Chapter 4

Graph Theory

(Part 1)

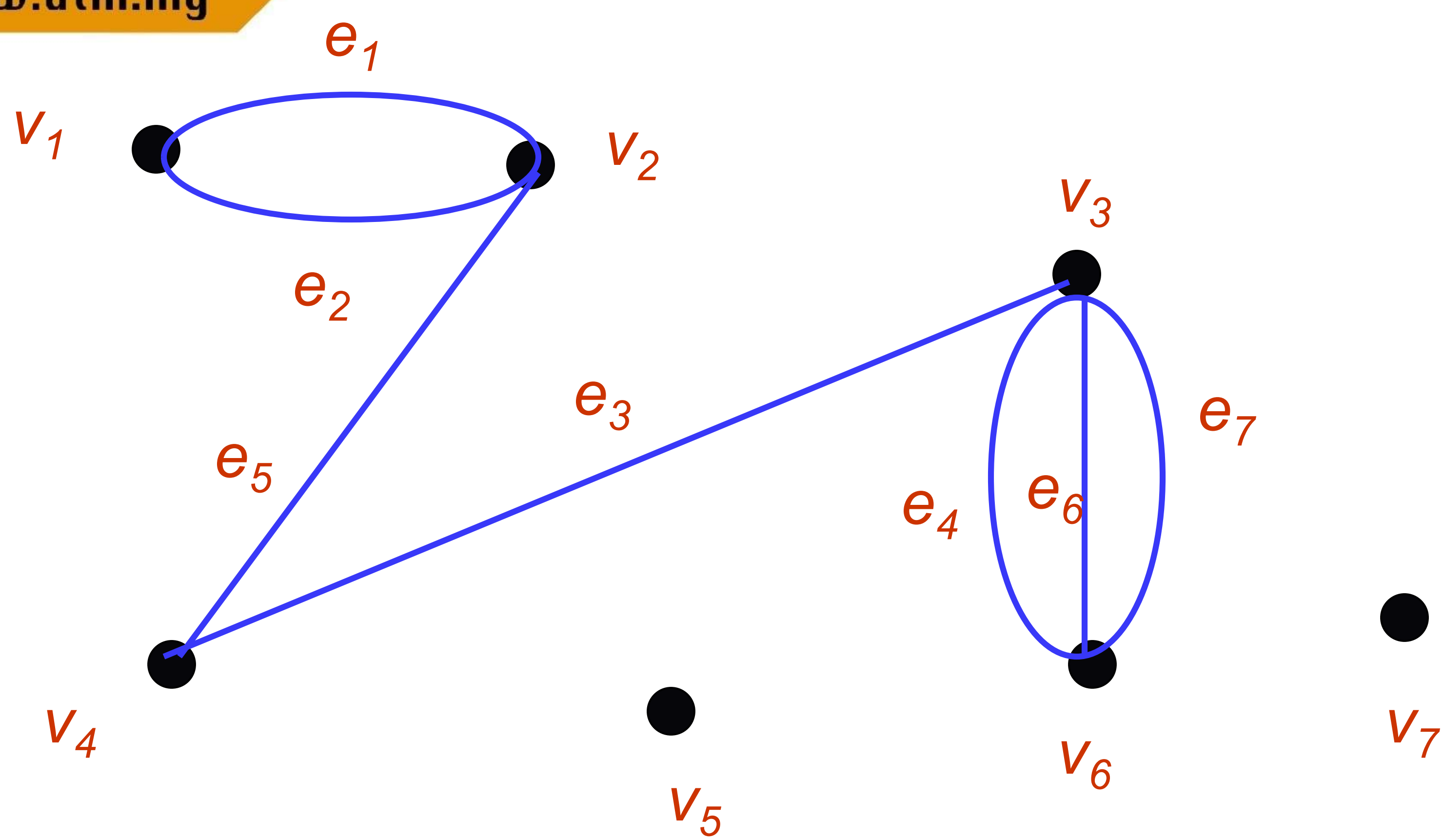
Definition

- A graph *G* is a triple (*V, E, f*), where
  - *V* is a finite nonempty set, called the set of vertices
  - *E* is a finite set (may be empty), called the set of edges
  - *f* is a function, called an incidence function, that assign to each edge, $e \in E$, a one-element subset $\{v\}$ or a two-element subset $\{v,w\}$, where *v* and *w* are vertices.
- We can write *G* as (*V,E, f*) or (*V,E*) or simply as *G*.

example

- Let,
  - $V=\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$
  - $E=\{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$
- and f be defined by
  - $f(e_1)= f(e_2)=\{v_1, v_2\}$
  - $f(e_3)= \{v_4, v_3\}$
  - $f(e_4)= f(e_6)= f(e_7)=\{v_6, v_3\}$
  - $f(e_5)= \{v_2, v_4\}$
- Then $G=(V,E,f)$ is a graph

example

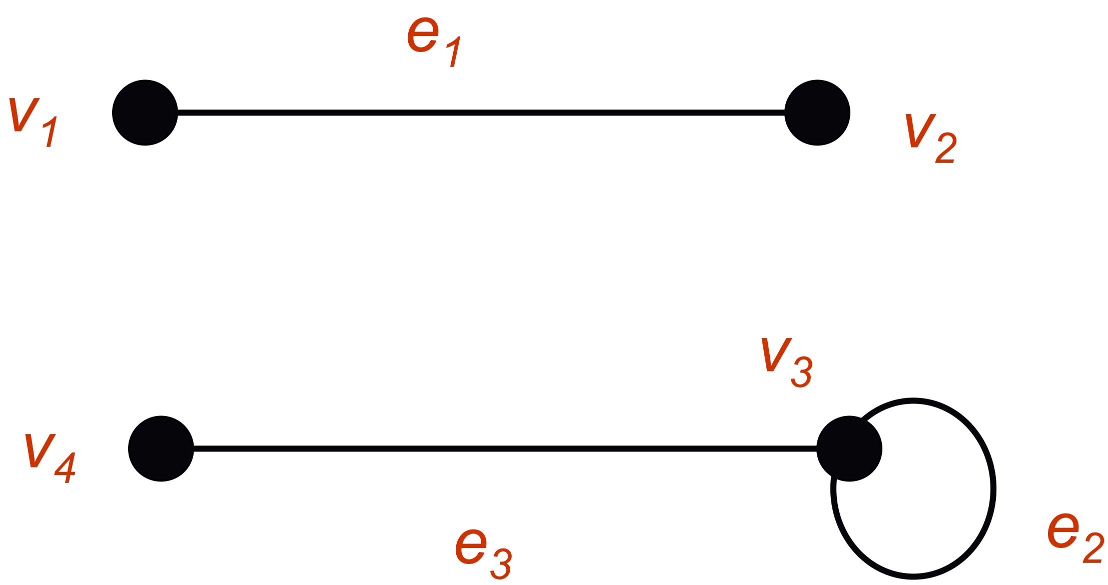


prepared by Razana Alwee

example

- Let $V=\{v_1, v_2, v_3, v_4\}$, $E= \{e_1, e_2, e_3\}$ and
  - $f(e_1)=\{v_1, v_2\}$
  - $f(e_2)= \{v_3, v_3\}$
  - $f(e_3)= \{v_3, v_4\}$
- Then $G=(V,E,f)$ is a graph

example



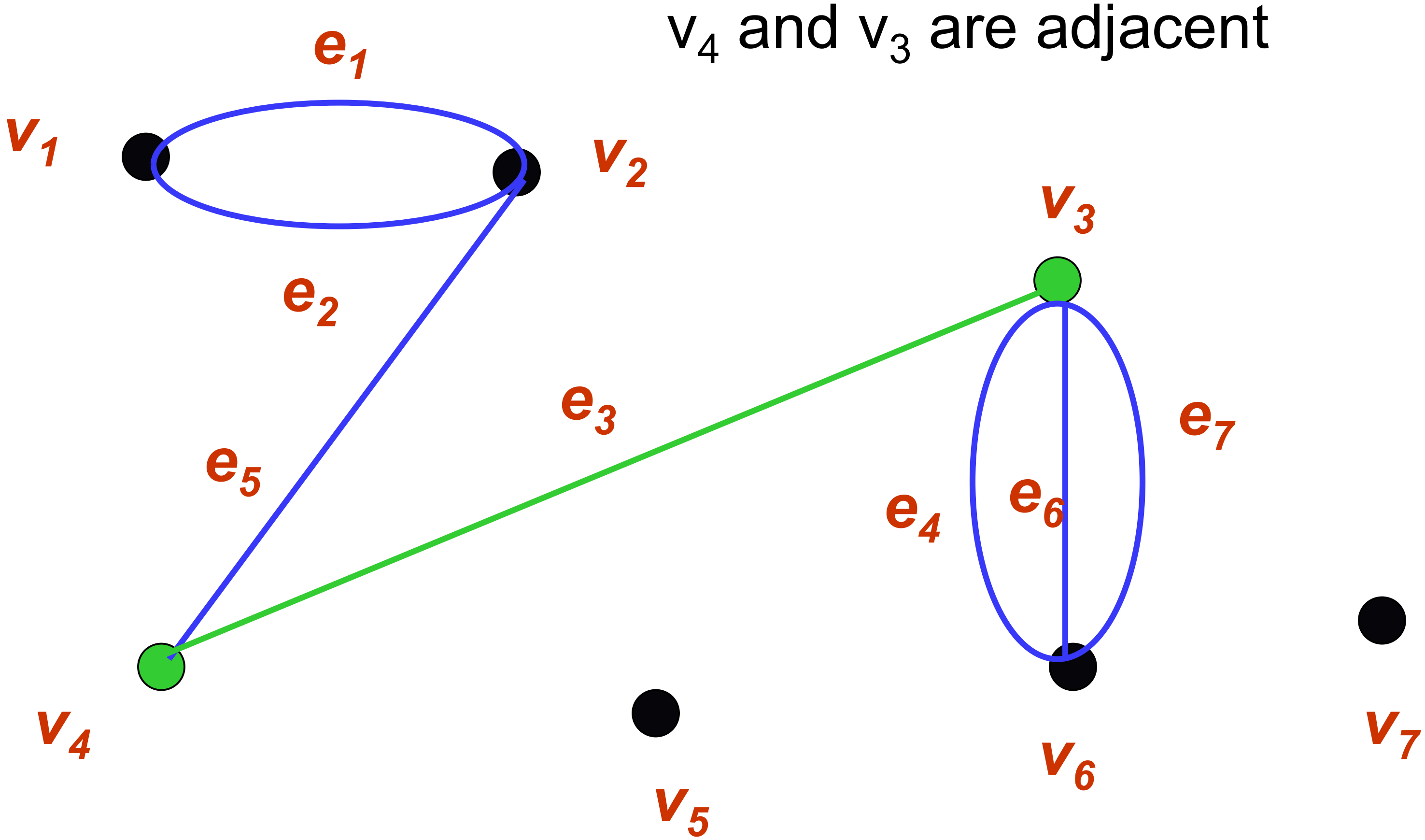
Characteristics of Graph

Adjacent Vertices

- An edge *e* in a graph that is associated with the pair of vertices *v* and *w* is said to be incident on *v* and *w*, and *v* and *w* are said to be incident on *e* and to be adjacent vertices.
- A vertex that is an endpoint of a loop is said to be adjacent to itself.

example

v_4 and v_3 are adjacent



prepared by Razana Alwee

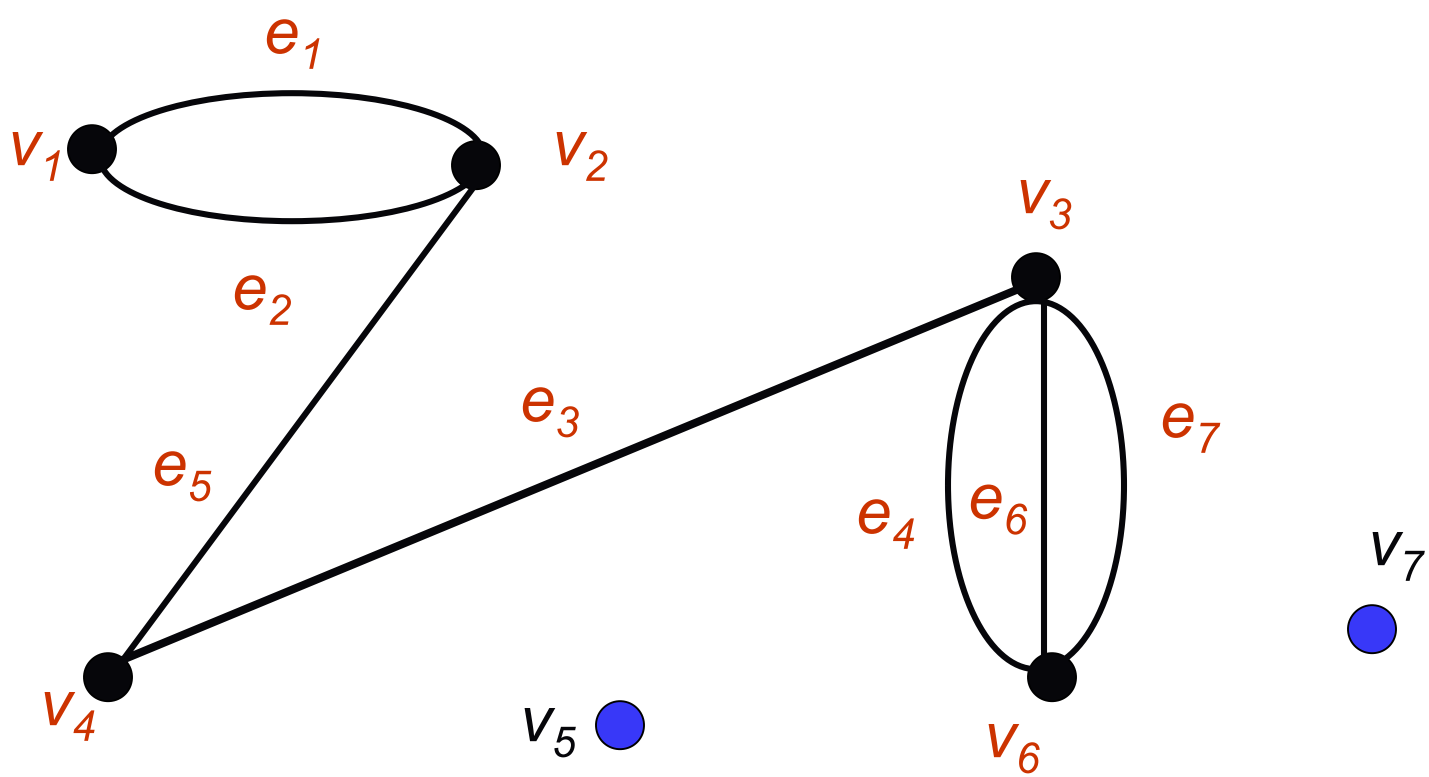
Isolated Vertex

- Let *G* be a graph and *v* be a vertex in *G*.

