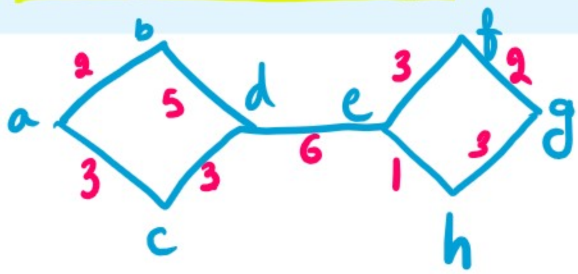
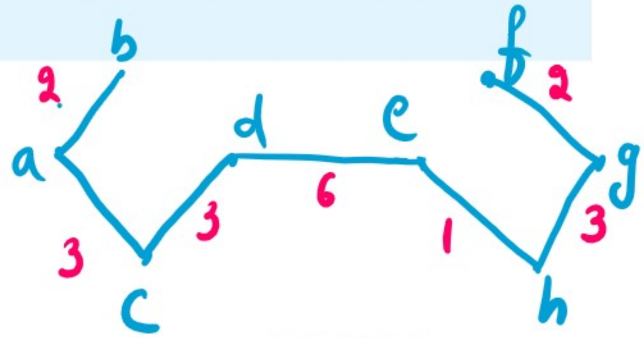
Lecture 36

26 November 2021 08:56

Minimum Spanning Tree

A *minimum spanning tree* in a connected weighted graph is a spanning tree that has the smallest possible sum of weights of its edges.



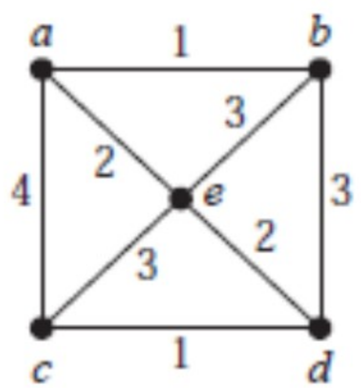


MST

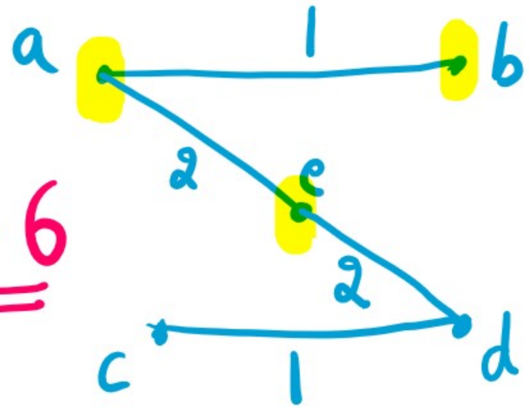
Algorithms to find Minimum spanning tree

ALGORITHM 1 Prim's Algorithm.

procedure *Prim*(G: weighted connected undirected graph with n vertices)
T := a minimum-weight edge
for i := 1 **to** $n - 2$
  e := an edge of minimum weight incident to a vertex in T and not forming a simple circuit in T if added to T
  T := T with e added
return T {T is a minimum spanning tree of G}

