite the proof, and add a short reflection on the changes you made.

(b) (Additional Question) Prove that if $f : A \to B$ is a function then f surjective if and only if there exists some $g : B \to A$ such that $\forall b \in B. f \circ g(b) = b$.

Solution:
We first prove the forwards direction. Let f be a surjection. Define g as follows. For any $b \in B$, let a be some a such that $f(a) = b$ (we know this exists since f is a surjection). Define $g(b) = a$. Now to prove that $f \circ g$ is the identity function. For any $b \in B$ $f(g(b)) = b$ since $g(b)$ gives an a such that $f(a) = b$.

Now for the backwards direction. Suppose that $f \circ g(b) = b$ for any $b \in B$. I claim f is a surjection. Suppose towards a contradiction that f is not a surjection, so $\forall a \in A.f(a) \neq b'$. Then, $\forall b \in B.f(g(b)) \neq b$. This contradicts that $f \circ g$ is the identity map.

Problem 5: Logistics

Purpose: This helps us make sure the course is going at the right speed!

(a) How long did you spend on the videos and readings this week?

(b) How long (including time in problem sessions) did you spend on this problem set?

(c) Do you have any feedback about the course in general (did the videos and readings sufficiently prepare you for the problem set)?