- We say that *v* is an isolated vertex if it is not incident with any edge.

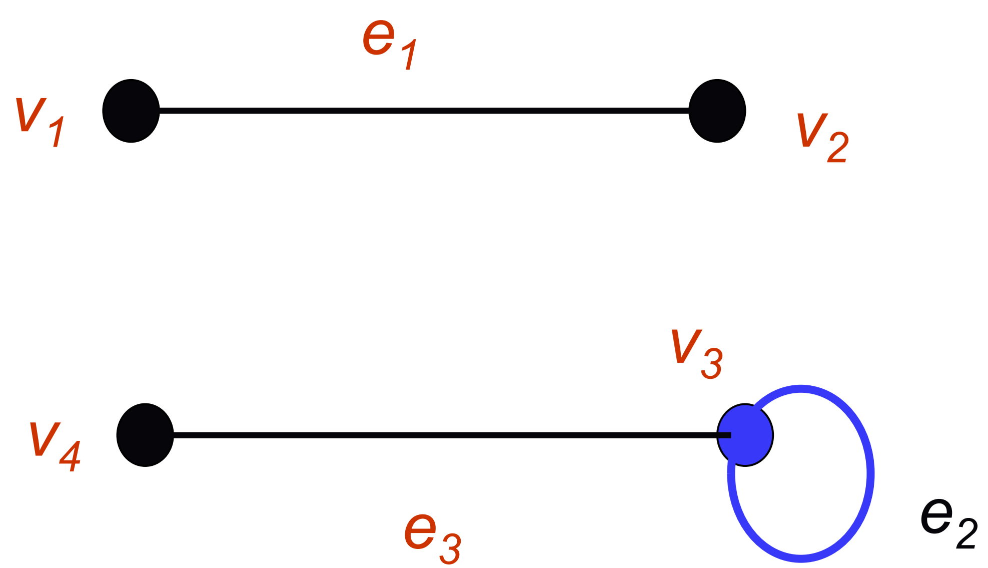
example

- *v5* and *v7* are isolated vertices.



Loop

- An edge incident on a single vertex is called a loop.
  Example: e_2 is a loop

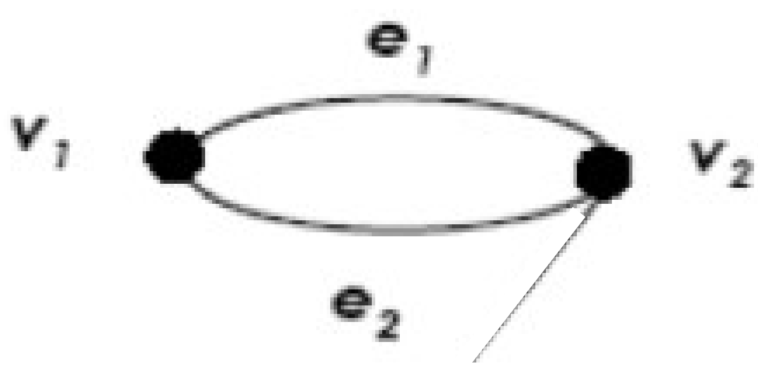


Parallel Edges

Two or more distinct edges with the same set of endpoints are said to be parallel.

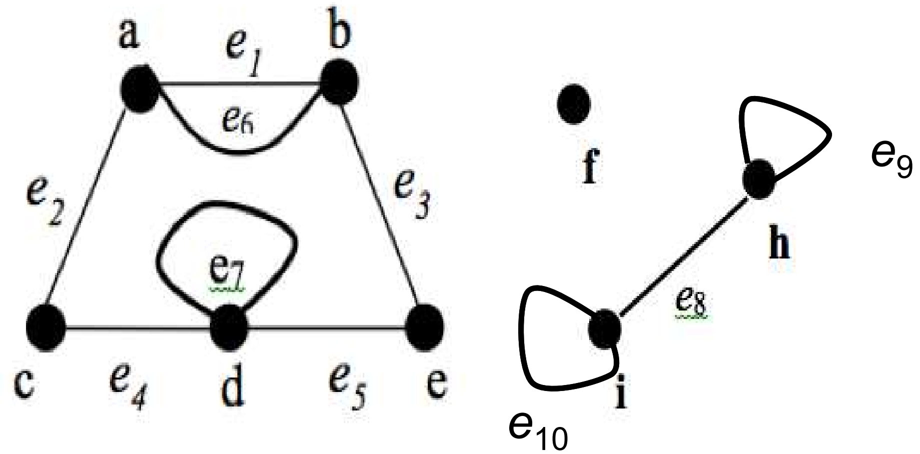
Example

- e_1 and e_2 are parallel.



Example

Given a graph as shown below,



a) Write a vertex set and the edge set, and give a table showing the edge-endpoint function.

b) Find all edges that are incident on a, all vertices that are adjacent to a, all edges that are adjacent to e_2, all loops, all parallel edges, all vertices that are adjacent to themselves and all isolated vertices.

Example 1 - Solution

Solution:

a) Vertex set, V = {a, b, c, d, e, f, i, h} and
the set of edges, E = {e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10}}

Edge	Endpoints
e_1	{a, b}
e_2	{a, c}
e_3	{b, e}
e_4	{c, d}
e_5	{d, e}
e_6	{a, b}
e_7	{d}
e_8	{i, h}
e_9	{h}
e_{10}	{i}

b)

incident on a, e1, e2, e6
adjacent to a, c, b
adjacent to e_2, e1, e4, e6
loops, e7, e9, e10
parallel edges, e1, e6
adjacent to themselves, i , h, d
isolated vertices, f

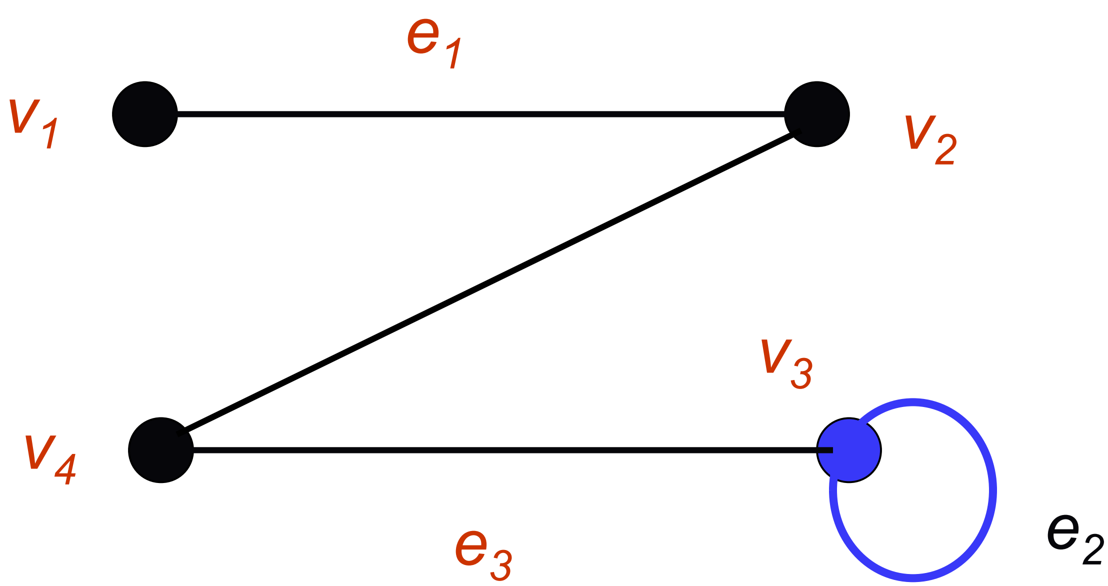
The Concept of Degree

Degree of a vertex

- Let *G* be a graph and *v* be a vertex of *G*.
- The degree of *v*, written **deg(*v*)** or **d(*v*)** is the number of edges incident with *v*.
- Each loop on a vertex *v* contributes **2** to the degree of *v*.

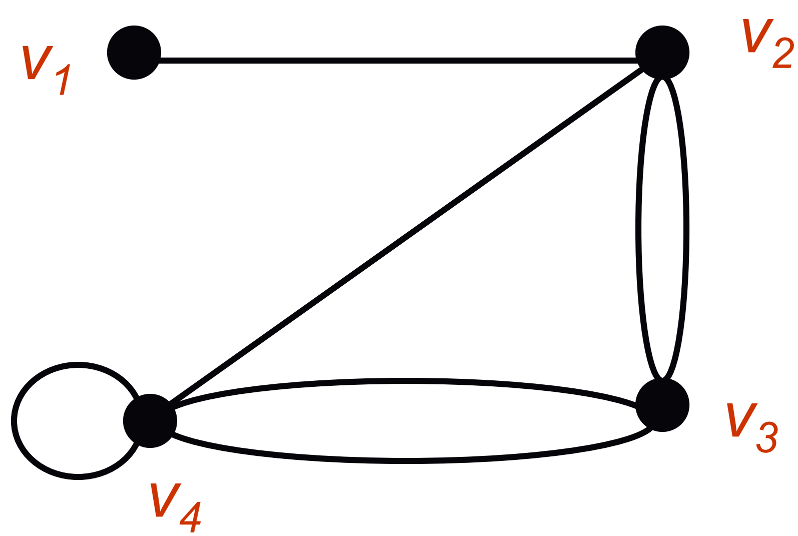
example

- deg(v_1)= 1; deg(v_2)= 2; deg(v_3)= 3; deg(v_4)= 2



exercise

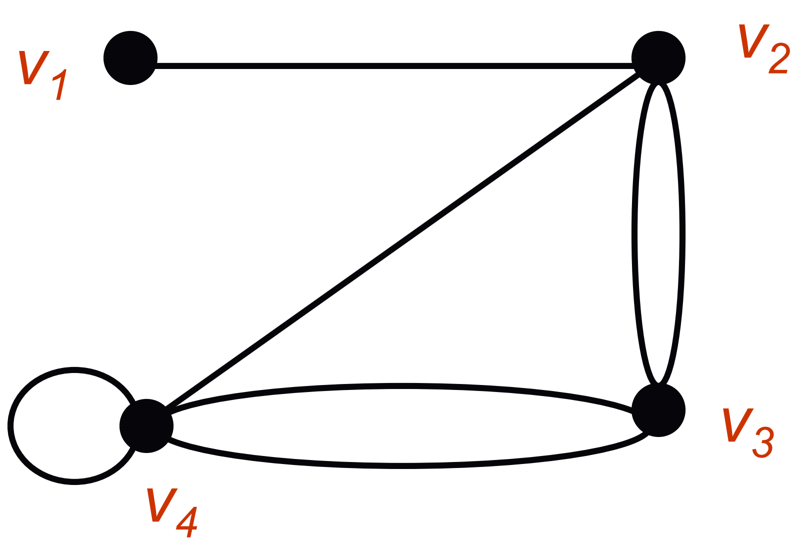
- Find the degree of each vertex in the graph.



prepared by Razana Alwee

solution

- Find the degree of each vertex in the graph.



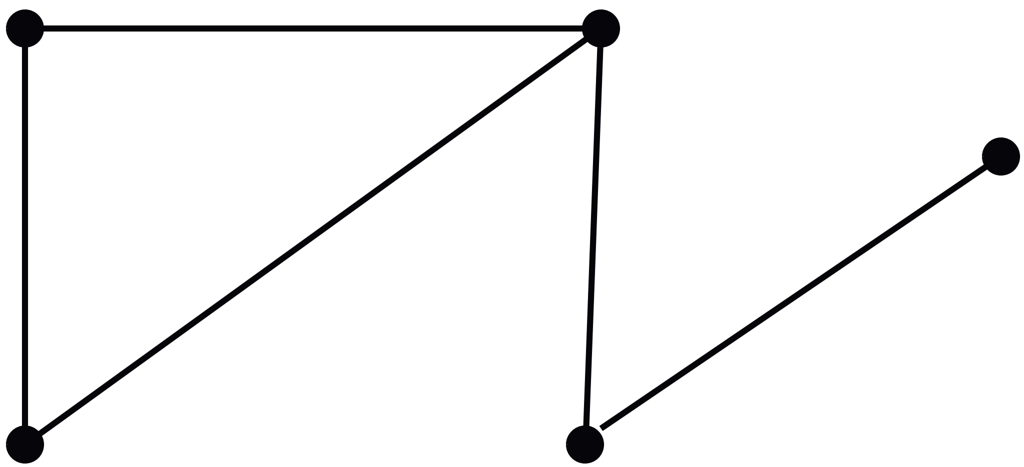
Solution: deg(v_1)= 1; deg(v_2)= 4; deg(v_3)= 4; deg(v_4)= 5

prepared by Razana Alwee

Types of Graphs

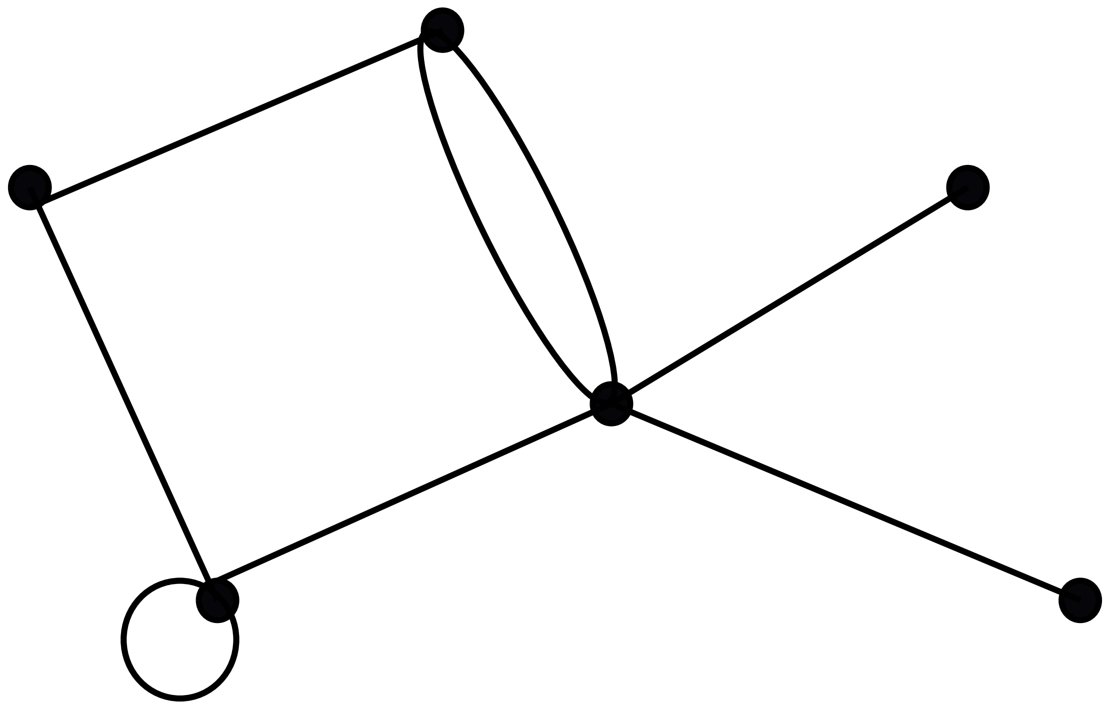
Simple Graphs

- A graph *G* is called a simple graph if *G* does not contain any parallel edges and any loops.
- **Example**



