

Student Information

Full Name : Mithat Can Timurcan

Id Number : 2581064

Answer 1

- Let's look at our graph G . Since for every vertex, there's a path to all vertices, we can see that the graph is connected. Let's also write the degrees of all vertices:

$$\deg(a) = \deg(b) = \deg(f) = \deg(k) = \deg(m) = 2$$

$$\deg(c) = \deg(d) = \deg(e) = \deg(g) = \deg(h) = \deg(i) = \deg(j) = \deg(l) = 4$$

a)

- By the Theorem 1 on the Section 10.5 of the textbook, since G is connected and all of its vertices have even degrees, we can say that there exists an Eulerian circuit.
- A sample circuit can be written as follows: $a - c - b - g - k - l - h - g - c - d - h - i - l - m - j - i - d - e - f - j - e - a$.

b)

- By the Theorem 2 on the Section 10.5 of the textbook, since our graph G is connected, we only need to have 2 odd degree vertices in order to have an Eulerian path which is not a circuit.
- Since all of our vertices have even degrees, we can see that there is no Eulerian path which is not a circuit.

c)

- A Hamilton Circuit is a circuit that traverses all vertexes (except the starting vertex) once. Although there's no easy way to conclude that there is no Hamilton circuit in our graph, according to the textbook (last paragraph of page 699), we can check 3 properties:
 - A graph with a vertex of degree one cannot have a Hamilton circuit, because in a Hamilton circuit, each vertex is incident with two edges in the circuit.
 - If a vertex in the graph has degree two, then both edges that are incident with this vertex must be part of any Hamilton circuit.
 - A Hamilton circuit cannot contain a smaller circuit within it.

- However in our graph, the vertexes a, b, f, k, m have degree of two. For the second property, the edges incident with those vertices must be included in our Hamilton circuit. When we consider those edges, we have a smaller circuit. We get a contradiction with the 3rd rule. Therefore, we can say that there exists no Hamilton circuit in our graph G .

d)

- A Hamilton Path is a path that traverses all vertexes once. According to Ore's theorem, a simple graph has a Hamiltonian Path if for every pair of vertices in G sum of their degrees is greater than $n - 1$ where $n = |V|$. In our case, $n = 13$. There's no such pair that satisfy this condition. So, we can't conclude anything from Ore's Theorem. However, we can try to find a path ourselves. Such path can be: $a - c - b - g - k - l - m - j - f - e - d - h - i$.

e)

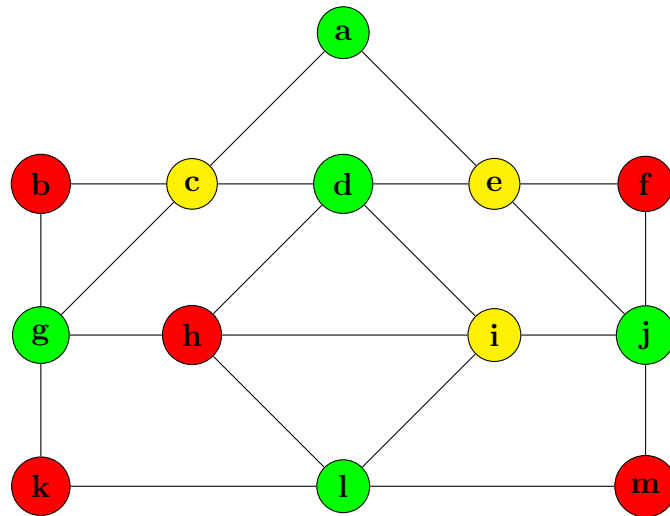


Figure 1: Coloring of the graph G .

- From Figure 1. we can say that the chromatic number of this graph is $\chi(G) = 3$.
- In 2-colorable graphs, there is no simple cycle such has an odd number of edges. A sample cycle $a - c - g - h - i - j - e - a$ has 7 edges, leading us to use at least 3 colors to color those vertices.

f)

- Since the chromatic number of our graph G is $\chi(G) = 3 \neq 2$, we can say that it's not bipartite.
- We need to delete the edges forming an odd numbered cycle. Such edges will be $\{b, g\}$, $\{f, j\}$ and $\{h, i\}$.

