Take Home Exam 5

Student Information

Full Name : Gürhan İlhan Adıgüzel

Id Number : 2448025

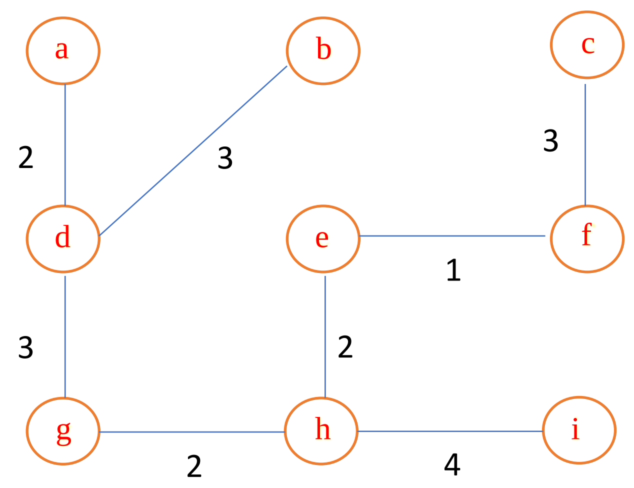
Question 1

Prim's algorithm is choosed for to find a minimum spanning tree for the graph G.

a)

Choice	Edge	Weight
1	{ e, f }	1
2	{ e, h }	2
3	{ h, g }	2
4	{ f, c }	3
5	{ g, d }	3
6	{ d, a }	2
7	{ d, b }	3
8	{ h, i }	4
		Total: 20

b)



c) We observe the minimum spanning tree of G is in the 1.b part. We need to prove that the weight of edges in G that are not in the minimum spanning tree exceeds the weight of every other edge on the cycle produced by adding those edges to the graph to demonstrate that the minimum spanning tree is unique.
When we look at the weight of the edges [(a,b), (b,c), (b,f), (d,e), (f,h), (i,f)] , we can see that these weights are exceed the other edges in the cycle that are formed when they added.
Therefore, the minimum spanning tree is unique for the graph G.
However, for any connected edge-weighted undirected graph, the minimum spanning tree is not unique. If we add an edge with the same weight as any of the edges on the cycles, implying that the added edge does not exceed the weight of other edges, the minimum spanning tree is not unique since we might choose that edge instead of the one we chose while generating the minimum spanning tree.

d) If the minimum edge was not included in the minimum spanning tree, constructing a cycle by adding e and removing any of the edges in that cycle with greater weight would result in a tree with a reduced weight. This demonstrates that the preceding tree was not a minimum spanning tree, implying that the minimum weight edge should be included in the minimum spanning tree.

Question 2

The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there is a one-to-one and onto function f from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if f(a) and f(b) are adjacent in G_2 , for all a and b in V1 . Such a function f is called an isomorphism.

Both graphs are 6 vertices and 8 edges.

They also both have 3 vertices of degree two, 2 vertices of degree three, 1 vertice of degree four.

We define an injection *f* from the vertices of *G* to the vertices of *H* that preserves the degree of vertices. We will determine whether it is an isomorphism.

The function *f* with

$f(a) = m$, $f(b) = q$, $f(c) = p$, $f(d) = r$, $f(e) = n$, and $f(f) = o$

is a one-to-one correspondence between *G* and *H*.

We can analyze the adjacency matrix of G and H to determine whether if f preserves edges

$$A_G = \begin{array}{c|c} & \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \end{array} & \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \qquad A_H = \begin{array}{c|c} & \begin{array}{cccccc} m & q & p & r & n & o \end{array} \\ \begin{array}{c} m \\ q \\ p \\ r \\ n \\ o \end{array} & \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

Because $A_G = A_H$ f preserves the edges, *f* is an isomorphism, and it follows that *G* and *H* are isomorphic graphs.

Question 3

a) The number of the vertices : 7

The number of the edges : 6

The height of T : 3

b) Postorder : q – s – u – v – t – r – p

Preorder : p – q – r – s – t – u – v

Inorder : q – p – s – r – u – t – v

c) T is a full binary tree because each of its internal vertices has two children.

d) A complete binary tree of height h is a binary tree that is full down to level h, with level h+1 filled in from left to right.

So, T is not a complete binary tree because levels are not filled in from left to right.

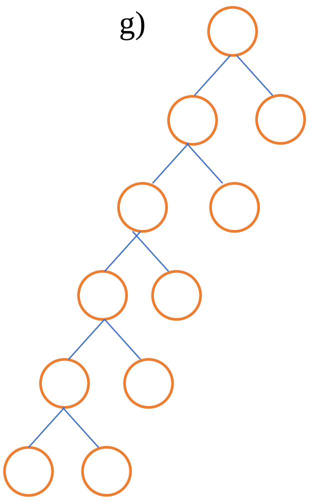
e) A rooted binary tree of height h is balanced if all leaves are at levels h or h − 1.
The leaves level of T are :
(q:13) is a leaf at level 1, (s:19) is a leaf at level 2, (u:23) and (v:58) are the leaves at level 3.

So, T is not a balanced binary tree because the level differences between nodes is more than 1.

f) In a binary search tree, the value of left node must be smaller than the parent node, and the value of right node must be greater than the parent node. This rule is applied recursively to the left and right subtrees of the root.

Although (u:23) has a smaller value than the node (r:24), (u:23) is in the right subtree of the node (r:24). So , T is not a binary search tree.

g)



This is the full binary tree with height 5 with the minimum number of nodes. So, there are 11 nodes in this full binary tree.