Connected Graph

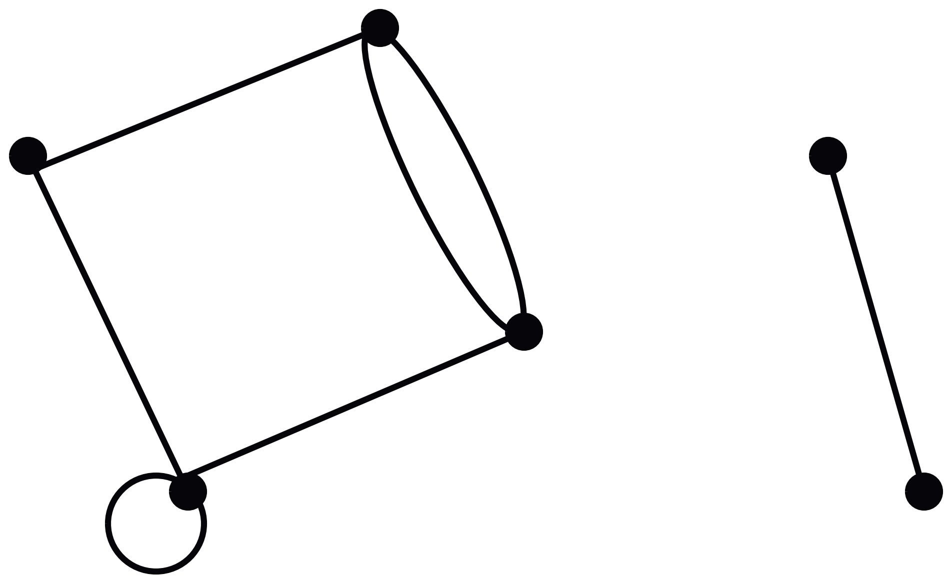
- A graph *G* is connected if given any vertices *v* and *w* in *G*, there is a path from *v* to *w*.
- **Example:**



prepared by Razana Alwee

example

- not connected



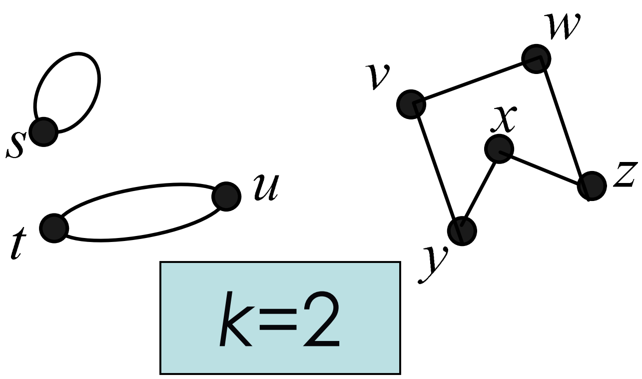
prepared by Razana Alwee

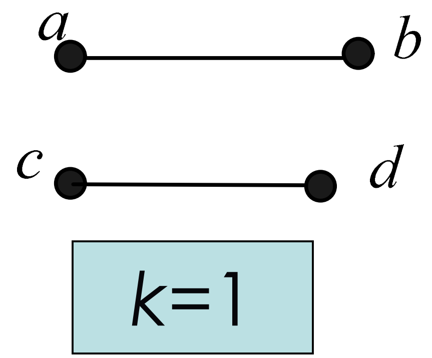
Regular Graphs

- Let G be a graph and k be a nonnegative integer.
- G is called a k-regular graph if the degree of each vertex of G is k.

prepared by Razana Alwee

example

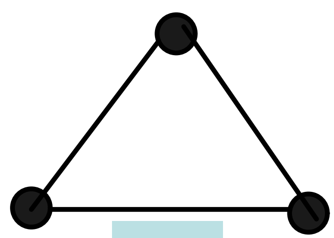




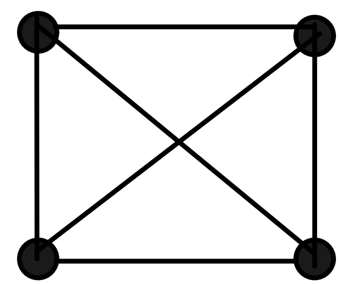
Complete Graph

- A simple graph with n vertices in which there is an edge between every pair of distinct vertices is called a complete graph on n vertices.
- This is denoted by K_n.
- **Example**

K_2



K_3



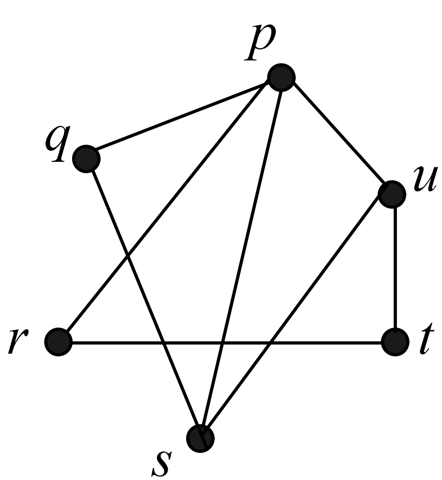
K_4

Subgraph

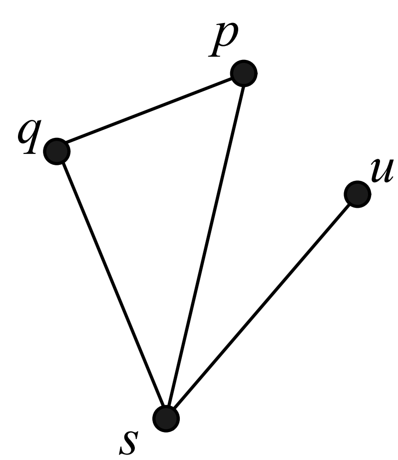
- Let $G=(V,E)$ be a graph.

- $H=(U,D)$ is a subgraph of G if
  - $U \subseteq V$ and $D \subseteq E$
  - for every edge $e \in D$, if e is incident on v and w, then $v,w \in V$.

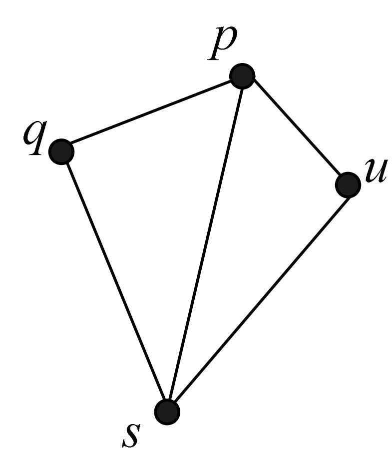
example



G



H_1



H_2

Graph Representation

Matrix Representation of a Graph

- To write programs that process and manipulate graphs, the graphs must be stored, that is, represented in computer memory.

- A graph can be represented (in computer memory) in several ways.

- 2-dimensional array: adjacency matrix and incidence matrix.

prepared by Razana Alwee

Adjacency Matrices

- Let G be a graph with n vertices.
- The adjacency matrix, A_G is an $n \times n$ matrix $[a_{ij}]$ such that,

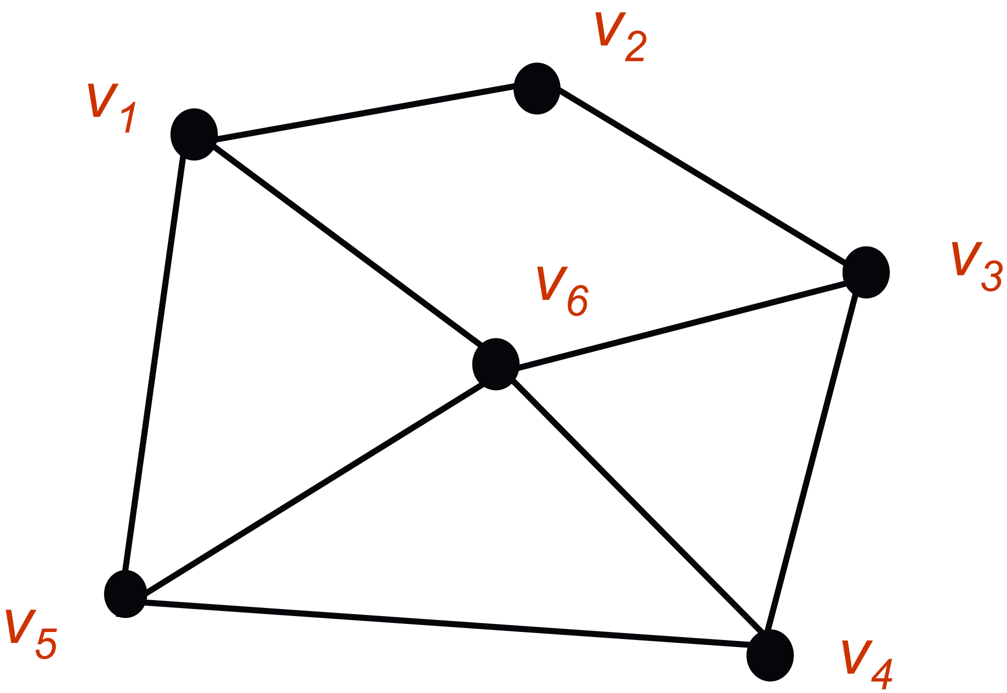
a_{ij}= the number of edges from v_i to v_j, {undirected G}

or,

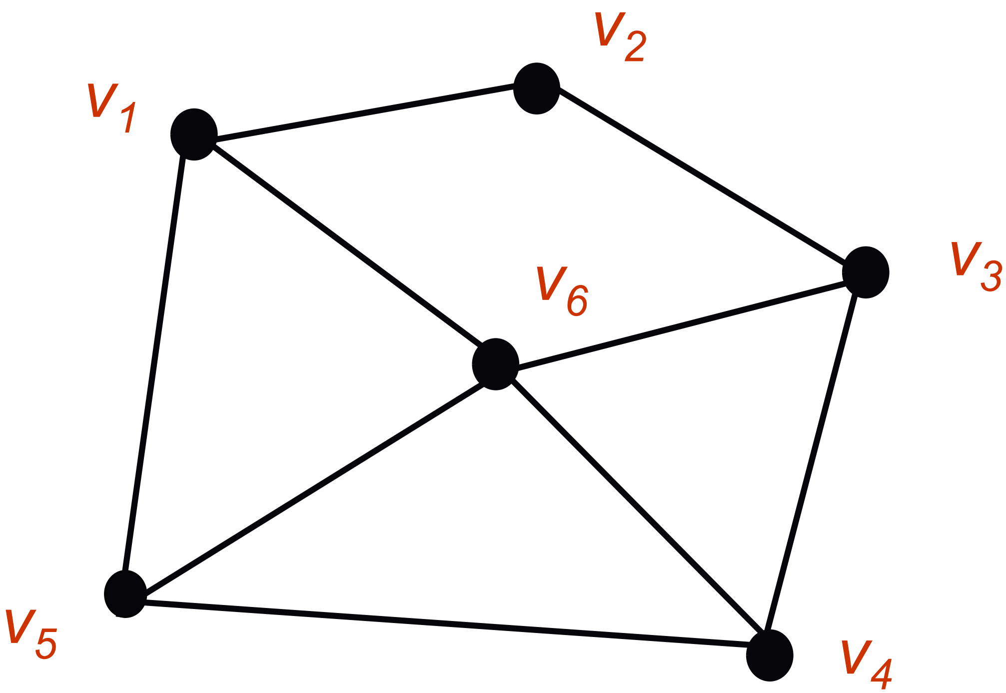
a_{ij}= the number of arrows from v_i to v_j, {directed G}

for all i, j = 1,2,….., n.

