CS 70 Discrete Mathematics and Probability Theory
Spring 2018 Satish Rao and Babak Ayazifar

HW 3

Sundry

Before you start your homework, write down your team. Who else did you work with on this homework? List names and email addresses. (In case of homework party, you can also just describe the group.) How did you work on this homework? Working in groups of 3-5 will earn credit for your "Sundry" grade.

Please copy the following statement and sign next to it:

I certify that all solutions are entirely in my words and that I have not looked at another student's solutions. I have credited all external sources in this write up.

I certify that all solutions are entirely in my words and that I have not looked at another student's solutions. I have credited all external sources in this write up. (*signature here*)

1 Build-Up Error?

What is wrong with the following "proof"? In addition to finding a counterexample, you should explain what is fundamentally wrong with this approach, and why it demonstrates the danger build-up error.

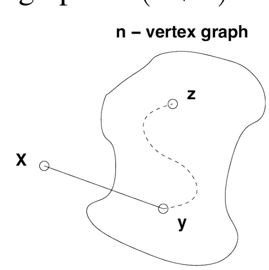
False Claim: If every vertex in an undirected graph has degree at least 1, then the graph is connected.

Proof: We use induction on the number of vertices $n \geq 1$.

Base case: There is only one graph with a single vertex and it has degree 0. Therefore, the base case is vacuously true, since the if-part is false.

Inductive hypothesis: Assume the claim is true for some $n \geq 1$.

Inductive step: We prove the claim is also true for $n+1$. Consider an undirected graph on n vertices in which every vertex has degree at least 1. By the inductive hypothesis, this graph is connected. Now add one more vertex x to obtain a graph on $(n+1)$ vertices, as shown below.



All that remains is to check that there is a path from x to every other vertex z. Since x has degree at least 1, there is an edge from x to some other vertex; call it y. Thus, we can obtain a path from x to z by adjoining the edge $\{x,y\}$ to the path from y to z. This proves the claim for $n+1$.

Solution:

The mistake is in the argument that *"every $(n+1)$-vertex graph with minimum degree 1 can be obtained from an n-vertex graph with minimum degree 1 by adding 1 more vertex"*. Instead of starting by considering an arbitrary $(n+1)$-vertex graph, this proof only considers an $(n+1)$-vertex graph that you can make by starting with an n-vertex graph with minimum degree 1. As a counterexample, consider a graph on four vertices $V = \{1,2,3,4\}$ with two edges $E = \{\{1,2\},\{3,4\}\}$. Every vertex in this graph has degree 1, but there is no way to build this 4-vertex graph from a 3-vertex graph with minimum degree 1.

More generally, this is an example of *build-up error* in proof by induction. Usually this arises from a faulty assumption that every size $n+1$ graph with some property can be "built up" from a size n graph with the same property. (This assumption is correct for some properties, but incorrect for others, such as the one in the argument above.)

One way to avoid an accidental build-up error is to use a *"shrink down, grow back"* process in the inductive step: start with a size $n+1$ graph, remove a vertex (or edge), apply the inductive hypothesis $P(n)$ to the smaller graph, and then add back the vertex (or edge) and argue that $P(n+1)$ holds.

Let's see what would have happened if we'd tried to prove the claim above by this method. In the inductive step, we must show that $P(n)$ implies $P(n+1)$ for all $n \geq 1$. Consider an $(n+1)$- vertex graph G in which every vertex has degree at least 1. Remove an arbitrary vertex v, leaving an n-vertex graph G' in which every vertex has degree... uh-oh! The reduced graph G' might contain a vertex of degree 0, making the inductive hypothesis $P(n)$ inapplicable! We are stuck —— and properly so, since the claim is false!

2 Leaves in a Tree

A *leaf* in a tree is a vertex with degree 1.

(a) Consider a tree with $n \geq 3$ vertices. What is the largest possible number of leaves the tree could have? Prove that this maximum m is possible to achieve, and further that there cannot exist a tree with more than m leaves.

