GRAPHS, BIJECTIONS, FLT, MODULAR ARITHMETIC 2

COMPUTER SCIENCE MENTORS 70

September 11 to September 15, 2017

1 Graph Theory

1.1 Introduction

1. Let $G = (V, E)$ be an undirected graph. Match the term with the definition.

   Walk Cycle Tour Path

   ______________Walk that starts and ends at the same node
   ______________Sequence of edges.
   ______________Sequences of edges with possibly repeated vertex or edge.
   ______________Sequence of edges that starts and ends on the same vertex and does not repeat vertices (except the first and last)

   Solution:
   tour Walk that starts and ends at the same node
   path Sequence of edges.
   walk Sequences of edges with possibly repeated vertex or edge.
   cycle Sequence of edges that starts and ends on the same vertex and does not repeat vertices (except the first and last)

2. What is a tournament?

Solution: A directed graph for which any pair of vertices, u, v either have an edge $u \to v$ or $v \to u$. It can be used to represent a tournament in which every pair of player play at least once.

3. What is a simple path?

Solution: Sequence of edges where the vertices are distinct

1.2 Questions

1. Given a graph G with n vertices, where n is even, prove that if every vertex has degree $\frac{n}{2}+1$, then G must contain a 3-cycle.

Solution: Let G be a graph with n vertices, where n is even, and every vertex has degree $\frac{n}{2}+1$. Select any two vertices u and v, with an edge between them. There are $n-2$ remaining vertices, and both u and v are connected to $\frac{n}{2}$ of these (because they have degree $\frac{n}{2}+1$ and are connected to each other). Therefore, there must be some vertex w such that both u and v are connected to w (otherwise the set of $\frac{n}{2}$ vertices connected to u and the set of $\frac{n}{2}$ vertices connected to v would be disjoint, which contradicts the fact that there are only $n-2$ of these vertices). Thus we have edges $(u, v), (v, u)$ and (w, u), so the graph contains a 3-cycle.

2. Every tournament has a Hamiltonian path. (Recall that a Hamiltonian path is a path that visits each vertex exactly once)

Solution: *Base Case*: For $n = 1$ nodes, there is a trivial Hamiltonian path.
Inductive Hypothesis: Assume that for a tournament with n nodes, there is a Hamiltonian path.
Inductive Step: Consider a tournament T with $n+1$ nodes. Take an arbitrary node x, and remove it along with its incident edges. The resulting subgraph T' is also a tournament (each node in T' still shares some edge with every other node in T'). By the Inductive Hypothesis, there is some Hamiltonian path in T'. Let this Hamiltonian Path be $v_1, v_2, v_3, \ldots, v_n$. Now we consider T. Note that since T is a tournament, x shares an edge with every other node in T. There are three possible cases:

Case 1: Everybody beat x (there is no edge from x to any node in T'). Then there is an edge (v_n, x). Thus, there is a Hamiltonian Path in T, namely

$v_1, v_2, v_3, \ldots, v_n, x$.

Case 2: x beat everybody (there is no edge from any node in T' to x). Then there is an edge (x, v_1). Thus, there is a Hamiltonian Path in T, namely $x, v_1, v_2, v_3, \ldots, v_n$.

Case 3: There is some v_i that is the last person who beat x, in the ordering $v_1, \ldots, v_n$. Note that v_i must exist because we are not in Case 2, and $i \neq n$ because we are not in Case 1. Then since v_i is the last person who beat x, there is an edge (v_i, x), and an edge (x, v_{i+1}). Thus, there is a Hamiltonian path in T, namely $v_1, v_2, v_3, \ldots, v_i, x, v_{i+1}, \ldots, v_n$. These are the only possible cases, so it must be that T has a Hamiltonian Path.

Therefore by induction, any tournament has a Hamiltonian Path.

2 Eulerian Tour

2.1 Introduction

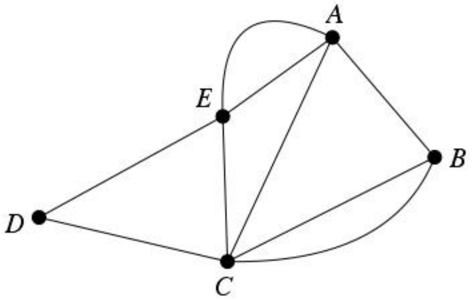
An **Eulerian path** is a path that uses every edge exactly once.

