Student Information

Full Name : AYŞEGÜL ERDEM
Id Number : 2633196

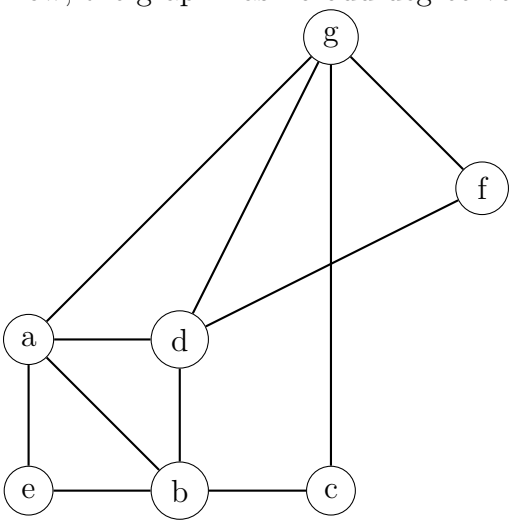
Answer 1

a)

YES, to have an Euler Path the graph should have 0 or 2 Vertices that has odd degree. In this graph, $deg(a) = 3$ and $deg(g) = 3$, so graph have 2 odd degree vertices which means that it is still satisfying the condition. Moreover, the path is:
$a, e, b, d, a, b, c, g, d, f, g$.

b)

NO,To have an Euler Circuit graph the graph cannot have any odd degree vertices, but as mentioned above it has two (a and g). Furthermore, there is no Euler Circuit for this graph.
However,since a and g are the ones that has odd degree, I can create an Euler Circuit for this graph by adding a new edge between the vertices a and g. After this new connection, new changed degrees compared to previous will become:
$deg(a) = 4$ (since I added a new edge)
$deg(g) = 4$ (since I added a new edge)
Now, the graph has no odd degree vertices, and satisfies the condition



Hence new obtained circuit is:
a,b,e,a,d,b,c,g,f,d,g,a

c)

YES, the graph can be traveled by passing vertices just once. Hamilton Path is: d, f, g, c, b, e, a

d)

YES, a circuit can be created by using vertices just once. The Hamilton Circuit is:
a, e, b, c, g, f, d, a

Answer 2

First of all, we should check whether that is it possible for these graphs to be isomorphic. Since they have the same number of vertices and all vertices G and H has the degree 4. It can be possible to define a one-to-one and onto function for these graphs such that:

$f(g_i) = h_k$ (g_i and g_j for the vertices of G and h_k and h_l for the vertices of H)

Now, since we did the general observations we can analyze graphs in more detailed. Let's use matrix representation for our analysis.
If we can obtain the same matrices from G and H it means that:

$$\{g_i, g_j\} \in E_1 \iff \{f(g_i), f(g_j)\} \in E_2$$

where $G = (V_1, E_1)$ and $H = (V_2, E_2)$

Adjacency Matrix of G:

	a	b	c	d	e	f	g	h
a	0	1	0	1	1	1	0	0
b	1	0	1	0	1	0	1	0
c	0	1	0	1	0	0	1	1
d	1	0	1	0	0	1	0	1
e	1	1	0	0	0	1	0	1
f	1	0	0	1	1	0	1	0
g	0	1	1	0	0	1	0	1
h	0	0	1	1	1	0	1	0

Adjacency Matrix of H:

	z	u	s	y	t	x	v	r
z	0	1	0	1	1	1	0	0
u	1	0	1	0	1	0	1	0
s	0	1	0	1	0	0	1	1
y	1	0	1	0	0	1	0	1
t	1	1	0	0	0	1	0	1
x	1	0	0	1	1	0	1	0
v	0	1	1	0	0	1	0	1
r	0	0	1	1	1	0	1	0

Since Adjacency Matrix of G is equal to Adjacency Matrix of H. We can conclude that f is a function such that:
$f(a) = z$, $f(b) = u$, $f(c) = s$, $f(d) = y$, $f(e) = t$, $f(f) = x$, $f(g) = v$, and $f(h) = r$. Moreover, since it prevents the relations it is also a isomorphism. G and H are isomorphic.

Answer 3

a)

Imagine a bank with three employees and five customers, where each employee has specific skills or responsibilities, and each customer requires a unique service. Determine the possible matchings between employees and customers based on capabilities of the employees.