(b) Prove that every tree on $n \geq 2$ vertices must have at least two leaves.

(c) Let k be the maximum degree of a tree (The maximum degree of a graph is defined as the largest degree of any vertex in that graph). Prove that the tree contains at least k leaves.

Solution:

(a) We claim the maximum number of leaves is $n-1$. This is achieved when there is one vertex that is connected to all other vertices (this is called the *star graph*).

We now show that a tree on $n \geq 3$ vertices cannot have n leaves. Suppose the contrary that there is a tree on $n \geq 3$ vertices such that all its n vertices are leaves. Pick an arbitrary vertex x, and let y be its unique neighbor. Since x and y both have degree 1, the vertices x, y form a connected component separate from the rest of the tree, contradicting the fact that a tree is connected.

(b) We give a direct proof. Consider the longest path $\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{k-1}, v_k\}$ between two vertices $x = v_0$ and $y = v_k$ in the tree (here the length of a path is how many edges it uses, and if there are multiple longest paths then we just pick one of them). We claim that x and y must be leaves. Suppose the contrary that x is not a leaf, so it has degree at least two. This means x is adjacent to another vertex z different from v_1. Observe that z cannot appear in the path from x to y that we are considering, for otherwise there would be a cycle in the tree. Therefore, we can add the edge $\{z, x\}$ to our path to obtain a longer path in the tree, contradicting our earlier choice of the longest path. Thus, we conclude that x is a leaf. By the same argument, we conclude y is also a leaf.

The case when a tree has only two leaves is called the *path graph*, which is the graph on $V = \{1, 2, \ldots, n\}$ with edges $E = \{\{1,2\}, \{2,3\}, \ldots, \{n-1, n\}\}$.

(c) Consider a vertex v, which has degree k. Then, v has k neighbors. For each of v's neighbors, we can traverse down a path through that neighbor. We know that if we continue traversing, we must end at a leaf, because if we reach a vertex that is not a leaf, it has a degree greater than 1, so we can take that edge to continue traversing (and we know that this edge cannot be a part of a cycle). Furthermore, we know that for each of the k leaves that we end at through each of v's neighbors, no two of them can be the same leaf. To see this, assume for contradiction that we traversed through two of v's neighbors, u and w, and reached the same leaf l. Then we know there is a cycle in the graph, because we can travel from v to u to l to w back to v. Thus, no two discovered leaves can be the same, and so we have found at least k leaves in the tree.

Alternate proof (using degree argument):

For $k = 1$, we know that v has degree 1, so there is at least one leaf. Now consider $k \geq 2$.

For any graph, $\sum_{v \in V} \deg(v) = 2|E|$. Furthermore, we have shown in lecture that $|E| = |V| - 1$ for all trees. Then, we know that $\sum_{v \in V} \deg(v) = 2|V| - 2$.

We know that every leaf has degree 1 and every non-leaf has degree at least 2. Then, if we let L be the set of all leaves, we can conclude:

$$\sum_{x \in V} \deg(x) = \deg(v) + \sum_{u \in L} \deg(u) + \sum_{w \in V \setminus \{L \cup \{v\}\}} \deg(w) \geq k + |L| + 2(|V| - |L| - 1)$$

Then, we have

$$2|V| - 2 \geq k + 2|V| - |L| - 2$$

which implies that $|L| \geq k$. Thus, the number of leaves in the tree is at least k.

3 Bipartite Graphs

An undirected graph is bipartite if its vertices can be partitioned into two disjoint sets L, R such that each edge connects a vertex in L to a vertex in R (so there does not exist an edge that connects two vertices in L or two vertices in R).

(a) Suppose that a graph G is bipartite, with L and R being a bipartite partition of the vertices. Prove that $\sum_{v \in L} \deg(v) = \sum_{v \in R} \deg(v)$.

(b) Suppose that a graph G is bipartite, with L and R being a bipartite partition of the vertices. Let s and t denote the average degree of vertices in L and R respectively. Prove that $s/t = |R|/|L|$.