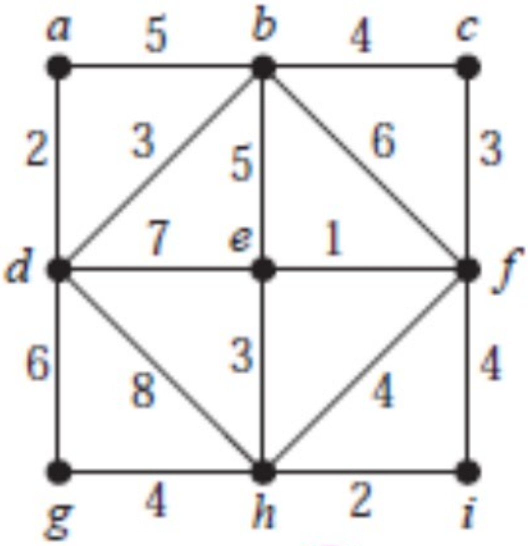


MST = 6



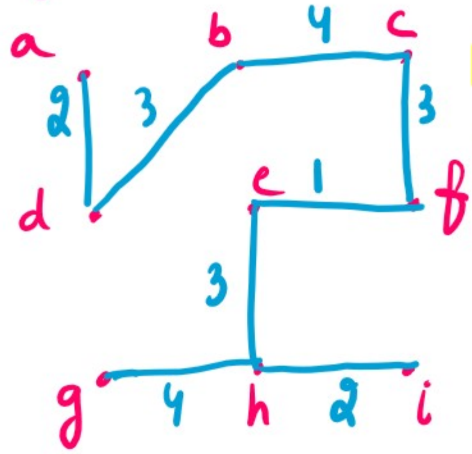
{a, b} 1
{a, e} 2
{e, d} 2
{d, c} 1

Q21. Use Prim's Algorithm to find minimum spanning tree



22

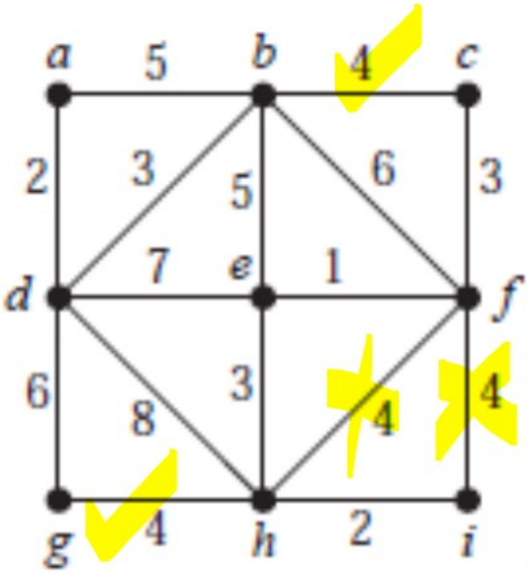
n=9
edges in Tree=8

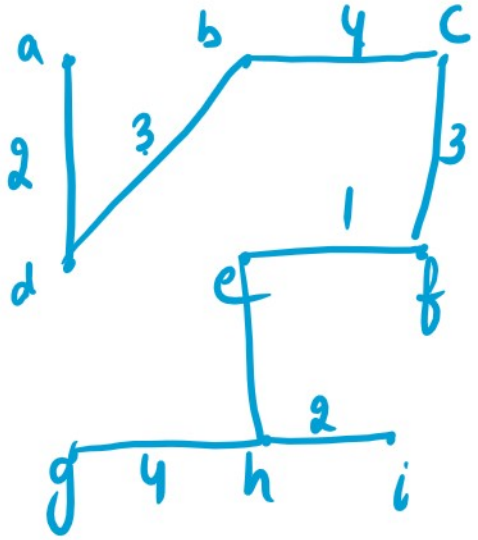


{e,f} 1
{e,h} 3
{h,i} 2
{f,c} 3
{g,h} 4
{b,c} 4
{b,d} 3
{a,d} 2

ALGORITHM 2 Kruskal's Algorithm.

procedure *Kruskal*(G: weighted connected undirected graph with n vertices)
T := empty graph
for i := 1 **to** $n - 1$
  e := any edge in G with smallest weight that does not form a simple circuit
    when added to T
  T := T with e added
return T {T is a minimum spanning tree of G}