example

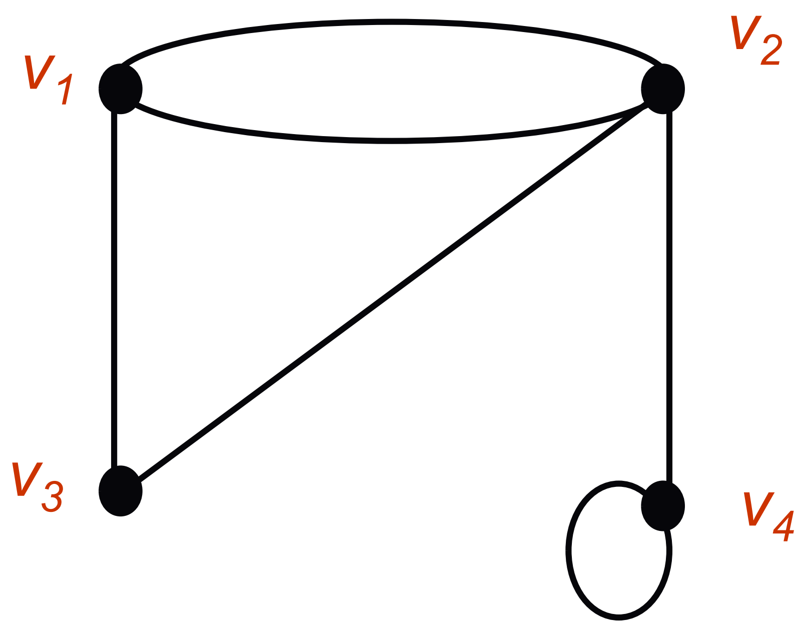


example

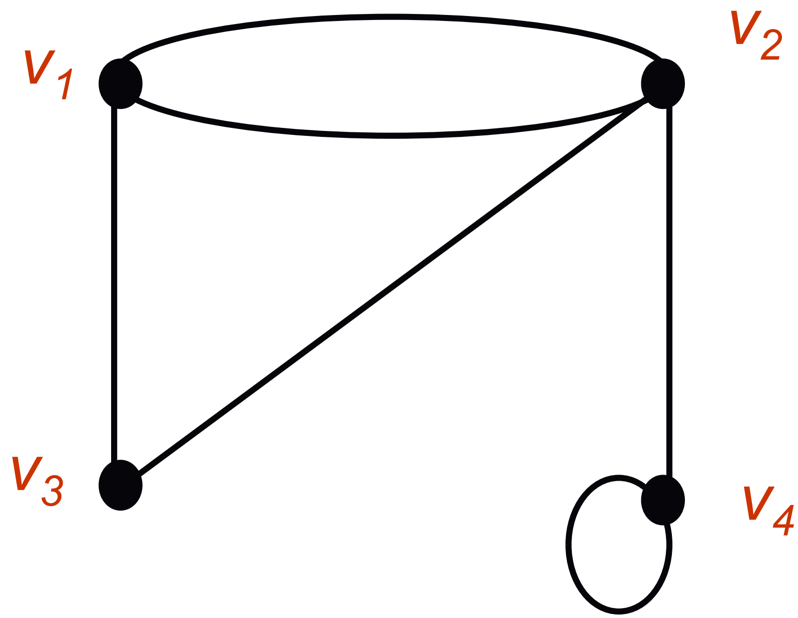


$$A_G = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

example



example

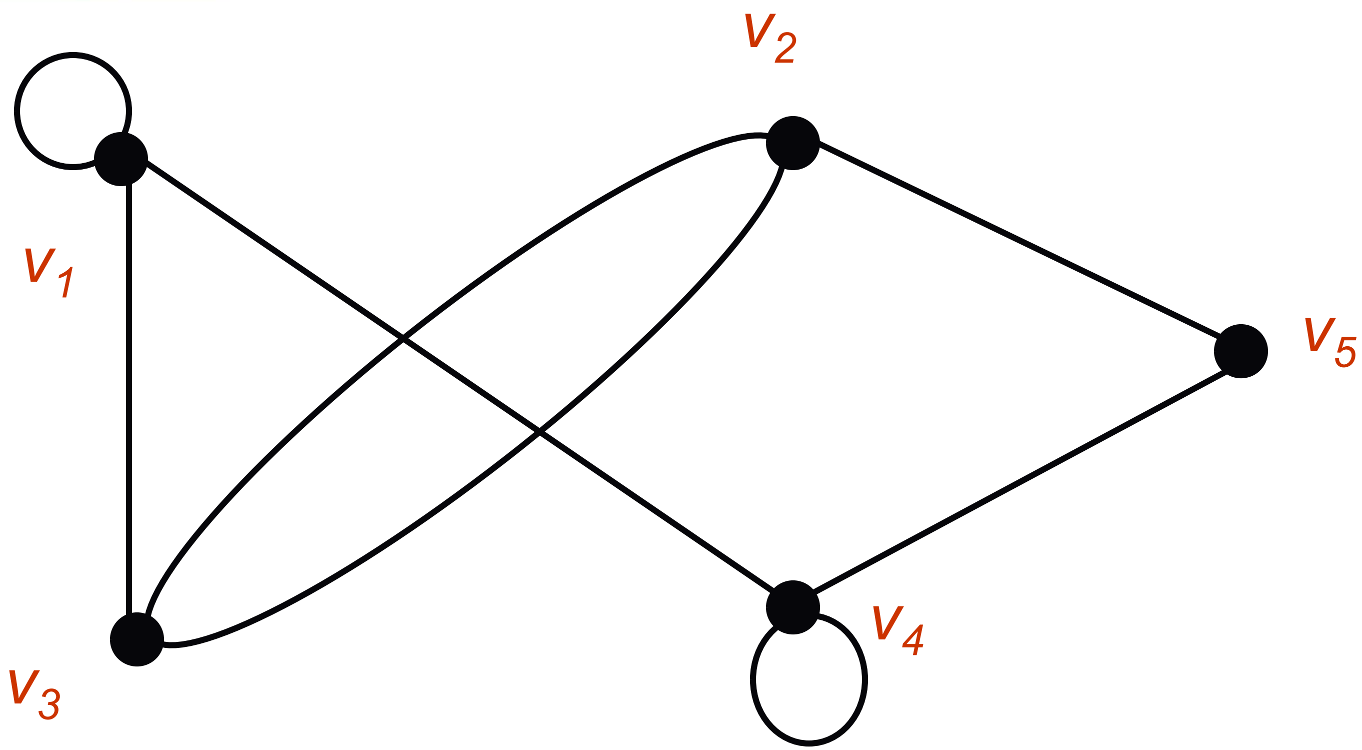


$$A_G = \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

example

$$A_G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 2 & 0 & 1 \\ 1 & 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

example



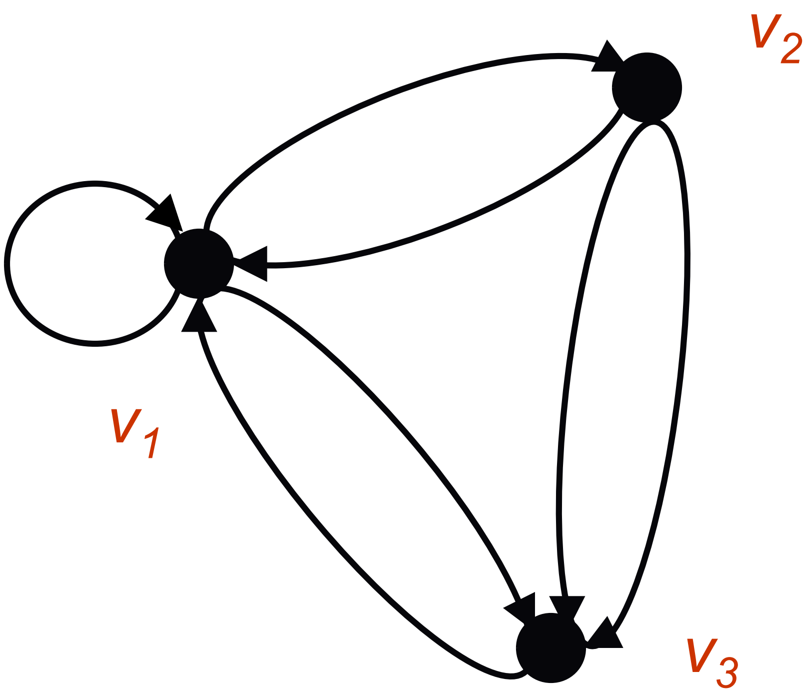
Adjacency Matrices

- Notice that the matrix A_G is a **symmetric matrix** if it is representing an undirected graph, where

$$a_{ij} = a_{ji}$$

- If G is a directed graph (**digraph**), then A_G need not be a symmetric matrix.

example



$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 0 & 0 \end{bmatrix}$$

prepared by Razana Alwee

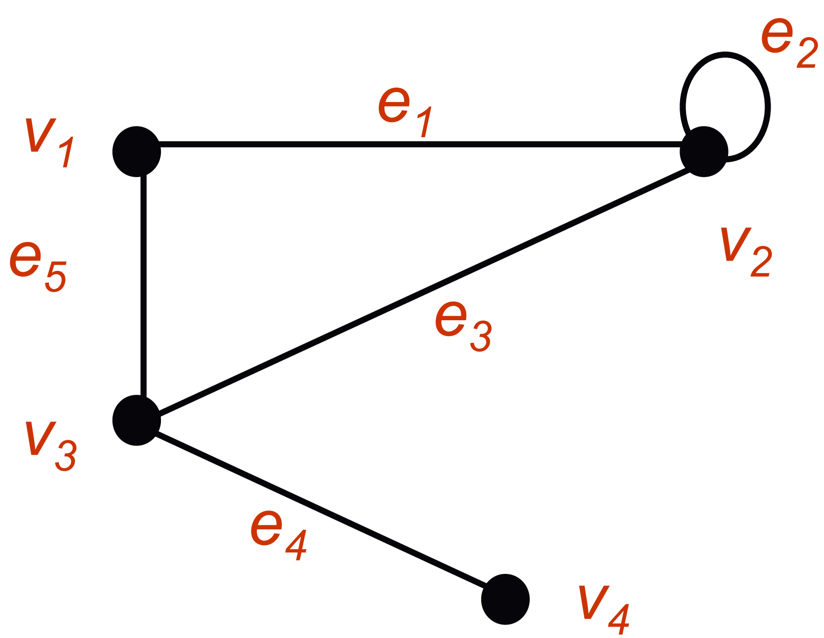
Incidence Matrices

- Let G be a graph with n vertices and m edges.

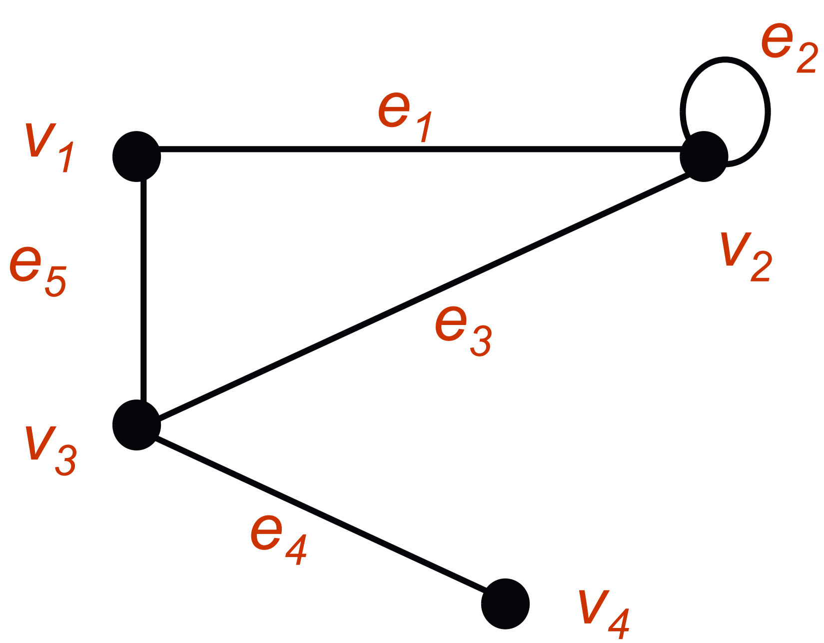
- The incidence matrix I_G is an $n \times m$ matrix $[a_{ij}]$ such that,

$$a_{ij} = \begin{cases} 0 & \text{if } v_i \text{ is not an end vertex of } e_j, \\ 1 & \text{if } v_i \text{ is an end vertex of } e_j \text{, but } e_j \text{ is not a loop} \\ 2 & \text{if } e_j \text{ is a loop at } v_i \end{cases}$$

example



example



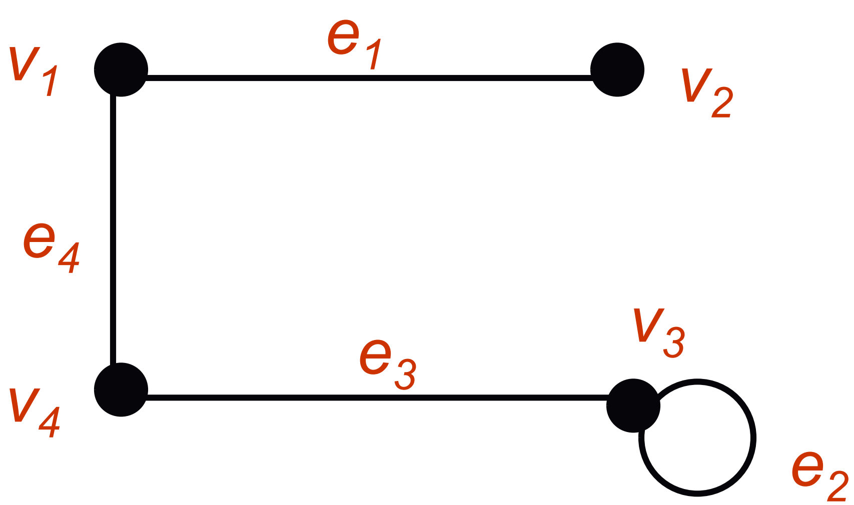
$$
\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array}
\begin{array}{c}
\begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\
\begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}
\end{array}
$$

Notice that the sum of the *i*th row is the degree of v_i

prepared by Razana Alwee

exercise

- Find the adjacency matrix and the incidence matrix of the graph.



Exercise
Past Year 2015/2016

A cat show is being judged from pictures of the cats. The judges would like to see pictures of the following pairs of cats next to each other for their final decision: Fifi and Putih, Fifi and Suri, Fifi and Bob, Bob and Cheta, Bob and Didi, Bob and Suri, Cheta and Didi, Didi and Suri, Didi and Putih, Suri and Putih, Putih and Jeep, Jeep and Didi.

Draw a graph modeling this situation. (3 marks)

Exercise
Past Year 2015/2016

Given a graph as shown in Figure 1.