(c) A double of a graph G consists of two copies of G with edges joining the corresponding "mirror" points. Now suppose that G_1 is a bipartite graph, G_2 is a double of G_1, G_3 is a double of G_2, and so on. (Each G_{i+1} has twice as many vertices as G_i). Show that $\forall n \geq 1$, G_n is bipartite.

(d) Prove that a graph is bipartite if and only if it can be 2-colored. (A graph can be 2-colored if every vertex can be assigned one of two colors such that no two adjacent vertices have the same color).

Solution:

(a) Since G is bipartite, each edge connects one vertex in L with a vertex in R. Since each edge contributes equally to $\sum_{v \in L} \deg(v)$ and $\sum_{v \in R} \deg(v)$, we see that these two values must be equal.

(b) By part (a), we know that $\sum_{v \in L} \deg(v) = \sum_{v \in R} \deg(v)$. Thus $|L| \cdot s = |R| \cdot t$. A little algebra gives us the desired result.

(c) We use induction. Let $P(n)$ be the proposition that G_n is bipartite. The base case is when $n = 1$. We see that $P(1)$ must be true by definition (and construction) of G_1. Now suppose that for $k \geq 1$, $P(k)$ holds. We see that the graph G_{k+1} consists of two subgraphs, each having the same structure as G_k, except the edges joining the corresponding edges of the two subgraphs. Remove the extra edges. Since $P(k)$ is true, we can label the two subgraphs into disjoint sets $\{L_1, R_1\}$ and $\{L_2, R_2\}$. Then we can define new sets $L = \{L_1, R_2\}$ and $R = \{R_1, L_2\}$ that are disjoint. Every edge connects a vertex from L_1 to R_1 and from L_2 to R_2, so it connects from L to R. Adding back all the removed edges, we see that each added edge connects a vertex from L_1 to L_2 and from R_1 to R_2, so every added edge connects from L to R. Thus, the remaining graph G_{k+1} is still bipartite.

(d) Given a bipartite graph, color all of the vertices in L one color, and all of the vertices in R the other color. Conversely, given a 2-colored graph (call the colors red and blue), there are no edges between red vertices and red vertices, and there are no edges between blue vertices and blue vertices. Hence, take L to be the set of red vertices and R to be the set of blue vertices. We see that the graph is bipartite.

4 Edge-Disjoint Paths in a Hypercube

Prove that between any two distinct vertices x, y in the n-dimensional hypercube graph, there are at least n edge-disjoint paths from x to y (i.e., no two paths share an edge, though they may share vertices).

Solution:

We use induction on $n \geq 1$. The base case $n = 1$ holds because in this case the graph only has two vertices $V = \{0, 1\}$, and there is 1 path connecting them. Assume the claim holds for the $(n-1)$-dimensional hypercube. Let $x = x_1 x_2 \dots x_n$ and $y = y_1 y_2 \dots y_n$ be distinct vertices in the n-dimensional hypercube; we want to show there are at least n edge-disjoint paths from x to y. To do that, we consider two cases:

1. Suppose $x_i = y_i$ for some index $i \in \{1, \dots, n\}$. Without loss of generality (and for ease of explanation), we may assume $i = 1$, because the hypercube is symmetric with respect to the indices. Moreover, by interchanging the bits 0 and 1 if necessary, we may also assume $x_1 = y_1 = 0$. This means x and y both lie in the 0-subcube, where recall the 0-subcube (respectively, the 1-subcube) is the $(n-1)$-dimensional hypercube with vertices labeled $0z$ (respectively, $1z$) for $z \in \{0, 1\}^{n-1}$.

   Applying the inductive hypothesis, we know there are at least $n-1$ edge-disjoint paths from x to y, and moreover, these paths all lie within the 0-subcube. Clearly these $n-1$ paths will still be edge-disjoint in the original n-dimensional hypercube. We have an additional path from x to y that goes through the 1-subcube as follows: go from x to x', then from x' to y' following any path in the 1-subcube, and finally go from y' back to y. Here $x' = 1x_2 \dots x_n$ and $y' = 1y_2 \dots y_n$ are the corresponding points of x and y in the 1-subcube. Since this last path does not use any edges in the 0-subcube, this path is edge-disjoint to the $n-1$ paths that we have found. Therefore, we conclude that there are at least n edge-disjoint paths from x to y.