{e,f} 1
{h,i} 2
{a,d} 2
{c,f} 3
{e,h} 3
{b,d} 3

{b,c} 4
{g,h} 4

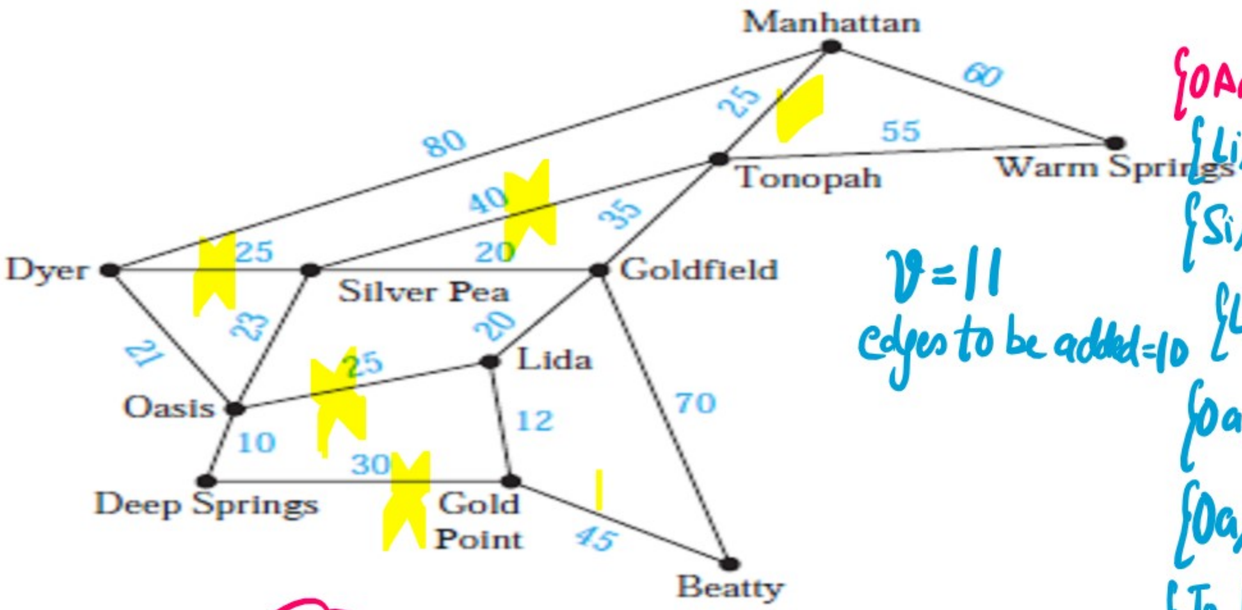
22

Q21.

Q21.

{Y, N} Y

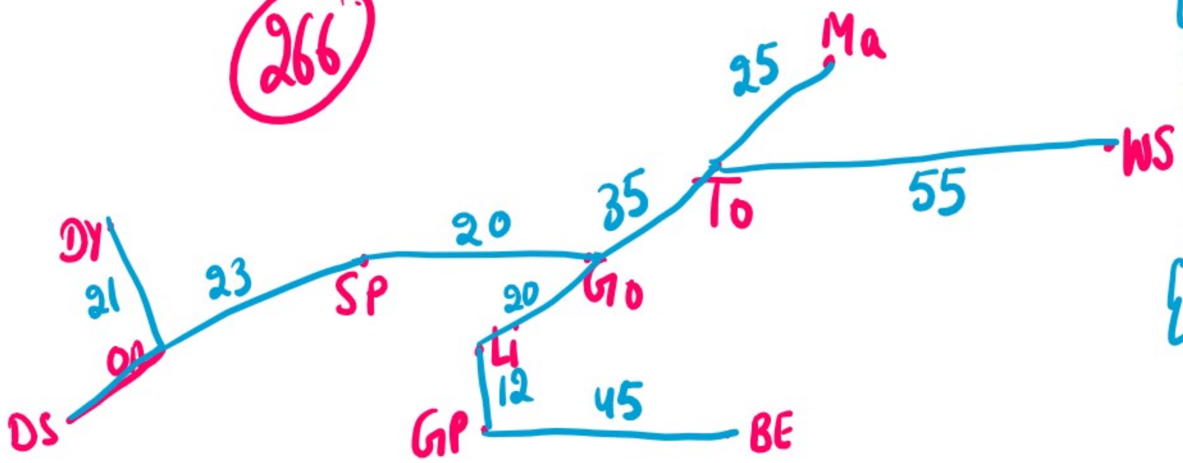
1. The roads represented by this graph are all unpaved. The lengths of the roads between pairs of towns are represented by edge weights. Which roads should be paved so that there is a path of paved roads between each pair of towns so that a minimum road length is paved? (*Note:* These towns are in Nevada.)