An **Eulerian tour** is a path that uses each edge exactly once and starts and ends at the same vertex.

Eulers Theorem: An undirected graph $G = (V, E)$ has an Eulerian tour if and only if G is even degree and connected (except possibly for isolated vertices).

2.2 Questions

1. Is there an Eulerian Tour? If so, find one. Repeat for an Eulerian Path.



Solution: There is no Eulerian Tour in the graph, because not all vertices have an even degree. An Eulerian Tour must visit every edge and end up at the same vertex. So the number of times it leaves/enters the start vertex must be even (every time it leaves, it must come back). Now every other vetex, must have the same condition. Since our tour doesn't end at any of these vertices, every time the tour enters a vertex, it must leave that vertex. Therefore, every vertex must have an even degree for there to be an Eulerian Tour.
There is an Eulerian Path. An Eulerian Path is almost like an Eulerian Cycle, without the condition that the start and end vertices must be the same. Therefore there are two vertices where the path can leave, and not return or enter and not leave. So there can be 2 vertices of odd degree. This graph does have 2 vertices of odd

degree.

$$B \to C \to D \to E \to A \to E \to C \to A \to B \to C$$

2. If every node has even degree except two nodes that have odd degree, prove that the graph has a Eulerian path.

Solution: First, add an edge from one odd degree vertex X to the other odd degree Y. This modified graph has an Eulerian Tour by Eulers Theorem. This tour contains something like $Z \to X \to Y \to \ldots \to Z$, where $\ldots \to Z$ contains every edge in the original graph. Deleting an edge leaves a Eulerian path in the graph.

3 Trees

3.1 Introduction

If complete graphs are maximally connected, then trees are the opposite: Removing just a single edge disconnects the graph! Formally, there are a number of equivalent definitions for identifying a graph $G = (V, E)$ as a tree.

Assume G is connected. There are 3 other properties we can use to define it as a tree.

1. G contains ________cycles.
2. G has ________edges.
3. Removing any additional edge will ______________

Solution: no, $n - 1$, disconnect G

One additional definition:

4. G is a tree if it has no cycles and ________________________

Solution: adding any edge creates a cycle

Theorem: G is connected and contains no cycles if and only if G is connected and has $n - 1$ edges.

3.2 Questions

1. Now show that if a graph satisfies either of these two properties then it must be a tree:

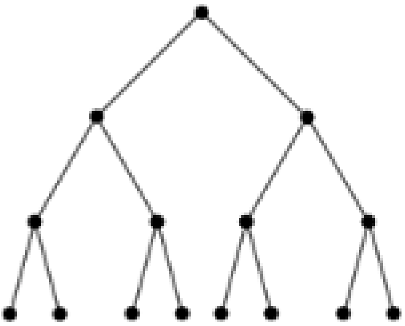
   a If for every pair of vertices in a graph they are connected by exactly one simple path, then the graph must be a tree.

   Solution: Assume we have a graph with the property that there is a unique simple path between every pair of vertices. We will show that the graph is a tree, namely, it is connected and acyclic. First, the graph is connected because every pair of vertices is connected by a path. Moreover, the graph is acyclic because there is a unique path between every pair of vertices. More explicitly, if the graph has a cycle, then for any two vertices x, y in the cycle there are at least two simple paths between them (obtained by going from x to y through the right or left half of the cycle), contradicting the uniqueness of the path. Therefore, we conclude the graph is a tree.

b If the graph has no simple cycles but has the property that the addition of any single edge (not already in the graph) will create a simple cycle, then the graph is a tree.

Solution: Assume we have a graph with no simple cycles, but adding any edge will create a simple cycle. We will show that the graph is a tree. We know the graph is acyclic because it has no simple cycles. To show the graph is connected, we prove that any pair of vertices x, y are connected by a path. We consider two cases: If (x, y) is an edge, then clearly there is a path from x to y. Otherwise, if (x, y) is not an edge, then by assumption, adding the edge (x, y) will create a simple cycle. This means there is a simple path from x to y obtained by removing the edge (x, y) from this cycle. Therefore, we conclude the graph is a tree.