- Now let's check our graph again, if it's 2-colorable, then we can say that it's bipartite:

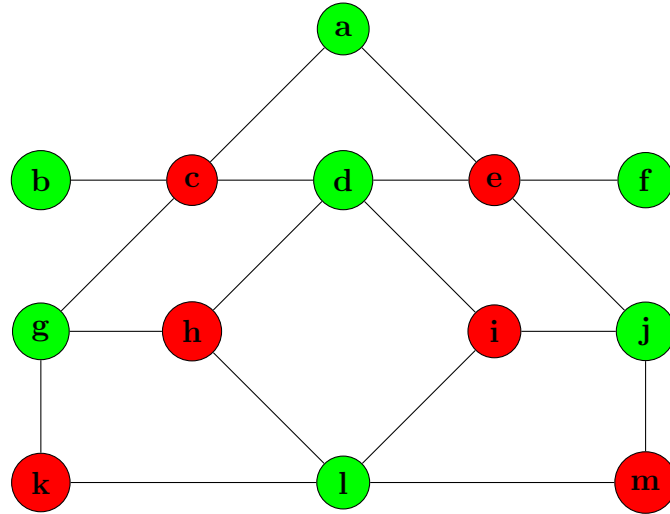


Figure 2: New coloring of the graph G .

- Since we can now color it using only 2 colors, we can say that it's bipartite and therefore, the minimum number of edges that should be removed to make this graph bipartite is 3.

g)

- No, there isn't any subgraphs that contain the complete graph with 4 vertices since our graph's chromatic number is $\chi(G) = 3$. Complete graphs are always n -colorable where n denotes the number of vertices because all vertices are connected to each other.
- We can add an edge between the vertices d and l to obtain a subgraph that is complete.

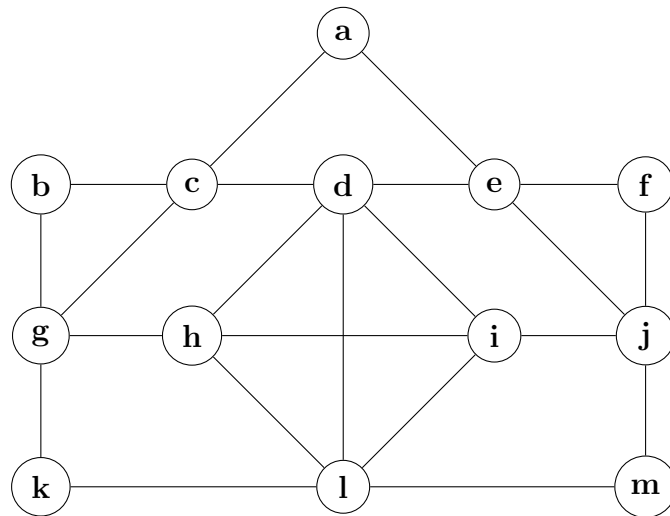


Figure 3: New graph G' which has a complete graph with 4 vertices as a subgraph.

Answer 2

- First of all, we need to check the graph invariants. If they're the same, we can continue.
 - **Number of vertices:** 8 for graph G , 8 for graph H . They're the same.
 - **Number of edges:** 16 for graph G , 16 for graph H . They're the same.
 - **Degrees of vertices:** Every vertex in graph G has the degree of 4, every vertex in graph H also has the degree 4. They're the same.
- Since the graph invariants are the same, now we need to find a bijective isomorphism function to map the graph G 's vertexes to graph H 's vertexes. From now on, let's denote this function as f .
- We should examine the cycles and try to preserve the edges, by doing so we get the following isomorphism function:

$$f(a) = a', f(b) = c', f(c) = e', f(d) = g'$$

$$f(e) = b', f(f) = h', f(g) = d', f(h) = f'$$

- We now have a bijection between the vertex sets of G and H . Let's check the adjacency matrixes of G and H :

$$\begin{array}{c}
 \begin{matrix} & a & b & c & d & e & f & g & h \end{matrix} \\
 \begin{matrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{matrix} \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}
 \end{array}$$

Figure 4: Adjacency matrix representation for graph G , M_G .

$$\begin{array}{c}
\begin{array}{cccccccc}
& a' & c' & e' & g' & b' & h' & d' & f'
\end{array} \\
\begin{array}{c}
a' \\
c' \\
e' \\
g' \\
b' \\
h' \\
d' \\
f'
\end{array}
\begin{bmatrix}
0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 0
\end{bmatrix}
\end{array}$$

Figure 5: Adjacency matrix representation for graph H , M_H .