{OA, DS} 10
{Li, GP} 12
{Si, GF} 20
{Li, GF} 20
{Oa, DY} 21
{Oa, SP} 23
{To, Ma} 25
{GO, To} 35
{GP, BE} 45
{To, WS} 55

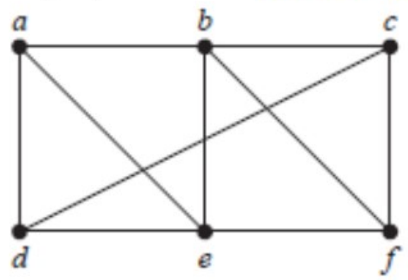
V = 11
edges to be added = 10

266



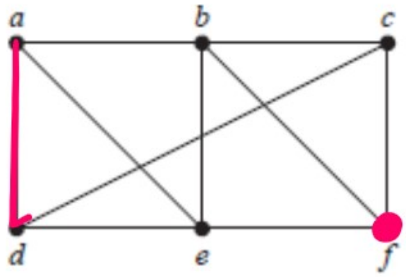
Vertex Connectivity

- A graph without cut vertices are called non-separable graphs.



No cut vertices.

- A subset V' of the vertex set V of $G = (V, E)$ is a **vertex cut or separating cut**, if $G - V'$ is disconnected.



$\{b, c, e\} \to$ vertex cut

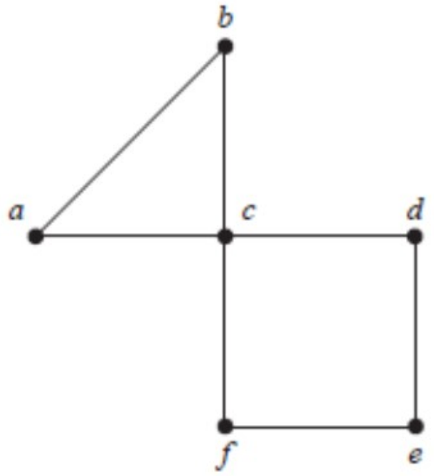
$K(G) = 3$

$K(K_n) = n-1$

- The minimum no. of vertices that can be removed from G to either disconnect G or produce a graph with a single vertex is known as vertex connectivity and is denoted as $\kappa(G)$.

Kappa(G)

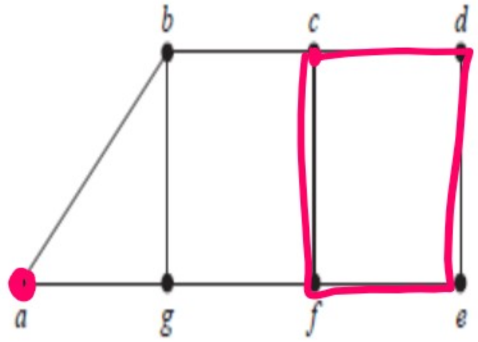
Q22. Find vertex connectivity for the following graphs.



Cut vertices ??