2. Recall from the notes that a **rooted tree** is a tree with a particular node designated as the root, and the other nodes arranged in levels, growing down from the root. An alternative, recursive, definition of rooted tree is the following: A rooted tree consists of a single node, the root, together with zero or more branches, each of which is itself a rooted tree. The root of the larger tree is connected to the root of each branch.



Prove that given any tree, selecting any node to be the root produces a rooted tree according to the definition above.

Solution: Use induction! (on number of vertices)
Base case: one-vertex tree; have to select that to be the root; trivially a rooted tree (with zero branches)
Inductive hypothesis: tree with k or fewer vertices can be made into rooted tree by selecting any vertex as root
Inductive step: given a tree with $k+1$ vertices, let an arbitrary vertex v be selected as the root. This vertex v has, let us say, m neighbors.
Disconnecting each neighbor from k would produce m subtrees (which must be disjoint, or else we would have a cycle). By the inductive hypothesis, because each of these has at most k vertices, they each form a rooted tree when we select vs neighbor as the root. Overall, then, we have a root node, v, connected to a number of disjoint rooted trees, which are its branches.

3. A **spanning tree** of a graph G is a subgraph of G that contains all the vertices of G and is a tree.
Prove that a graph $G = (V, E)$ if connected if and only if it contains a spanning tree.

Solution: First the if direction. If a graph contains a spanning tree, which is a connected graph that contains all the vertices, there is a path between any two vertices, so the graph is connected.

Now the only if. Let G be a connected graph. Either G is already a tree, in which case it is its own spanning tree, or else there is an edge that can be removed from G while it remains connected. Because there are only a finite number of edges, we can continue this process until no more edges can be removed, at which point we have found our spanning tree.

4. Show that the edges of a complete graph on n vertices for even n can be partitioned into $\frac{n}{2}$ edge disjoint spanning trees.
   Hint: Recall that a complete graph is an undirected graph with an edge between every pair of vertices. The complete graph has $\frac{n*(n-1)}{2}$ edges. A spanning tree is a tree on all n vertices – so it has $n-1$ edges. So the complete graph has enough edges (for even n) to create exactly $\frac{n}{2}$ edge disjoint spanning trees (i.e. each edge participates in exactly one spanning tree). You have to show that this is always possible.

Solution: We proceed by induction.
Base Case: Consider a complete graph on 2 vertices. This can clearly be partitioned into 2/2 = 1 edge disjoint spanning tree, because the graph is already a tree.
Inductive Hypothesis: Assume that the edges of a complete graph on k vertices (for k even) can be partitioned into $\frac{k}{2}$ edge disjoint spanning trees.
Inductive Step: We need to partition the edges of a complete graph G_{k+2} on $k+2$ vertices into $\frac{k}{2}+1$ edge disjoint spanning trees.
To do this, label the vertices of G_{k+2} as $v_1, v_2, \ldots, v_{k+2}$. Remove the vertices $v_k + 1$ and v_{k+2} (and associated edges) to form a complete graph G_k with k vertices $v_1, \ldots, v_k$. By the inductive hypothesis, G_k has $\frac{k}{2}$ edge disjoint spanning trees; call these trees $T_1, \ldots, T_k = 2$. Add the vertices v_{k+1} and v_{k+2} back into G_k to once again form the graph G_{k+2}. These vertices come with $2k+1$ extra edges, connecting (v_i, v_{k+1}) and (v_i, v_{k+2}) for each $i = 1, 2, \ldots, k$, and also (v_{k+1}, v_{k+2}). These edges must be included into spanning trees. We wish to extend the trees $T_1, \ldots, T_{k+2}$ to include the new vertices v_{k+1} and v_{k+2}. To do this, for each tree T_i, attach two new edges (v_i, v_{k+1}) and $(v_{i+\frac{k}{2}}, v_{k+2})$. This extends each tree T_i to be a spanning tree. The remaining edges form one additional spanning tree. These edges are $(v_{i+\frac{k}{2}}, v_{k+1})$ and (v_i, v_{k+2}) for $i = 1to\frac{k}{2}$, along with the connecting edge (v_{k+1}, v_{k+2}). These edges connect each of the vertices v_{k+1} and v_{k+2} to half the remaining vertices, and together with the edge between v_{k+1} and v_{k+2} this gives the desired spanning tree. Therefore, we have covered the graph in $\frac{k}{2}+1$ edge disjoint spanning trees. This completes the induction.