- Since $M_G = M_H$, we can see that our isomorphism function f , preserves edges. Therefore, we've shown that graphs G and H are isomorphic.

Answer 3

a)

- For odd number of vertices ($n = 3, 5, 7, \dots$): The chromatic number is 3. This is because we can't color an odd cycle with just 2 colors without causing adjacent vertices to share the same color. For example, in a triangle, alternating colors will result in the first and last vertex (which are adjacent) being the same color if only two colors are used.
- For even number of vertices ($n = 4, 6, \dots$): The chromatic number is 2. This is because we can actually alternate between two colors around the cycle and return to the starting vertex with a different color than the one we started with.
- If a graph is 2-colorable, then we can say that it's bipartite. Therefore, we can say that for even number of vertices C_n is bipartite. However, for odd number of vertices, C_n is not bipartite.

b)

- For the n -dimensional cube graph Q_n , the graph will always be bipartite. We can show this by induction.
- **Base Case** ($n = 1$): Q_1 is a graph with two vertices connected by just one edge, we can clearly see that it's bipartite. We can set a partitioning such that no vertexes at those partitions connect each other.
- **Inductive Step** ($n \geq 1$):
 - Assume that for some integer $k \geq 1$, Q_k is bipartite. We need to show that Q_{k+1} is also bipartite.

- We can form Q_{k+1} by taking two copies of Q_k and adding edges between corresponding vertices in these copies. In each copy, we're labeling the vertices such that they differ only by the $(k+1)$ th bit.
 - In Q_k , we can divide vertices into two sets based on the parity of the sum of their bits (even sum and odd sum) since Q_k is bipartite by the inductive hypothesis.
 - In Q_{k+1} , this property is preserved. One copy of Q_k will have all vertices with an even sum of bits, and the other with an odd sum, considering the $(k+1)$ th bit.
 - Therefore, each edge in Q_{k+1} either connects vertices with even bit sums to those with odd sums (within the same Q_k copy) or connects corresponding vertices in different copies, which also have different parity sums.
 - This shows that Q_{k+1} is bipartite, completing our proof.
- Since we've shown that cube graph Q_n is always bipartite, we can say that it's 2-colorable and the graph chromatic number is equal to 2. $\chi(Q_n) = 2$.

Answer 4

a)

- We can use Kruskal's Algorithm to form a minimum spanning tree (MST).
- Let's write the steps of the algorithm:
 - We are going to sort all the edges in non-decreasing order of their weights.
 - We will pick the smallest edge (in terms of their weight) and check if it forms a simple circuit with the spanning tree formed so far. If it doesn't form a simple circuit, we can include this edge. Otherwise, we won't include that edge.
 - We will repeat the 2nd step until there are $(V - 1)$ edges in the spanning tree.

Choice	Edge	Weight
1	$\{a, b\}$	1
2	$\{c, e\}$	2
3	$\{c, f\}$	2
4	$\{a, d\}$	3
5	$\{b, c\}$	4

b)

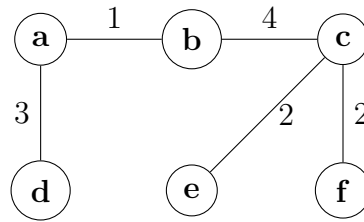


Figure 6: Minimum Spanning Tree with Kruskal's algorithm for the given graph G

c)

- No, the minimum spanning tree we found is not the only one we can form. When we're picking the 2nd and 3rd edges while using Kruskal's Algorithm, we have more than one option. Such spanning trees can be also formed as follows:

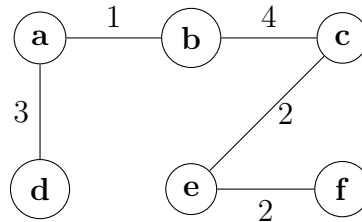


Figure 7: 2nd Minimum Spanning Tree with Kruskal's algorithm for the given graph G