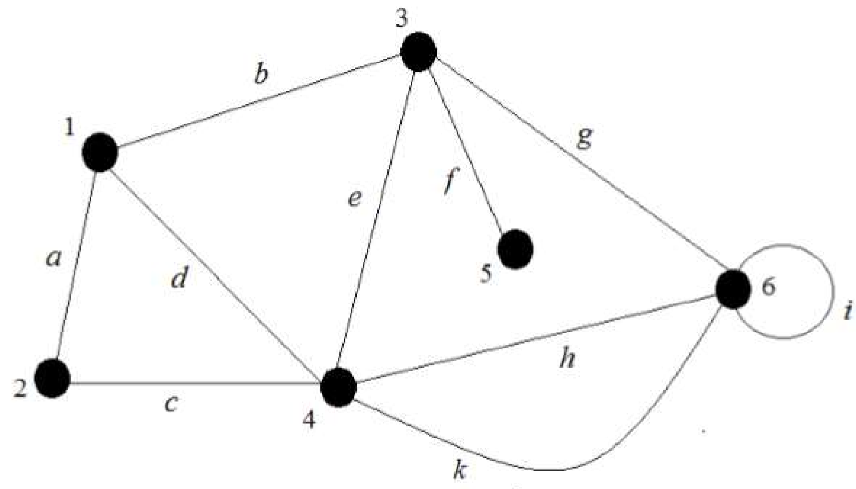


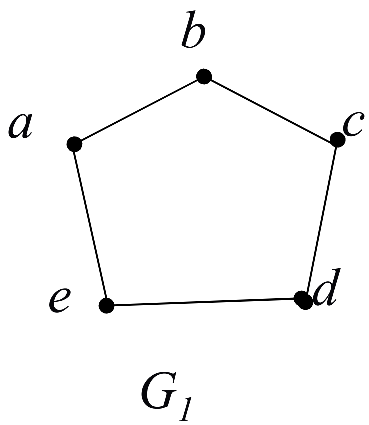
Figure 1

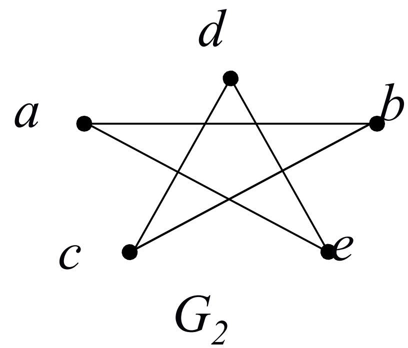
i. Find the incidence matrix of the graph. (4 marks)

ii. Find the adjacency matrix of the graph. (3 marks)

Isomorphisms

Isomorphism





- Are these 2 graphs the same?

- When we say that 2 graphs are the same mean they are isomorphic to each other.

Isomorphism

- Graphs G_1 and G_2 are isomorphic if there is

  a one-to-one, onto function *f* from the vertices of G_1 to the vertices of G_2

  and

  a one-to-one, onto function *g* from the edges of G_1 to the edges of G_2

Isomorphism

- An edge *e* is incident on *v* and *w* in G_1 if and only if the edge *g(e)* is incident on *f(v)* and *f(w)* in G_2.
- The pair of functions *f* and *g* is called an isomorphism of G_1 onto G_2.
- Graphs G_1 and G_2 are isomorphic if and only if for some ordering of their vertices, their adjacency matrices are equal.

Definition

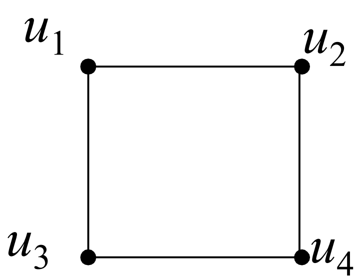
Let $G=\{V,E\}$ and $G'=\{V',E'\}$ be graphs. G and G' are said to be isomorphic if there exist a pair of functions $f:V\rightarrow V'$ and $g:E\rightarrow E'$ such that f associates each element in V with exactly one element in V' and vice versa; g associates each element in E with exactly one element in E' and vice versa, and for each $v\in V$, and each $e\in E$, if v is an endpoint of the edge e, then $f(v)$ is an endpoint of the edge $g(e)$.

Isomorphism

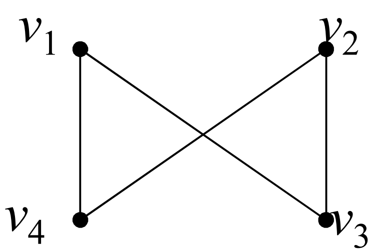
- If two graphs is isomorphic, they must have:
  - the same number of vertices and edges,
  - the same degrees for corresponding vertices,
  - the same number of connected components,
  - the same number of loops and parallel edges,
  - both graphs are connected or both graph are not connected,
  - pairs of connected vertices must have the corresponding pair of vertices connected.
- In general, it is easier to prove two graphs are not isomorphic by proving that one of the above properties fails.

example

- Determine whether *G* is isomorphic to *H.*



G



H

example

- Both graphs are simple and have the same number of vertices and the same number of edges.

example

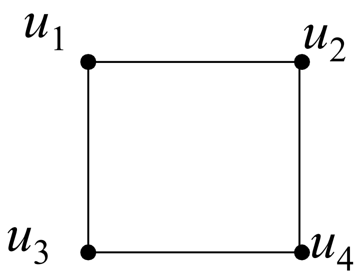
- All the vertices of both graphs have degree 2.

prepared by Razana Alwee

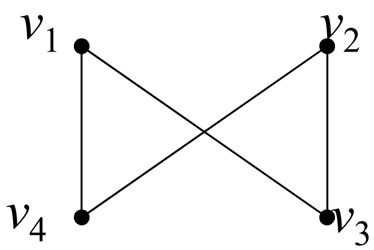
example

- Define f: $U \rightarrow V$, where $U=\{u_1, u_2, u_3, u_4\}$ and $V=\{v_1, v_2, v_3, v_4\}$

$f(u_1)=v_1$, $f(u_2)=v_4$, $f(u_3)=v_3$, $f(u_4)=v_2$



G



H

prepared by Razana Alwee

example

- To verify whether G and H are isomorphic, we examine the adjacency matrix A_G with rows and columns labeled in the order u_1, u_2, u_3, u_4
  and
  the adjacency matrix A_H with rows and columns labeled in the order v_1, v_4, v_3, v_2.

example

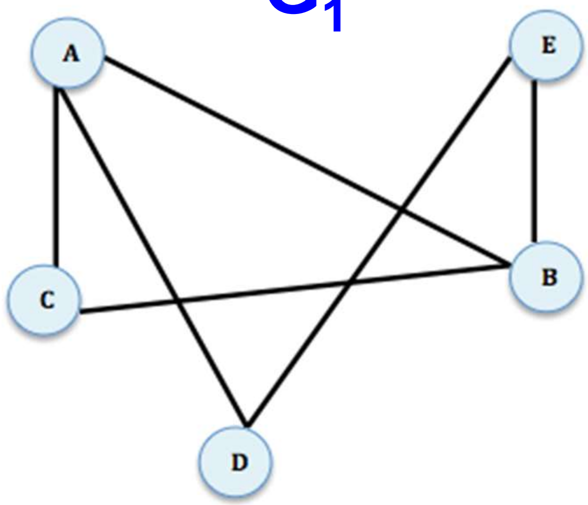
- A_G and A_H are the same, G and H are isomorphic.

$$A_G = \begin{array}{c c} & \begin{array}{cccc} u_1 & u_2 & u_3 & u_4 \end{array} \\ \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \end{array} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array} \qquad A_H = \begin{array}{c c} & \begin{array}{cccc} v_1 & v_4 & v_3 & v_2 \end{array} \\ \begin{array}{c} v_1 \\ v_4 \\ v_3 \\ v_2 \end{array} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

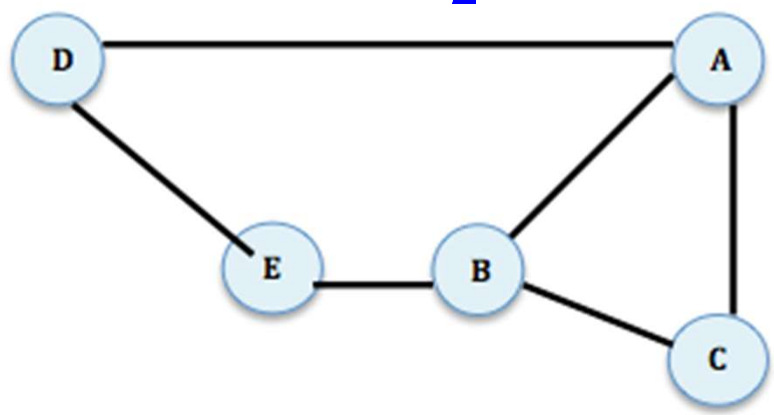
Exercise

Q: Show that the following two graphs are isomorphic.

G_1



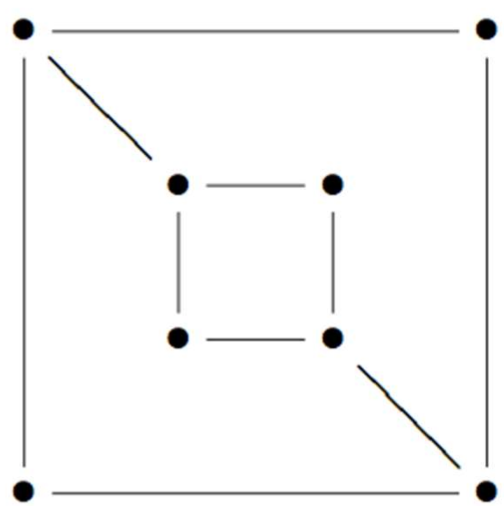
G_2



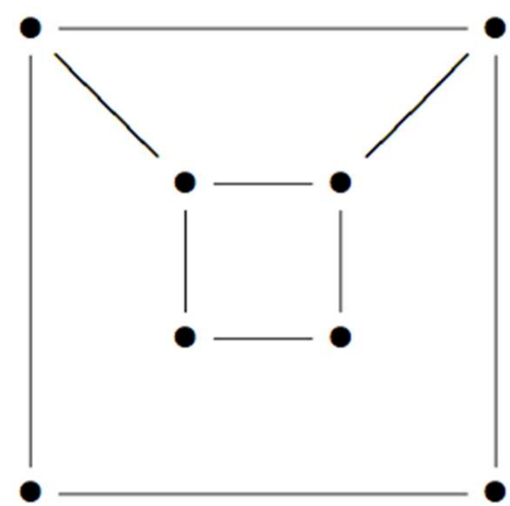
Exercise

Q: Is these two graphs are isomorphic?

$G:$

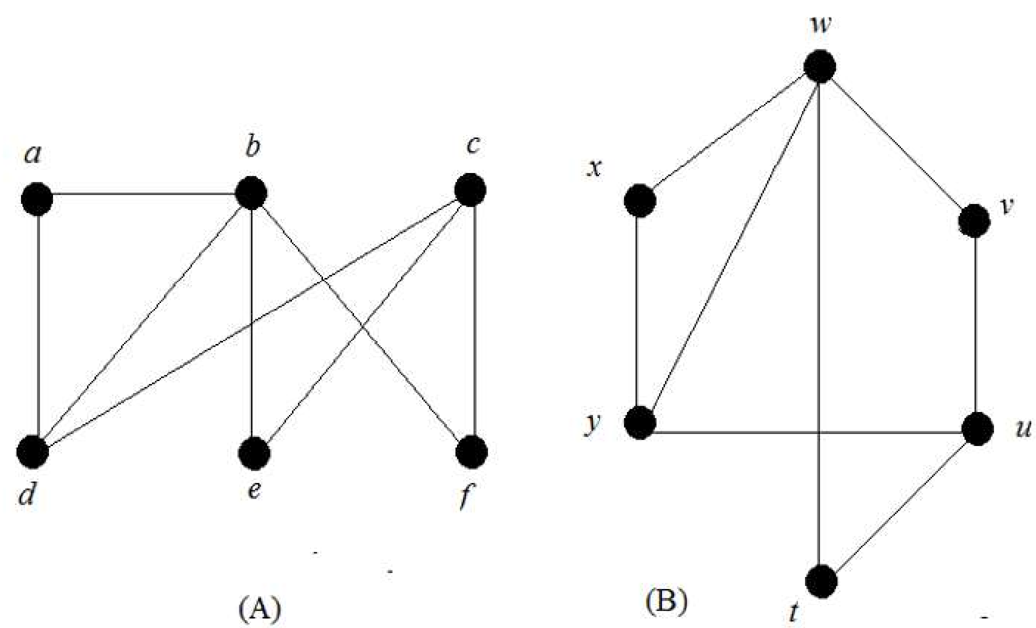


$H:$



Exercise
Past Year 2015/2016

Determine whether the graphs in Figure 2 (A and B) are isomorphic. If the graphs are isomorphic, find their adjacency matrices; otherwise, give an invariant that the graphs do not share. (6 marks)



Trails, Paths & Circuits

Term and Description

- A walk from v to w is a finite alternating sequence of adjacent vertices and edges of G. Thus a walk has the form

$$(v_0, e_1, v_1, e_2, v_2, \ldots\ldots, v_{n-1}, e_n, v_n)$$

where the v's represent vertices, the e's represent edges, $v = v_0$, $w = v_n$, and for i = 1,2,…,n. v_{i-1} and v_i are the endpoints of e_i.