Remark: The key idea here is the following:
Take a graph with k vertices that is partitioned into $\frac{k}{2}$ spanning trees. In the inductive step, we want to add two vertices (with associated edges). To maintain a partitioning into spanning trees, we must expand the preexisting $\frac{k}{2}$ trees to the new vertices, but this is a bit subtle! We need to add the two new vertices to each of the preexisting $\frac{k}{2}$ trees, which takes 2 $\frac{k}{2} = k$ edges connecting the preexisting k vertices to the two new vertices. Its really important that we use only one edge out of each of the original vertices! This is because otherwise, we would use up both new edges out of one of the vertices v_j, but then our final new spanning tree

wouldnt be able to reach v_j, so the remaining $k+1$ edges wouldnt be able to form a spanning tree!
So to do this, we need to split the original k vertices into two equal subsets of $\frac{k}{2}$ vertices each, and connect each half to one of the two new vertices. Once we do that, we can then justify forming a new spanning tree from the remaining edges, which allows us to complete the argument.

4 Hypercubes

4.1 Introduction

What is an n dimensional hypercube?

Bit definition: Two ____________ x and y are ____________ and only if _____ and _____ differ in __________________________ bit position.

Recursive definition: Define the 0-____________ as the $(n-1)$ dimensional ____________

with vertices labeled 0x (x is an element of _________________ (hint: how many remaining bits are there?). Do the same for the 1-____________ with vertices labeled ________. Then an n dimensional ____________ is created by placing an edge between ________ and ________ in the _______________ and _______________

respectively.

Solution: Bit definition: vertices, adjacent, x, y, exactly one

Recursive definition: subcube, hypercube, $(0,1)^{n-1}$, subcube, 1x, hypercube, 0x, 1x, 0-subcube, 1-subcube

4.2 Questions

1. How many vertices does an n dimensional hypercube have?

Solution: 2^n

2. How many edges does an n dimensional hypercube have?

Solution: $n * 2^{n-1}$

3. How many edges do you need to cut from a hypercube to isolate one vertex in an n-dimensional hypercube?

Solution: n because each node has n edges.

4. Prove that any cycle in an n-dimensional hypercube must have even length.

Solution: Answer: Here are three ways to solve this problem: here we will argue via bit flips, but there also exist arguments using the parity of Hamming distance, or induction on n. Note that induction on n is more difficult and prone to build-up error.

Answer 1: Bit flips

Main idea: moving through an edge in a hypercube flips exactly one bit, and moreover each bit must be flipped an even number of times to end up at the starting vertex of the cycle.

Proof: Each edge of the hypercube flips exactly one bit position. Let E_i be the set of edges in the cycle that flip bit i. Then $|E_i|$ must be even. This is because bit i must be restored to its original value as we traverse the cycle, which means that bit i must be flipped an even number of times. Since each edge of the cycle must be in exactly one set E_j, the total number of edges in the cycle = $\sum_j |E_j|$ is a sum of even numbers and therefore even.

5. **Coloring Hypercubes**
Let $G = (V, E)$ be an undirected graph. G is said to be k-vertex-colorable if it is possible to assign one of k colors to each vertex of G so that no two adjacent vertices receive the same color. G is k-edge-colorable if it is possible to assign one of k colors to each edge of G so that no two edges incident on the same vertex receive the same color.
Show that the n-dimensional hypercube is 2-vertex-colorable for every n.

Solution: Base case: For $n = 1$, the hypercube is a single edge. If we color one vertex red and the other blue we have a 2-vertex coloring, since the adjacent vertices are colored differently.