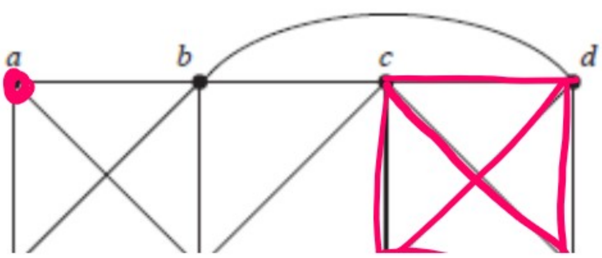
Yes $\{c\}$

$K(G) = 1$



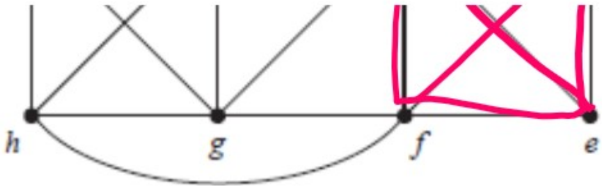
Cut vertices ?? No

$\therefore K(G) = 2$ $\{b, g\}$



No Cut vertices, $K(G) > 1$

$\{b, h, g\}$ $K(G) = 3$

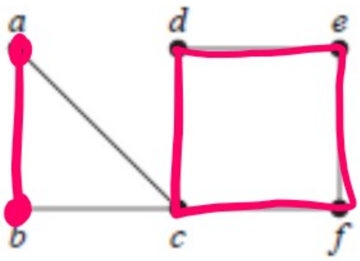


$\{b, h, g\}$ $\kappa(G) = 3$

- For disconnected graphs or K_1, $\kappa(G) = 0$.
- For connected graphs with cut vertices or K_2 , $\kappa(G) = 1.$
- A graph is k −connected or k −vertex connected, if $\kappa(G) \geq k$.

Edge Connectivity

- A subset E' of the vertex set E of $G = (V, E)$ is a **edge cut**, if $G - E'$ is disconnected.



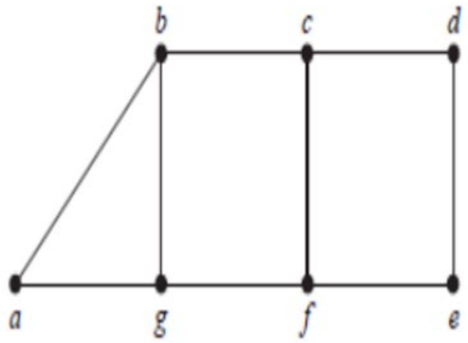
No Cut edge

Edge Cut

$\lambda(G) = 2$

- The minimum no. of edges in edge-cut is known as edge connectivity and is denoted as $\lambda(G)$.

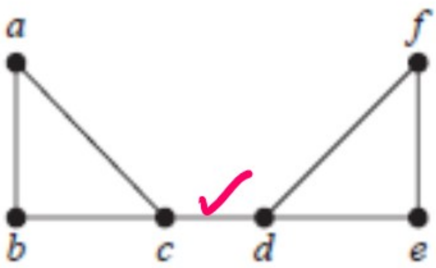
Q23. Find edge connectivity for the following graphs.



No Cut edge

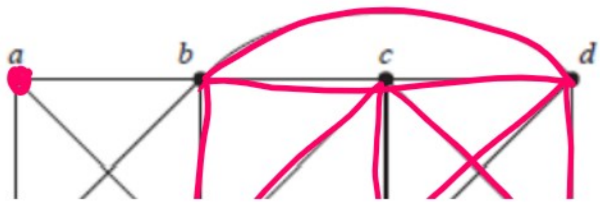
Edge Cut – $\{\{b, c\}, \{g, f\}\}$

$\lambda(G) = 2$

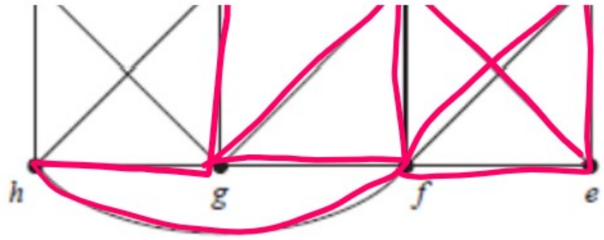


Yes Cut edge

$\lambda(G) = 1$



$\lambda(G) = 3$

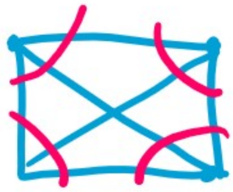


$\lambda(G) = 3$

- $\lambda(G) = 0$ if G is not connected or G consisting of single vertex.
- $0 \leq \lambda(G) \leq n - 1$.