- A trivial walk from v to w consist of the single vertex v. The walk contains zero edges (has length zero)

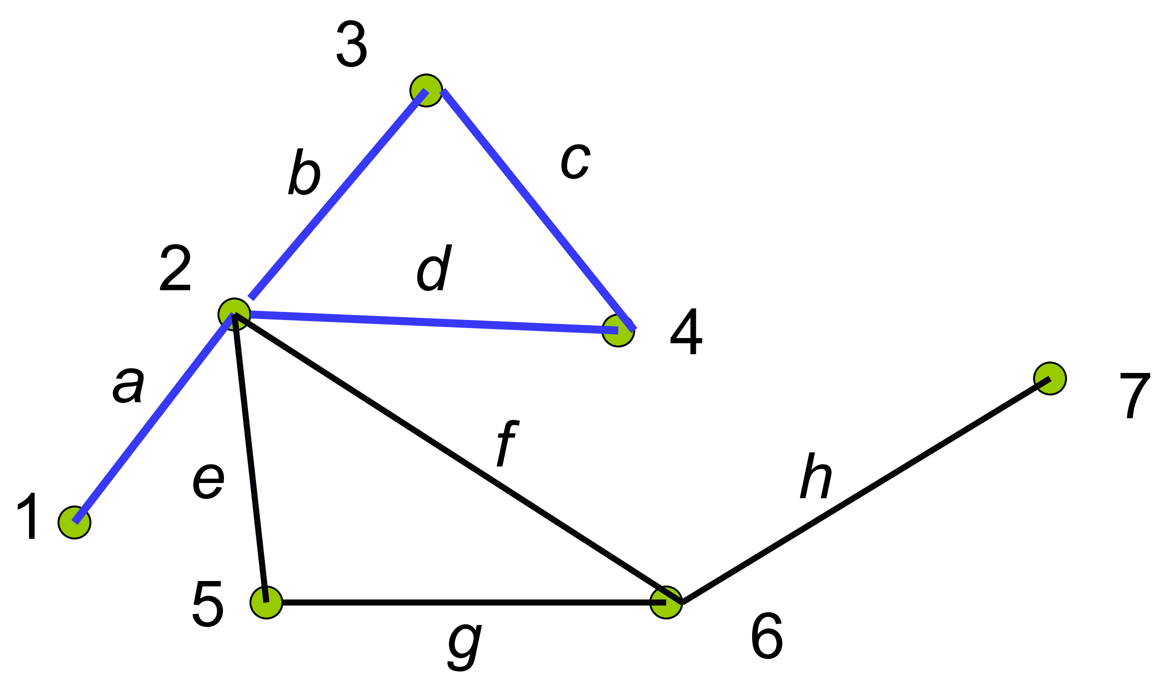
- The length of a walk is the number of edges it has.

Term and Description (cont.)

- A trail from v to w is a walk from v to w that does not contain a repeated edge.
- A path from v to w is a trail from v to w that does not contain a repeated vertex.
- A closed walk is a walk that start and ends at the same vertex.
- A circuit/cycle is a closed walk that contains at least one edge and does not contain a repeated edge.
- A simple circuit is a circuit that does not have any other repeated vertex except the first and the last.

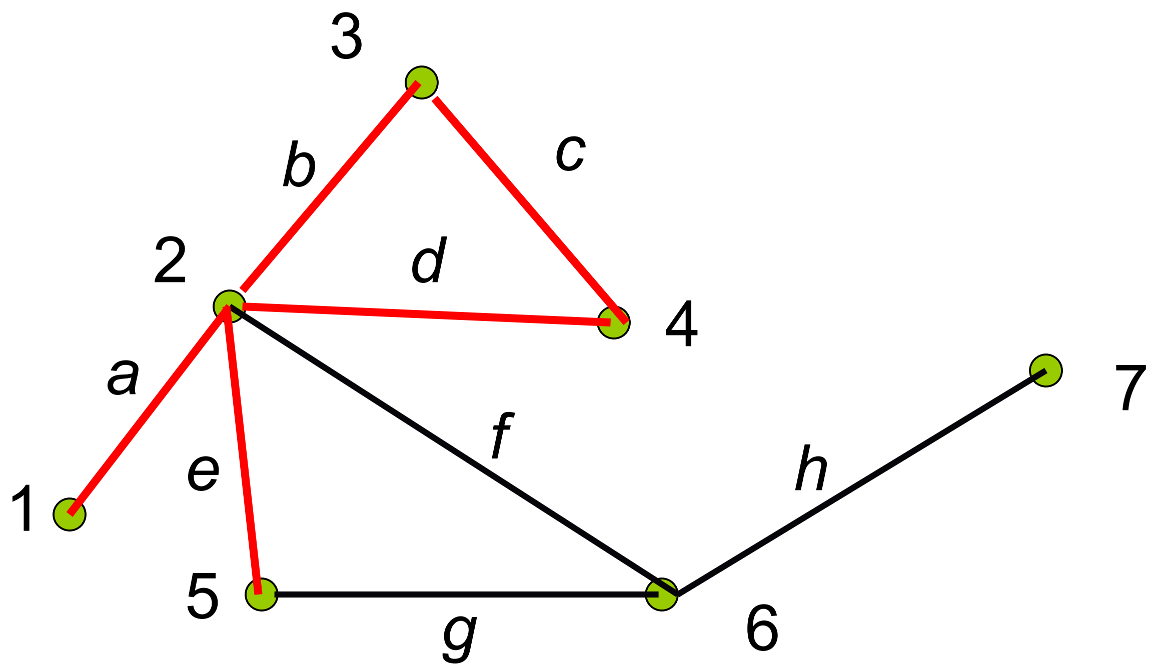
example

- (1, *a*, 2, *b*, 3, *c*, 4, *d*, 2) is a walk of length 4 from vertex 1 to vertex 2.



example

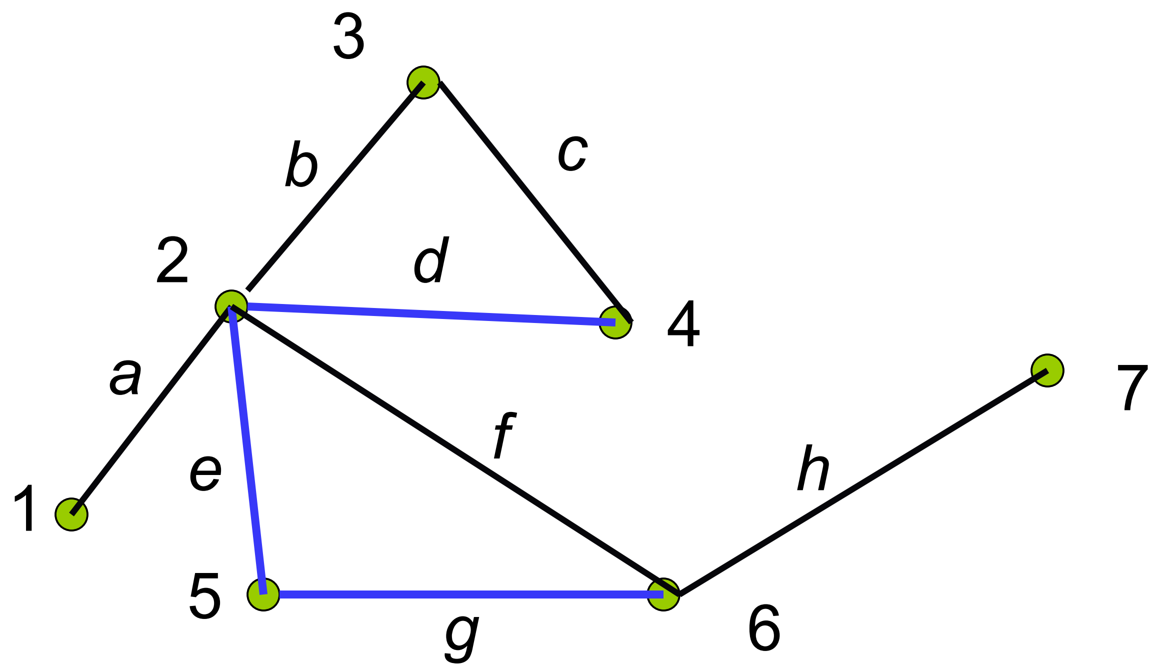
- (1, a, 2, b, 3, c, 4, d, 2, e, 5) is a trail.



Note:
Trail: No repeated edge (can repeat vertex).

example

- (6, g, 5, e, 2, d, 4) is a path.

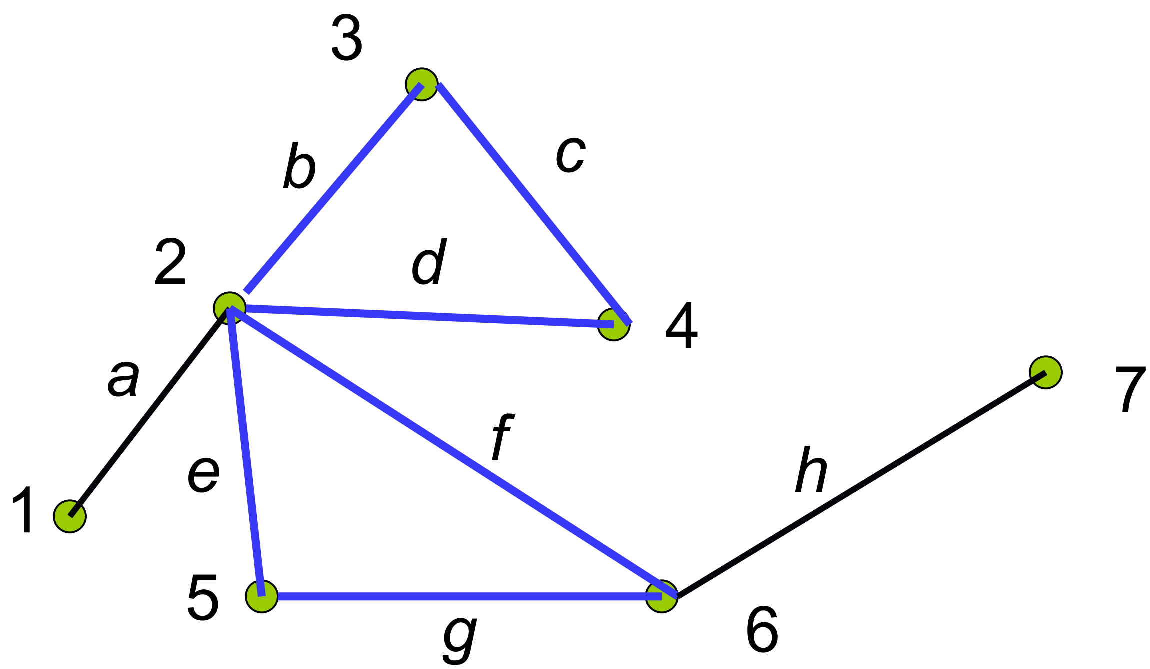


Note:

Path: No repeated vertex and edge.

example

- (2, *f*, 6, *g*, 5, *e*, 2, *d*, 4, *c*, 3, *b*, 2) is a circuit/cycle.

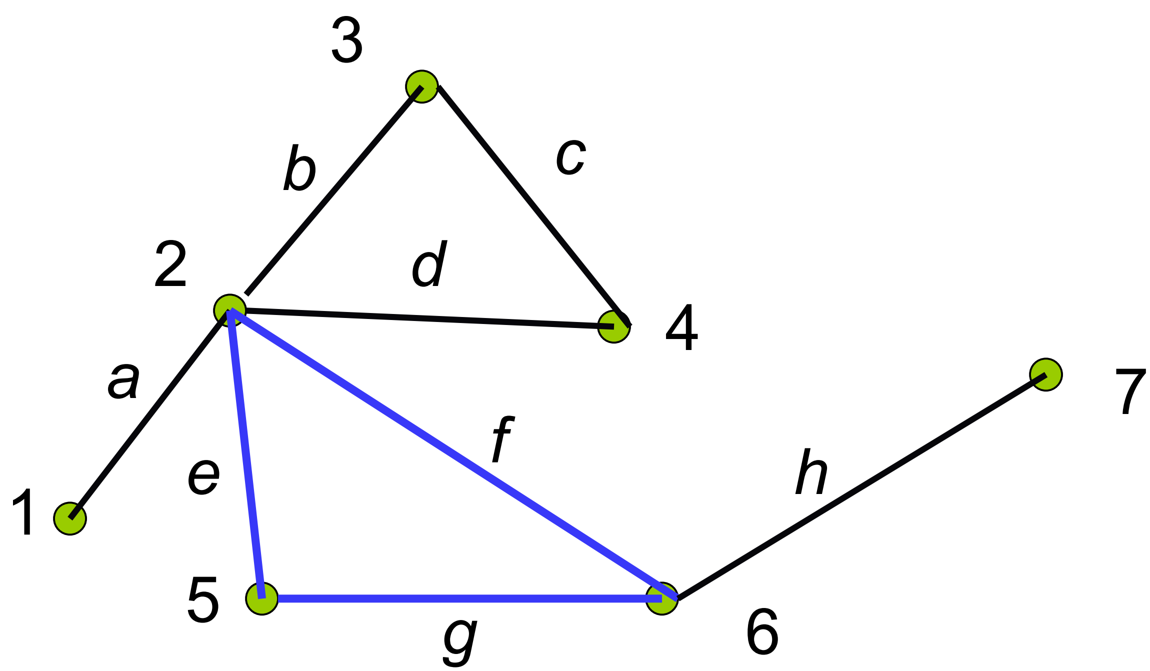


Note: circuit → start and end at same vertex, no repeated edge.

prepared by Razana Alwee

example

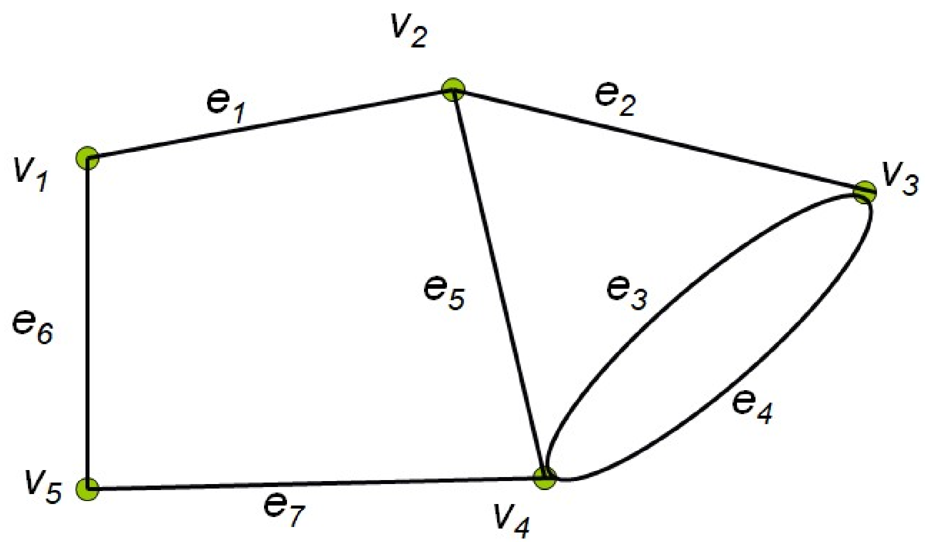
- (5, *g*, 6, *f*, 2, *e*, 5) is a simple circuit.