Inductive Step: Assume weve shown this to hold for n-dimensional hypercubes, we will show this holds for $n + 1$-dimensional hypercubes. Recall that we can define an $n + 1$ dimensional hypercube as two n-dimensional hypercubes where every vertex i in the first hypercube is connected to vertex i in the second hypercube. Considering this definition, for a given $n + 1$ dimensional hypercube, let H_0, H_1 denote the first and second n-dimensional hypercubes, respectively. By the inductive hypothesis, we assume that H_0 is 2-vertex colorable, and therefore there exists some coloring scheme which is a legal 2-vertex coloring of H_0. Given this coloring, we will color H_1 in the opposite coloring scheme which, given a color of vertex i in H_0 assigns the opposite color to the vertex i in H_1. Since we colored the vertices in both H_0 and H_1, we have colored all the vertices in the hypercube. It remains to show that this coloring scheme is legal. Assume, for purpose of contradiction that there is a given vertex i in H_1 which has an adjacent neighbor colored with the same color. If that neighbor is in H_1, then this means that the coloring of H_1 is not legal. Observe that if a coloring scheme is a legal 2-vertex coloring on some graph G, then the opposite coloring scheme is also a legal 2-vertex coloring on G. Since we colored H_1 with the opposite scheme of H_0, and H_0 is identical to H_1, this implies that the coloring of H_1 is a legal 2-vertex coloring, which contradicts having two adjacent vertices in H_1 sharing the same color. If the neighbor is in H_0, then we know, by definition of the $n + 1$-dimensional hypercube, that the neighbor must be i in H_0. In our coloring however, we colored$i$ in H_0 and i in H_1 in opposite colors, which again contradicts our assumption. Similarly, we can show for the case where i is in H_0

5 Extra Practice

5.1 Questions

1. Prove that every undirected finite graph where every vertex has degree of at least 2 has a cycle.

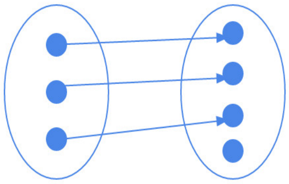
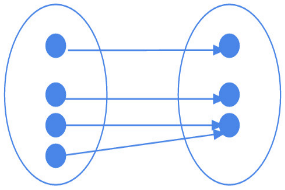
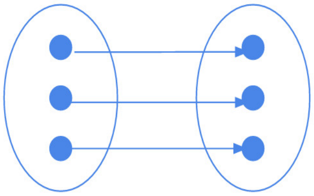
Solution: Assume that it does not have a cycle. So we have a graph G that is connected and does not have a cycle. It must be a tree. Every tree has at least one leaf. Leaves have degree 1. But we assumed that every vertex has degree at least 2. Contradiction, there must be a cycle.

2. Prove that every undirected finite graph where every vertex has degree of at least 3 has a cycle of even length.

Solution: Let P be a maximal path and v be an endpoint of P. v has at least 3 neighbors that are on P (why?) Let the path P be: $(\dots x, y, z, v)$ Now consider the following three paths: $v \to y$, $y \to x \to v$, and $y \to z \to v$ The union of two of these paths is a cycle. At least two of these paths must have lengths that are both even or both odd. Adding either of those cases together creates an even length cycle.

6 Bijections

1. Draw an example of each of the following situations

One to one AND NOT onto (injective but not surjective)	Onto AND NOT one to one (surjective but not injective)	One to one AND onto (bijection, i.e. injective AND surjective)
Solution: . 	**Solution:** . 	**Solution:** . 

2. Are the following functions **injections** from Z_{12} to Z_{24}?

a. $f(x) = 2x$

Solution: Yes: any two x_1 and x_2 will not equal each other as long as $x_1 \neq x_2$

b. $f(x) = 6x$

Solution: No: 0 and 4 both map to 0

c. $f(x) = 2x + 4$

Solution: Yes: same as 2x, except shifted

3. Are the following functions **surjections** from Z_{12} to Z_6? (Note: that $\lfloor x \rfloor$ is the floor operation on x)

   a. $f(x) = \lfloor \frac{x}{2} \rfloor$

   Solution: Yes: plug in every even number 0

   b. $f(x) = x$

   Solution: Yes: plug in 0 through 5

   c. $f(x) = \lfloor \frac{x}{4} \rfloor$

   Solution: No: the largest value we can get is f(12) which equals 3

4. Are the following functions **bijections** from Z_{12} to Z_{12}?

   a. $f(x) = 7x$

   Solution: Yes: the mapping works, since 7 is coprime to 12, so there exists a multiplicative inverse to 7 in Z_{12} ($7x7 = 49mod12 = 1,$ so $f^{-1}(x) = 7x$), which only occurs if the function is a bijection.