Details of the Employees' Skills

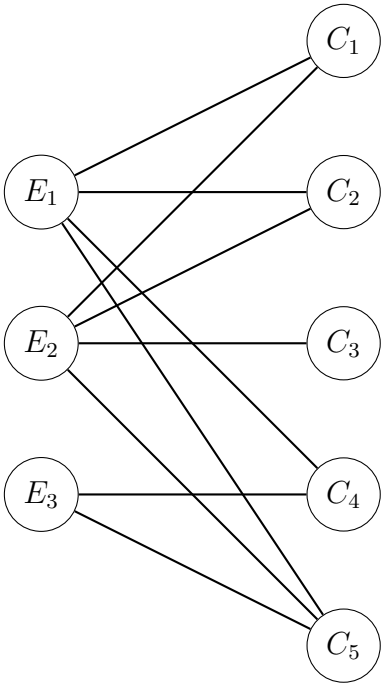
Employee 1:
Money Withdrawal
Credit Card Application
Account Creation
Money Transfer

Employee 2:
Money Withdrawal
Credit Card Application
Money Transfer
Loan Application

Employee 3:
Account Creation
Money Transfer

Details of the Customers' Needs:

Customer 1: Requires Money Withdrawal.
Customer 2: Needs assistance with a Credit Card Application.
Customer 3: Wants to apply for a Loan.
Customer 4: Needs help with Account Creation.
Customer 5: Requires a Money Transfer.



b)

A bipartite graph is a graph whose vertices can be divided into two disjoint sets, and these sets does not include adjacent vertices.
Let's start with creating two disjoint sets:

Assume G=(V,E) (G is a graph, V is the set of vertices and E is the set of edges)

v and e will be used for representing a vertex (v), and an edge (e) that are the elements of V and E.
V_1 and V_2 will be used for representing 2 disjoint subsets.

$V_1 \cap V_2 = \emptyset$

means that for all v that is an element of V , v is an element of V_1 or an element of V_2.

$\forall\ v \in V\ (V_1 \cap V_2 = \emptyset) \wedge (\text{v}\in V_1 \vee \text{v}\in V_2)$

Now, we should define edges:
An edge of a bipartite graph should connect the vertices that is in the different disjoint subsets. This means that if there exists an edge between v_1 and v_2, then v_1 and v_2 cannot be in the same subset.:

$\forall$ e($\{v_1,v_2\}$) $\in E$ $(((v_1 \in V_1) \wedge (v_2 \in V_2\ )) \vee ((v_1 \in V_2) \wedge (v_2 \in V_1))) \wedge \neg\ (((v_1 \in V_1) \wedge (v_2 \in V_1\ )) \vee ((v_1 \in V_2) \wedge (v_2 \in V_2)))$

Here is the resulting definition:

$\forall$ e($\{v_1,v_2\}$) $\in E$ $(((v_1 \in V_1) \wedge (v_2 \in V_2\ )) \vee ((v_1 \in V_2) \wedge (v_2 \in V_1))) \wedge \neg\ (((v_1 \in V_1) \wedge (v_2 \in V_1\ )) \vee ((v_1 \in V_2) \wedge (v_2 \in V_2))) \wedge \forall\ v \in V\ ((V_1 \cap V_2 = \emptyset) \wedge (\text{v}\in V_1 \vee \text{v}\in V_2))$

c)

Assume G=(V,E) (G is a graph, V is the set of vertices and E is the set of edges)
v will be used for representing a vertex (v) which is an element of V

$\forall\ v_i,v_j \in V\ (deg(v_i) = deg(v_j))$

(v_i and v_j is used for representing arbitrary vertices)

Let's assume this common degree is m:

$\forall\ v \in V\ (deg(v) = \text{m}\ )$

d)

Assume that a regular graph which has more than two vertices, and it is a tree at the same time:
G(V,E) (V is representing the set of vertices(v) and E is representing set of edges(e))
A regular graph means that degrees of all vertices are equal.
Let's take this common degree m:

$\forall\ v \in V\ (deg(v) = \text{m}\ )$

And if a regular graph has more than two vertices, m should be greater than or equal to 2 to satisfy the condition that all degrees are equal.

$\text{m} \geq 2$

A tree is a connected graph. It should have leaf vertices (degree 1) if it has more than two vertices.

Conclusion:
We simply observed that if a regular graph has more than two vertices:

m $\geq$ 2 for all vertices.

However, if we try to create a tree which is satisfying the degree condition, it cannot be possible because for a tree there should be at least one vertex with a degree 1 (leaf vertex), if it has more than two vertices. The basic properties of three and the degree condition are contradicting
Hence, proof is completed by proof by contradiction. It cannot be tree

e)

G=(V,E) Let's G be a graph which is a regular bipartite graph.
From our assumptions G should have some properties such that:
It should be divided into two disjoint subsets (V_1,V_2) and every edge of G should connect the vertices from different subsets. It should not contain any edge between the vertices within same subset (V_1 or V_2)

$V_1 \cap V_2 = \emptyset$

All degrees of vertices should be equal (take m). This also means that every vertex is connected to exactly m edges.

Let's start observations about the sizes od disjoint subsets

$|V_1| = n_1$ (number of vertices in this subset)
$|V_2| = n_1$ (number of vertices in this subset)

The number of edges (N) between two subsets is equal to total number of edges and it can be represented by using 2 different perspective:

Number of edges from V_1 to V_2 :
Since all n_1 vertices have m edges

$X = n_1 \cdot m$

Number of edges from V_2 to V_1 :
Since all n_2 vertices have m edges

$Y = n_2 \cdot m$

We already stated that both X and Y are equal to the total number of edges

$N = X = Y$

which means that two perspective is equal to each other

$X = Y$

$n_1 \cdot m = n_2 \cdot m$

and it can be concluded from this equation

$n_1 = n_2$

The two disjoint subsets have the same size when the bipartite graph is regular.