Note: Simple circuit →start and end at same vertex, no repeated edge or vertex except for the start and end vertex.

exercise

- Tell whether the following is either a walk, trail, path, circuit, simple circuit, closed walk or none of these.
  - (v_1, e_1, v_2)
  - $(v_2, e_2, v_3, e_3, v_4, e_4, v_3)$
  - $(v_4, e_7, v_5, e_6, v_1, e_1, v_2, e_2, v_3, e_3, v_4)$
  - $(v_4, e_4, v_3, e_3, v_4, e_5, v_2, e_1, v_1, e_6, v_5, e_7, v_4)$



Euler Trail & Circuit

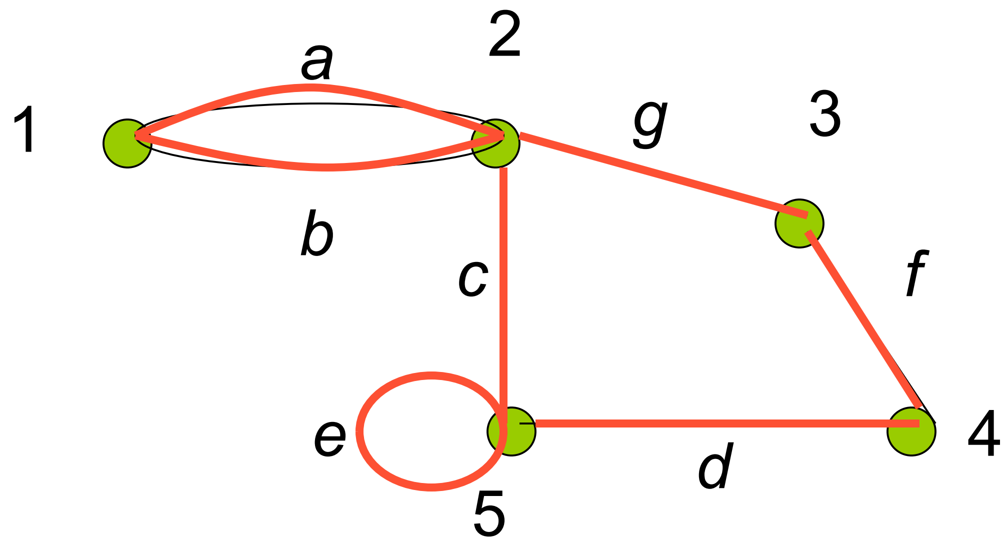
Euler Circuits

- A circuit in a graph that includes all the edges of the graph is called an Euler circuit.

- Let G be a graph. An Euler circuit for G is **a circuit that contains every vertex and every edges of G**. That is, an Euler circuit for G is a sequence of adjacent vertices and edges in G that has at least one edges, starts and ends at the same vertex, **uses every vertex of G at least once**, and **uses every edge of G exactly once**.

prepared by Razana Alwee

example



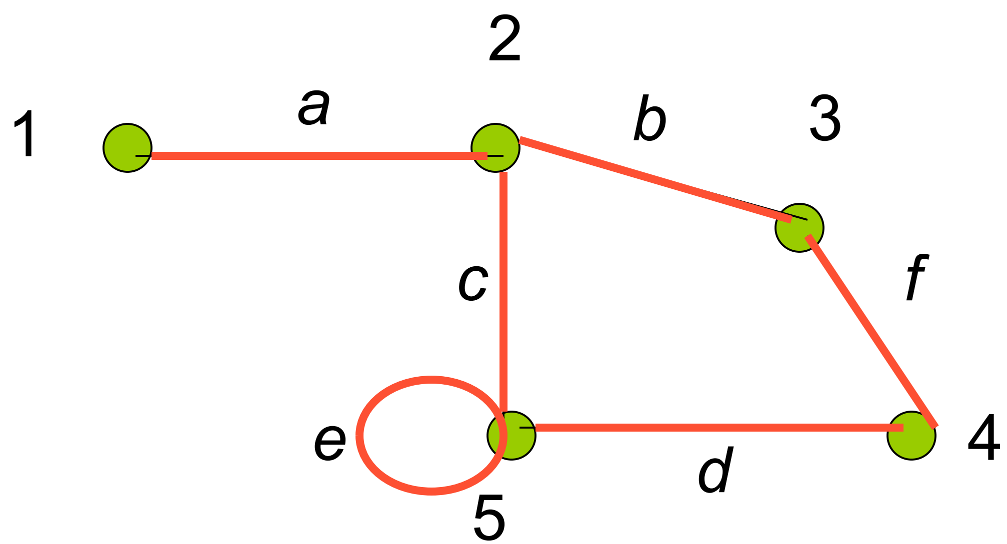
(**1**, *a*, 2, *c*, 5, *e*, 5, *d*, 4, *f*, 3, *g*, 2, *b*, **1**)
is an Euler circuit

Euler Trail

- A trail from *v* to *w* ($v \neq w$) with no repeated edges is called an Euler trail if it **contains all the edges and all the vertices**.

- Let G be a graph, and let v and w be two distinct vertices of G. An Euler trail from v to w is a sequence of adjacent vertices and edges that starts at v and ends at w, passes through every **vertex** of G at least once, and traverses every **edge** of G exactly once.

example



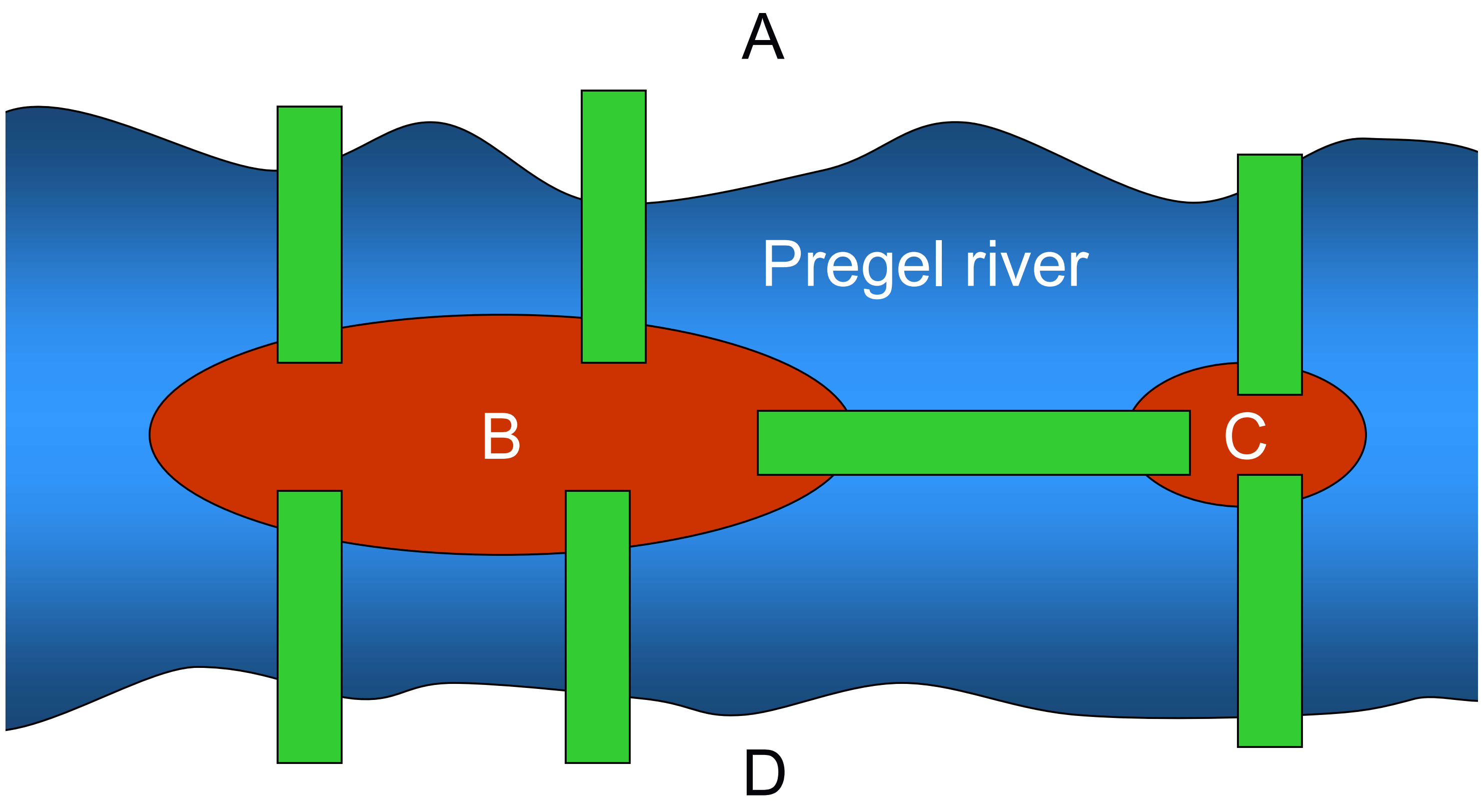
(1, *a*, 2, *c*, 5, *e*, 5, *d*, 4, *f*, 3, *b*, 2)
is an Euler trail

Theorem

- If *G* is a connected graph and **every vertex has even degree**, then *G* has an **Euler circuit**.
- A graph has an **Euler trail** from *v* to *w* ($v \neq w$) if and only if it is connected and ***v* and *w* are the only vertices having odd degree**.

prepared by Razana Alwee

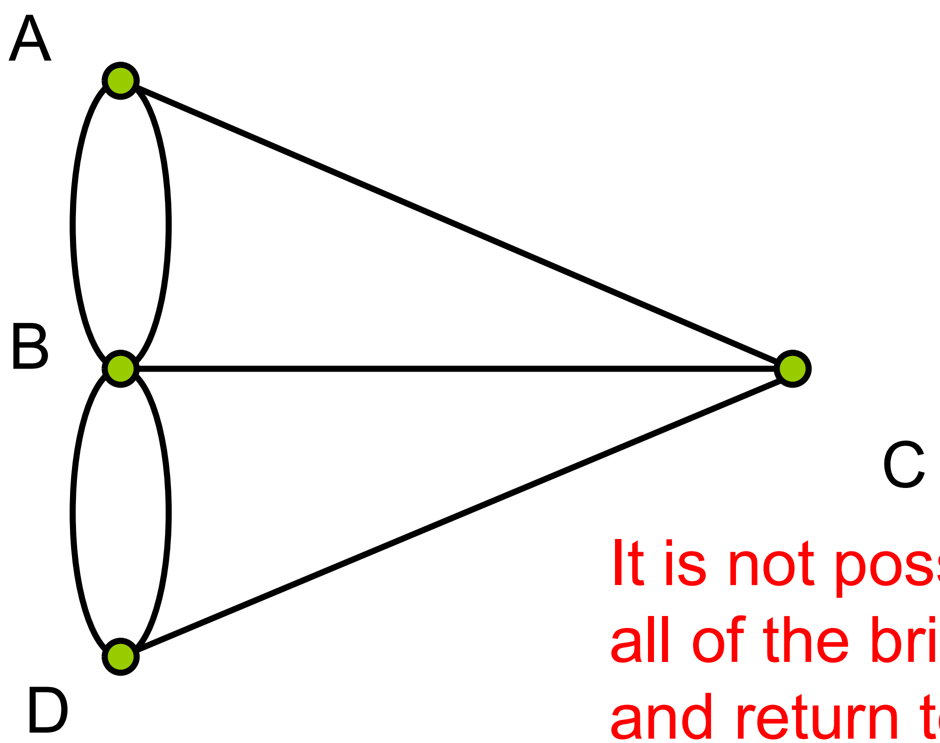
Königsberg Bridge Problem



Starting at one land area, is it possible to walk across all of the bridges exactly once and return to the starting land area?

Königsberg Bridge Problem

- Graph of the Königsberg Bridge Problem

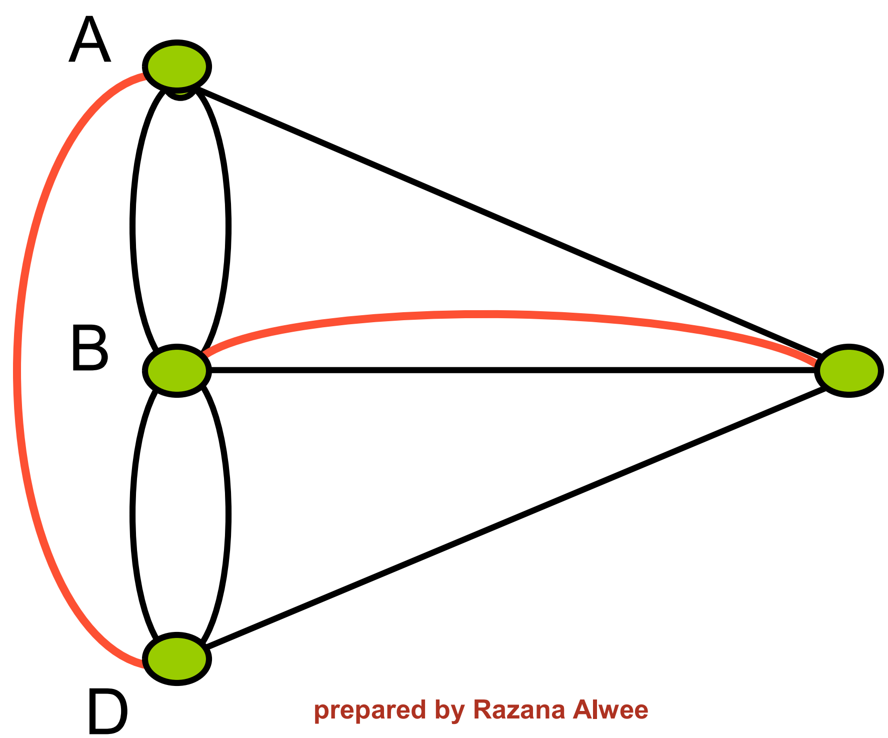


no Euler circuit

It is not possible to walk across all of the bridges exactly once and return to the starting land area