   b. $f(x) = 3x$

   Solution: No: $f(0) = f(4) = 0$.

   c. $f(x) = x - 6$

   Solution: Yes: can see its just $f(x) = x$, shifted by 6

7 Fermat's Little Theorem

7.1 Introduction

Fermat's Little Theorem: For any prime p and any $a \in \{1, 2, \ldots, p-1\}$, we have $a^{p-1} \equiv 1 \mod p$

1. Prove Fermat's Little Theorem.

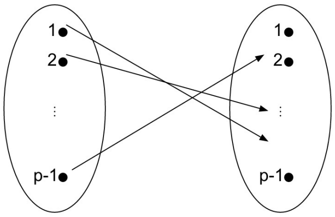
Solution: Proof from notes:
Claim: The function $a * x \mod p$ is a bijection where $x \in \{1, 2, \ldots, p-1\}$
The domain and range of the function are the same set, so it is enough to show that if $x \neq x'$ then $a * x \mod p \neq a * x' \mod p$.
Assume that $a * x \mod p \equiv a * x' \mod p$.
Since $gcd(a, p) = 1$, a must have an inverse: $a^{-1}(modp)$

$$ax \mod p \equiv ax \mod p$$

$$a^{-1} * a * x \mod p \equiv a^{-1} * a * x' \mod p$$

$$x \mod p \equiv x' \mod p$$

This contradicts our assumption that $x \neq x' \mod p$. Therefore f is a bijection. We want to use the above claim to show that $a^{p-1} \equiv 1 \mod p$. Note that now we have the following picture:



So if we multiply all elements in the domain together this should equal the product of all the elements in the image:

$$1 * 2 * \ldots * (p-1) \mod p \equiv (1a) * (2a) * \ldots * ((p-1)a) \mod p$$
$$(p-1)! \mod p \equiv a^{p-1} * (p-1)! \mod p$$
$$1 \equiv a^{p-1} \mod p$$

7.2 Questions

1. Find $3^{5000} \mod 11$

Solution:

$$(3^{10})^{500} \mod 11 = 1^{500} \mod 11 = 1$$

2. Show that $n^7 - n$ is divisible by 42 for any integer n

Solution: $42 = 7 * 3 * 2 \leftarrow$these factors are prime so lets apply FLT!!

$$n^7 \equiv n \mod 7$$
$$n^3 \equiv n \mod 3$$
$$n^2 \equiv n \mod 2$$

Were interested in n^7 so lets modify the bottom two equations to write n^7 in mod 3 and mod 2

$$n^7 \equiv n^3 * n^3 * n \equiv n * n * n \equiv n^3 \equiv n \mod 3$$
$$n^7 \equiv n \mod 3$$

$$n^7 \equiv n^2 * n^2 * n^2 * n \equiv n * n * n * n \equiv n^2 * n^2 \equiv n * n \equiv n^2 \equiv n \mod 2$$
$$n^7 \equiv n \mod 2$$

$$n^7 \equiv n \mod 7$$
$$n^7 \equiv n \mod 3$$
$$n^7 \equiv n \mod 2$$

Wouldnt it be great if the above equations implied that $n7 \equiv n \mod 7 * 3 * 2$?
Lets try to prove that.
Claim: If

$$x \equiv y \mod a_1$$
$$x \equiv y \mod a_2$$
$$\dots$$
$$x \equiv y \mod a_n$$

are true and $a_1, \ldots, a_n$ are coprime then $x \equiv y \mod a_1 a_2 \ldots a_n)$
$x \equiv y \mod a_i \rightarrow x = y + c_i * a_i$ for some constant c_i

$$x = y + c_1 * a_1$$

$$x = y + c_2 * a_2$$

$$\ldots$$

$$x = y + c_n * a_n$$

But this implies that $x = c * lcm(a_1, \ldots, a_n) + y$
Since $a_1, \ldots, a_n$ are coprime, $lcm(a_1, \ldots, a_n) = a_1 * a_2 * \ldots * a_n$
So we get $x = c * a_1 * a_2 * \ldots * a_n + y$
Therefore $x \equiv y \mod a_1 * a_2 * \ldots * a_n$
We can now say that $n^7 \equiv n \mod 7 * 3 * 2 \equiv n \mod 42$.