2. Suppose $x_i \neq y_i$ for all $i \in \{1, \dots, n\}$. This means x and y are two opposite vertices in the hypercube, and without loss of generality, we may assume $x = 00\dots0$ and $y = 11\dots1$. We explicitly exhibit n paths $P_1, \dots, P_n$ from x to y, and we claim they are edge-disjoint.

   For $i \in \{1, \dots, n\}$, the i-th path P_i is defined as follows: start from the vertex x (which is all zeros), flip the i-th bit to a 1, then keep flipping the bits one by one moving rightward from position $i+1$ to n, then from position 1 moving rightward to $i-1$. For example, the path P_1 is given by

   $$000\dots0 \to 100\dots0 \to 110\dots0 \to 111\dots0 \to \cdots \to 111\dots1$$

   while the path P_2 is given by

   $$000\dots0 \to 010\dots0 \to 011\dots0 \to \cdots \to 011\dots1 \to 111\dots1$$

   Note that the paths $P_1, \dots, P_n$ don't share vertices other than $x = 00\dots0$ and $y = 11\dots1$, so in particular they must be edge-disjoint.

Alternative for Case 2:
We can also reduce case 2, in which x and y have no bits in common, to case 1.

Suppose that $x_i \neq y_i$ for all $i = 1, \dots, n$. Let $\tilde{x}$ be x with the first bit flipped, so now $\tilde{x}$ and y both lie on a subcube together. From the inductive hypothesis, there are $n-1$ edge-disjoint paths from $\tilde{x}$ to y along the shared subcube (these paths only flip bits 2 through n because the paths lie entirely on a subcube, and these paths are of length at least $n-1$ since $\tilde{x}$ and y differ by that many bits). We would like to make these into edge-disjoint paths from y to x.

Starting at y, take the first of the $n-1$ paths, but before starting, flip the first bit. Then follow the first path to get a path from y to x. Then to use the second path starting at y, we travel one step along the second path, then flip the first bit, and then continue along the second path (carry out the sequence of bit flips in the second path, giving us another path from y to x). Continuing this way, we take the $(n-1)$st path, travel $n-2$ steps along the path, flip the first bit, then continue following the path.

It can be seen that this gives $n-1$ edge-disjoint paths from x to y. Where do we get the last path? Well we take the first path again, and now we follow the path for $n-1$ steps, and then flip the first bit – this gives us yet another edge-disjoint path for a total of n edge-disjoint paths! Cool!

5 Connectivity

Consider the following claims regarding connectivity:

(a) Prove: If G is a graph with n vertices such that for any two non-adjacent vertices u and v, it holds that $\deg u + \deg v \geq n-1$, then G is connected.

[*Hint:* Show something more specific: for any two non-adjacent vertices u and v, there must be a vertex w such that u and v are both adjacent to w.]

(b) Give an example to show that if the condition $\deg u + \deg v \geq n-1$ is replaced with $\deg u + \deg v \geq n-2$, then G is not necessarily connected.

(c) Prove: For a graph G with n vertices, if the degree of each vertex is at least $n/2$, then G is connected.

(d) Prove: If there are exactly two vertices with odd degrees in a graph, then they must be connected to each other (meaning, there is a path connecting these two vertices).

[*Hint:* Proof by contradiction.]

Solution:

(a) If u and v are two adjacent vertices, they are connected by definition. Then, consider non-adjacent u and v. Then, there must be a vertex w such that u and v are both adjacent to w. To see why, suppose this is not the case. Then, the set of neighbors of u and v has $n-1$ elements, but there are only $n-2$ other vertices. (This is the Pigeonhole Principle.) We have proven that for any non-adjacent u and v, there is a path $u \to w \to v$, and thus G is connected.