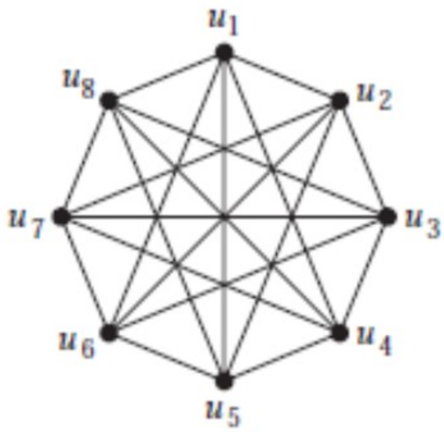
Simple graph n vertices
max degree $= n-1$

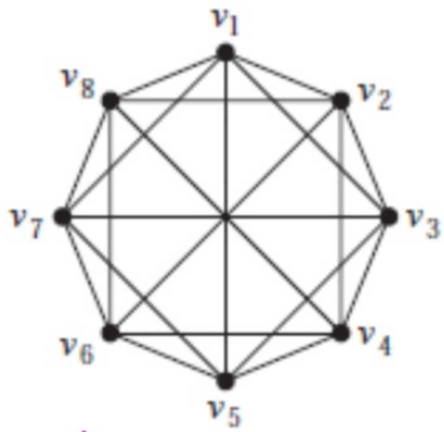
$\lambda(G) = n-1,\ G = K_n$



- $\lambda(G) \leq n - 2$ when G not a Complete graph.
- $\kappa(G) \leq \lambda(G) \leq \min_{v \in V} \deg(v)$.

Q24. To check whether the following graphs are isomorphic or not



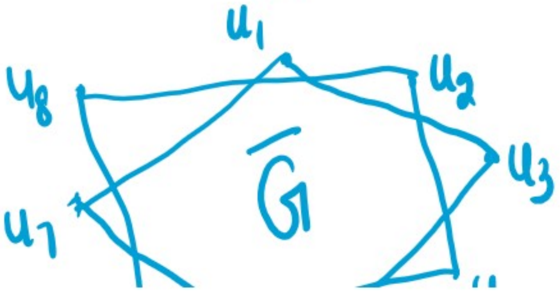


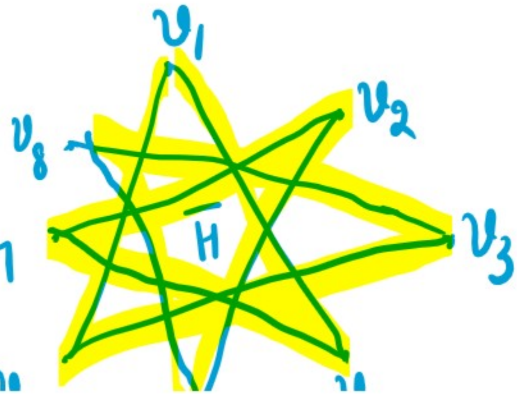
G

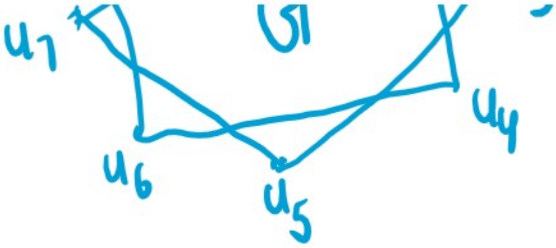
H

$\overline{G}$

$\overline{H}$







2 circuits of length 4.

1 circuit of length 8

Not isomorphic.

Read Decision Tree from Text Book