Answer 4

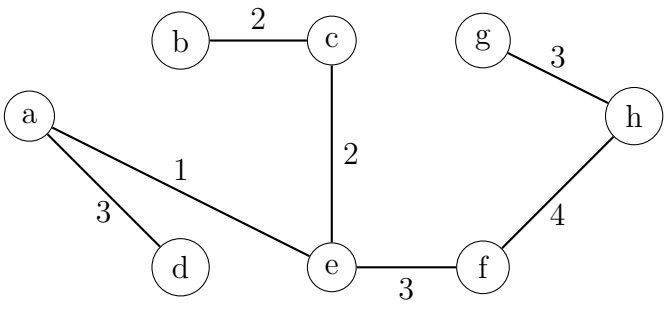
a)

To apply Prim's Algorithm to find MST, we should just start by choosing minimum edge of visited vertex (a is mentioned in question as starting vertex). Moreover, we will move on applying basically the same thing(choosing minimum edges) by avoiding creating circuits.

Edge	**Weight**	**Order**
$\{a, e\}$	1	1
$\{e, c\}$	2	2
$\{c, b\}$	2	3
$\{e, f\}$	3	4
$\{a, d\}$	3	5
$\{f, h\}$	4	6
$\{h, g\}$	3	7

$Table1$

Also, we should stop when all vertices are connected to MST.
From the Table1 the edges are added to the MST in this order:
{a,e},{e,c},{c,b},{e,f},{a,d},{f,h},{h,g}



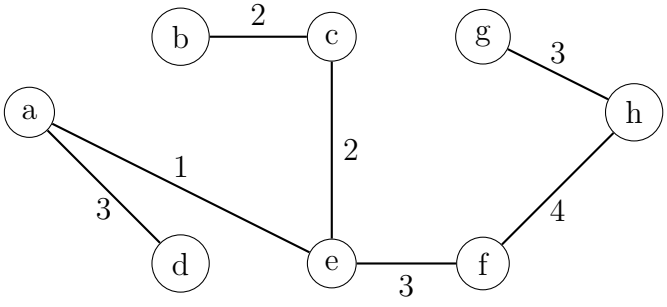
b)

To apply Kruskal's Algorithm, we should choose minimum weighted edge and continue to do this until all vertices be connected to the graph, and we shoul avoid creating circuit.

Edge	**Weight**	**Order**
$\{a, e\}$	1	1
$\{e, c\}$	2	2
$\{c, b\}$	2	3
$\{e, f\}$	3	4
$\{g, h\}$	3	5
$\{a, d\}$	3	6
$\{f, h\}$	4	7

*Table*2

From the Table1 the edges are added to the MST in this order:
{a,e},{e,c},{c,b},{e,f},{g,h},{a,d},{f,h}.



c)

Step	**Visited**	**a**	**b**	**c**	**d**	**e**	**f**	**g**	**h**
0	–	0	∞	∞	∞	∞	∞	∞	∞
1	a	0	7_a	∞	3_a	1_a	∞	∞	∞
2	e	0	4_e	3_e	3_a	1_a	4_e	∞	∞
3	d	0	4_e	3_e	3_a	1_a	4_e	∞	∞
4	c	0	4_e	3_e	3_a	1_a	4_e	8_c	∞
5	b	0	4_e	3_e	3_a	1_a	4_e	8_c	∞
6	f	0	4_e	3_e	3_a	1_a	4_e	8_c	8_f
7	g	0	4_e	3_e	3_a	1_a	4_e	8_c	8_f
8	h	0	4_e	3_e	3_a	1_a	4_e	8_c	8_f

Step 0: Since a is started point initialize the distance a as 0. the others are ∞ before starting the travel.
Step 1: Find the closest vertices to a (directly connected) and update their distances. Choose closest one which is e.
Step 2: Since the chosen one is e analyze its neighbors and update their distances if necessary (if the distances are smaller than the existing distances). Updates ones are b,c, and f. Choose closest vertex which is d.
Step 3: Look the neighbors of d and update their distances if necessary. There is no update there. Choose the vertex with smallest distance.
Step 4: Next selected one is c. Analyze its neighbors and update the distance of g from ∞ to 8.
Step 5: Next one is b, but there is no update here.
Step 6: Next one is f update the distance of h from ∞ to 8.
Step 7: Next chosen one is g, but no update there. Step 8: Next one is h again no update there.

Now, we reached the target vertex by using the a,e,f,h path from a to h. This path has the length 8. And this is the smallest path from a to h as we continuously updated distances if we found a smallest one.
Since there is no another way that has same weight from a to h, obtained path is unique. Because we did not do any arbitrary choice through applying this algorithm. If we did make some arbitrary choice, this can mean that there are some same weighted edges that can leads as the same target. However, although graph includes some same weighted edges, these could not affect the path.