(b) Consider a graph with $n = 2m$ vertices which consists of two disconnected copies of K_m. For non-adjacent u, v, it holds that $\deg u + \deg v = (m-1) + (m-1) = 2m - 2 = n - 2$, but the graph is not connected. (Use K_2 and K_2 as an example for an easy visualization with 4 vertices.)

(c) Suppose that G is not connected. There must be at least two connected components, say G_1 and G_2. One of them will have at most $n/2$ vertices, and the maximum degree in this subgraph will be at most $n/2 - 1$. A contradiction.

Notice that part (a) directly implies the claim in this part. If each vertex's degree is at least $n/2$, then for any two non-adjacent vertices u, v,

$$\deg u + \deg v \geq \frac{n}{2} + \frac{n}{2} = n > n - 1.$$

Then by part (a), the graph is connected.

(d) Suppose that they are not connected to each other. Then they must belong to two different connected components, say G_1 and G_2. Each of them will only have one vertex with odd degree. This leads to a contradiction since the sum of all degrees should be an even number.

6 Euclid's Algorithm

(a) Use Euclid's algorithm from lecture to compute the greatest common divisor of 527 and 323. List the values of x and y of all recursive calls.

(b) Use extended Euclid's algorithm from lecture to compute the multiplicative inverse of 5 mod 27. List the values of x and y and the returned values of all recursive calls.

(c) Find $x \pmod{27}$ if $5x + 26 \equiv 3 \pmod{27}$. You can use the result computed in (b).

(d) Assume a, b, and c are integers and $c > 0$. Prove or disprove: If a has no multiplicative inverse mod c, then $ax \equiv b \pmod{c}$ has no solution.

Solution:

(a) The values of x and y of all recursive calls are (you can get full credits without the column of x mod y):

Function Calls	(x, y)	x mod y
#1	$(527, 323)$	204
#2	$(323, 204)$	119
#3	$(204, 119)$	85
#4	$(119, 85)$	34
#5	$(85, 34)$	17
#6	$(34, 17)$	0
#7	$(17, 0)$	—

Therefore, $\gcd(527,323) = 17$.

(b) To compute the multiplicative inverse of 5 mod 27, we first call `extended-gcd(27,5)`. Note that `(x div y)` in the pseudocode means $\lfloor x/y \rfloor$. The values of x and y of all recursive calls are (you can get full credits without the columns of x div y and x mod y):

Function Calls	(x,y)	x div y	x mod y
#1	$(27,5)$	5	2
#2	$(5,2)$	2	1
#3	$(2,1)$	2	0
#4	$(1,0)$	—	—

The returned values of all recursive calls are:

Function Calls	(d,a,b)	Returned Values
#4	—	$(1,1,0)$
#3	$(1,1,0)$	$(1,0,1)$
#2	$(1,0,1)$	$(1,1,-2)$
#1	$(1,1,-2)$	$(1,-2,11)$

Therefore, we get $1 = (-2) \times 27 + 11 \times 5$ and

$$1 = (-2) \times 27 + 11 \times 5 \equiv 11 \times 5 \pmod{27},$$

so the multiplicative inverse of 5 mod 27 is 11.

(c)

$$\begin{aligned}
5x + 26 \equiv 3 \pmod{27} &\Rightarrow 5x \equiv 3 - 26 \pmod{27} \\
&\Rightarrow 5x \equiv -23 \pmod{27} \\
&\Rightarrow 5x \equiv 4 \pmod{27} \\
&\Rightarrow 11 \times 5x \equiv 11 \times 4 \pmod{27} \\
&\Rightarrow x \equiv 44 \pmod{27} \\
&\Rightarrow x \equiv 17 \pmod{27}.
\end{aligned}$$

(d) False. We can have a counterexample: $a = 3$, $b = 6$, and $c = 12$, so a has no multiplicative inverse mod c (because $a = 3$ and $c = 12$ are not relatively prime). However, $3x \equiv 6 \pmod{12}$ has solutions $x = 2, 6, 10$ mod 12.