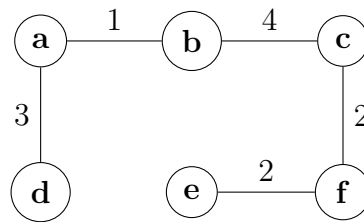


Figure 8: 3rd Minimum Spanning Tree with Kruskal's algorithm for the given graph G

Answer 5

a)

- We can use mathematical induction on the tree's height h to show that a full binary tree with n vertices has $\frac{n+1}{2}$ leaf vertices.
- **Base Case** ($h = 0$):

- When the tree's height is 0, it only has 1 vertex which is also the leaf vertex. Since $\frac{1+1}{2} = 1$ leaf vertex, the property is satisfied when the tree's height is 0.

• **Inductive Step ($h \geq 0$):**

- Assume that for a full binary tree of height h with n vertices, the property is satisfied. By the inductive hypothesis, the tree has $\frac{n+1}{2}$ leaf vertices.
- We need to prove the statement for the height $h + 1$. In order to create a tree with height $h + 1$, we add two children to one of the leaf vertices. Now we have $n + 2$ total vertices.
- The original leaf vertex is now an branch vertex, and the new leaves are the newly added children. The original leaf adds 2 new leaves but we need to subtract the original leaf, so the number of leaves in the tree of height $h + 1$ is $\frac{n+1}{2} + 2 - 1 = \frac{n+1}{2} + 1$. Let's calculate the number of leaf vertices with our formula, using $n + 2$ vertices:

$$\frac{(n+2)+1}{2} = \frac{n+1}{2} + 1 \text{ leaf vertices}$$

- Since we got the same result, we can say that the property is also satisfied when the height is $h + 1$.
- Therefore, we've shown that a full binary tree with n vertices has $\frac{n+1}{2}$ leaf vertices.

b)

- Let's denote the tree with n vertices as T_n . The chromatic number of a tree will always be equal to 2. $\chi(T_n) = 2$.
- This property holds for every integer n since trees are acyclic (containing no cycles) and connected graphs. We can always start from some vertex and color it with a specific color. Then, as we move away from that vertex along the edges, we will always alternate between the two colors for each level of the tree.
- Trees are also always bipartite graphs, meaning that we can set two partitions such that no vertices are not connected with each other in each partition. We can color the vertices in one partition same color, and for the other partition we can use another color, leading us having only 2 colors to color a tree.

c)

- To find the upper bound of a tree, we need to achieve the maximum height with the minimum number of nodes. In order to achieve this, we can add m nodes (since the tree will be full) to one child at each level while incrementing the height of the tree. We get the following tree:

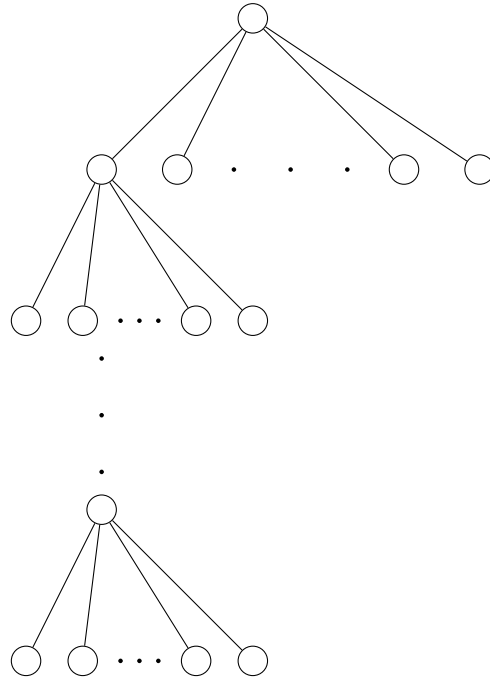


Figure 9: Full m -ary tree with n nodes and height h .

- Every level except the first level has m nodes. Calculating the total number of nodes, we get $mh + 1 = n$. This equation is the minimum number of nodes to form a full m -ary tree with height h .

$$mh + 1 \leq n$$

$$h \leq \frac{n-1}{m}$$

- Therefore we can say that $\frac{n-1}{m}$ is the upper bound for height of a full m -ary tree.