Königsberg Bridge Problem

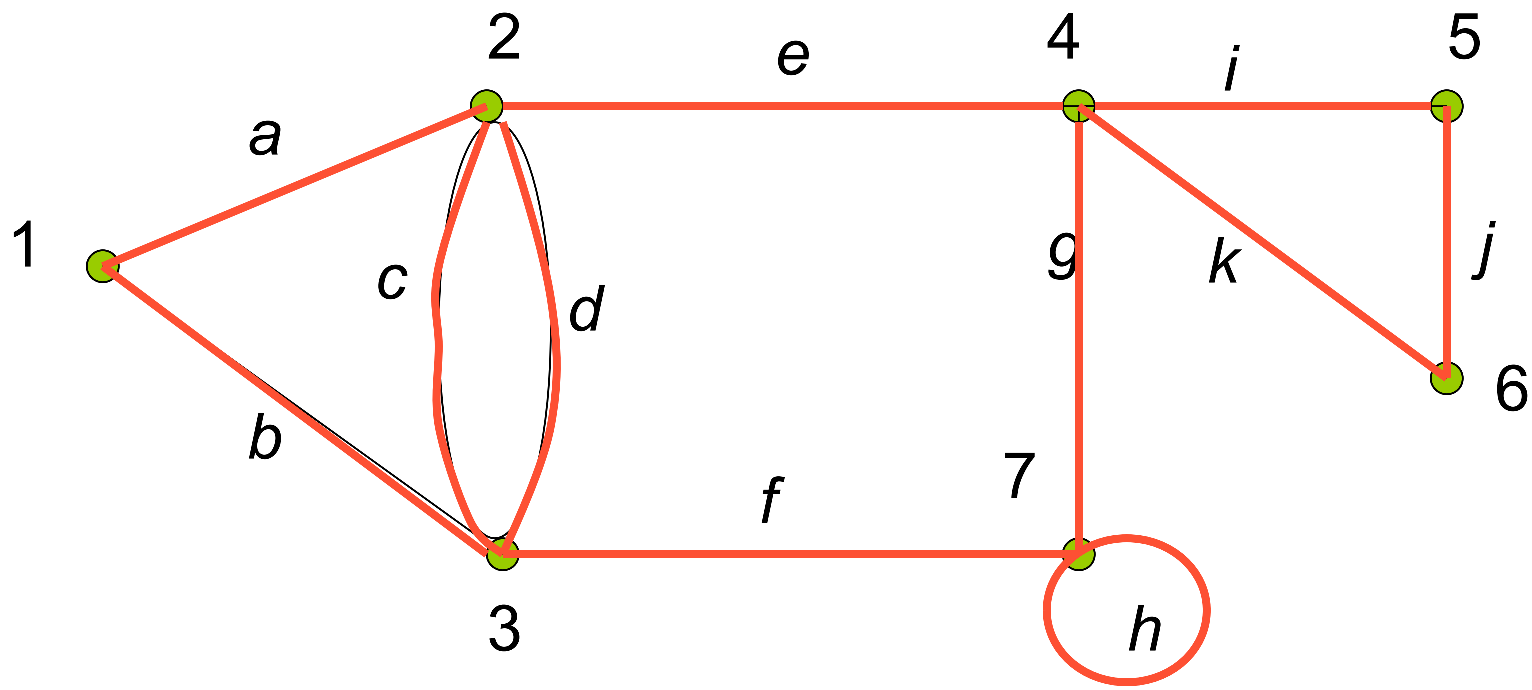
- Since 1736, two additional bridges have been constructed on the Pregel river.



prepared by Razana Alwee

example

Vertex	1	2	3	4	5	6	7
Degree	2	4	4	4	2	2	4

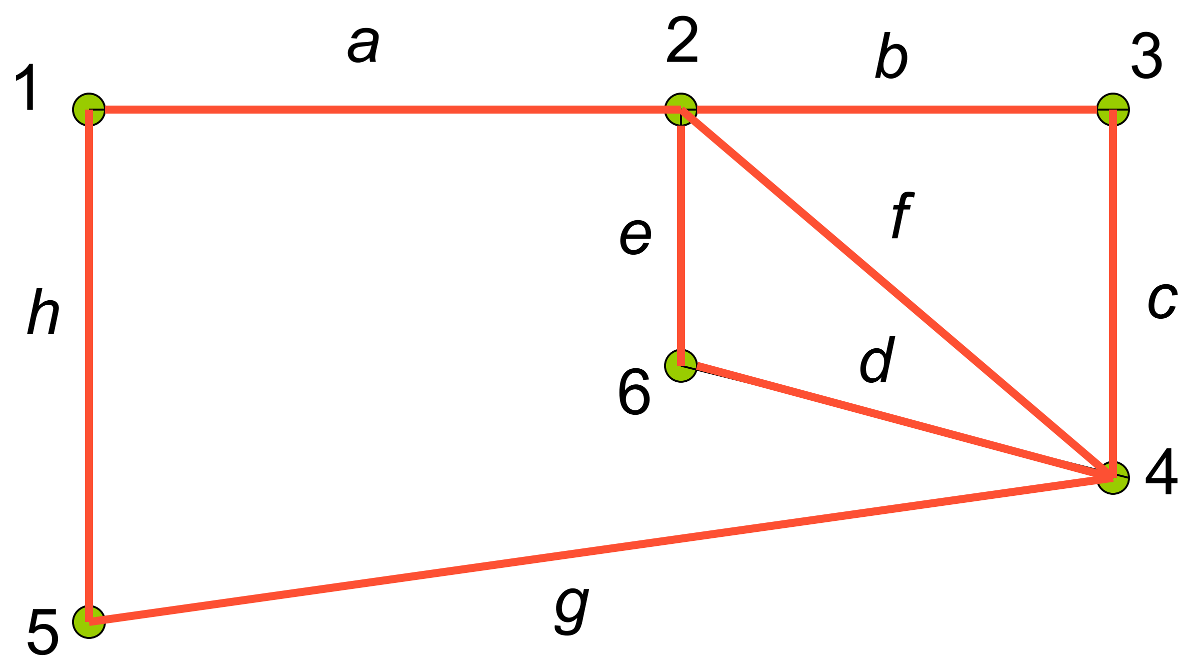


This graph has an Euler circuit

prepared by Razana Alwee

example

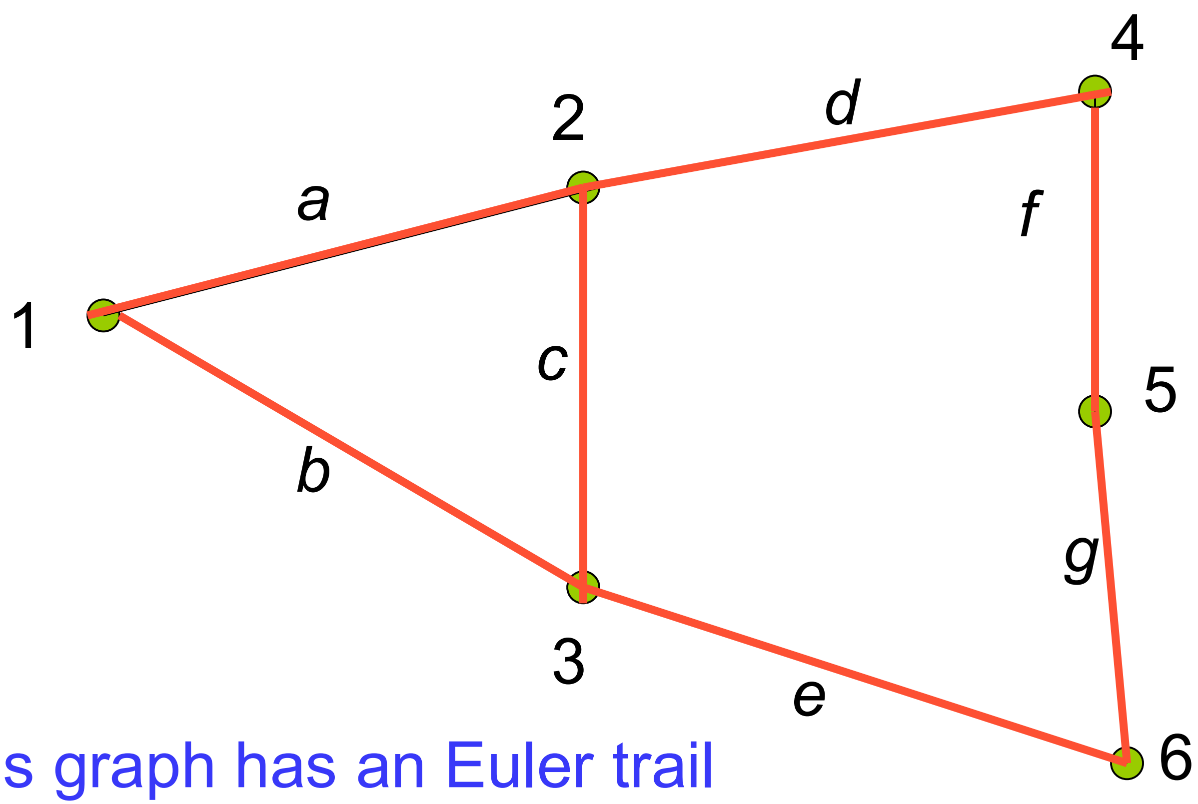
Vertex	1	2	3	4	5	6
Degree	2	4	2	4	2	2



This graph has an Euler circuit

example

Vertex	1	2	3	4	5	6
Degree	2	3	3	2	2	2

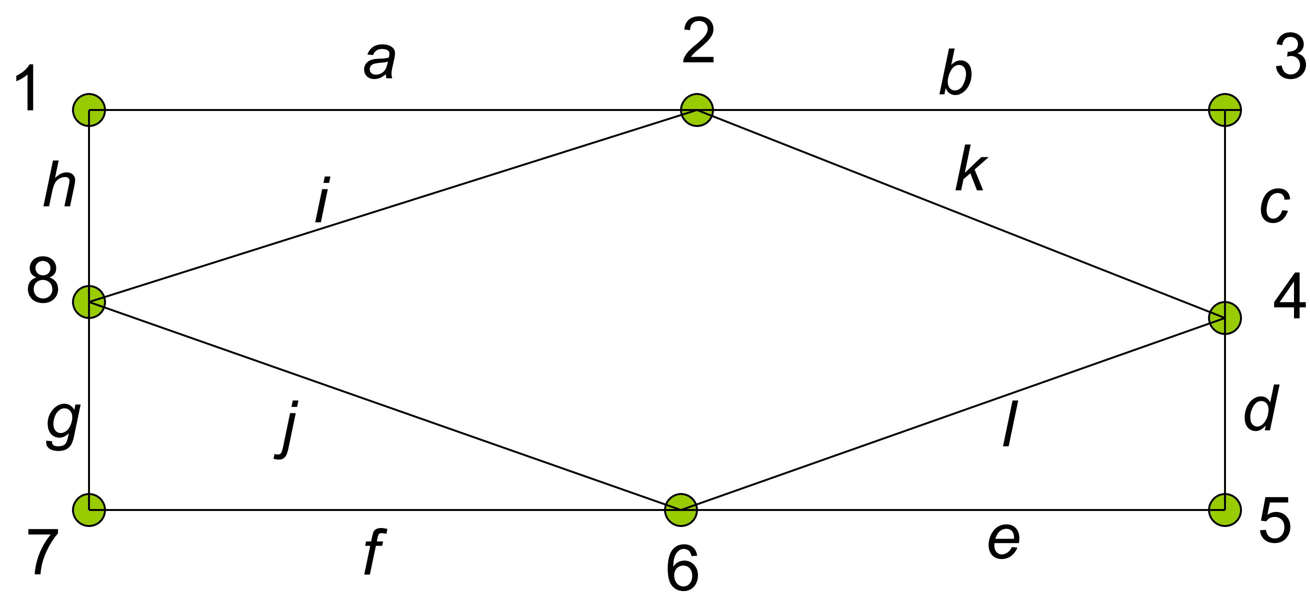


This graph has an Euler trail

prepared by Razana Alwee

exercise

- Decide whether the graph has an Euler circuit. If the graph has an Euler circuit, exhibit one.

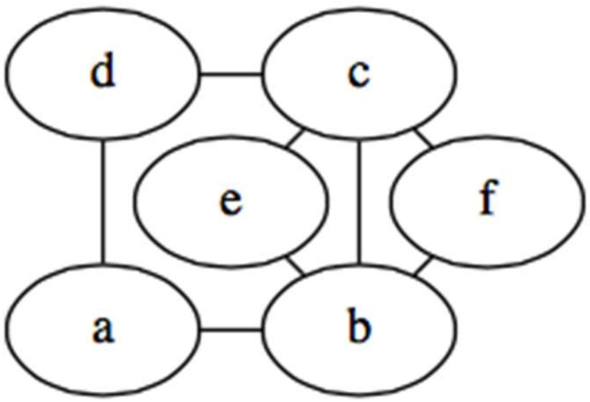


prepared by Razana Alwee

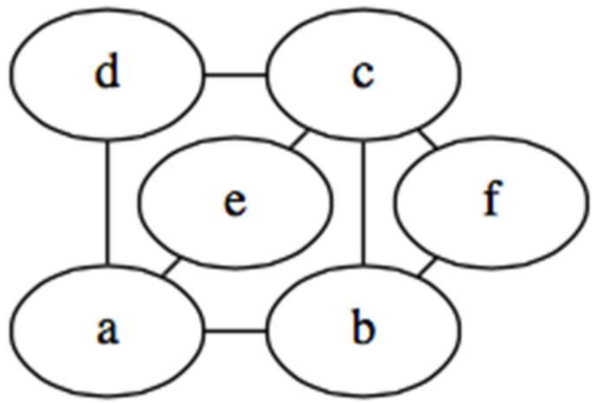
exercise

Q: Which of the following graphs has Euler circuit? Justify your answer.

G_1



G_2



Exercise
Past Year 2015/2016

Determine whether the graph in Figure 3 has an Euler cycle or Euler path. If the graph has an Euler cycle or Euler path, exhibit one; otherwise, give an argument that shows there is no Euler path. (4 marks)

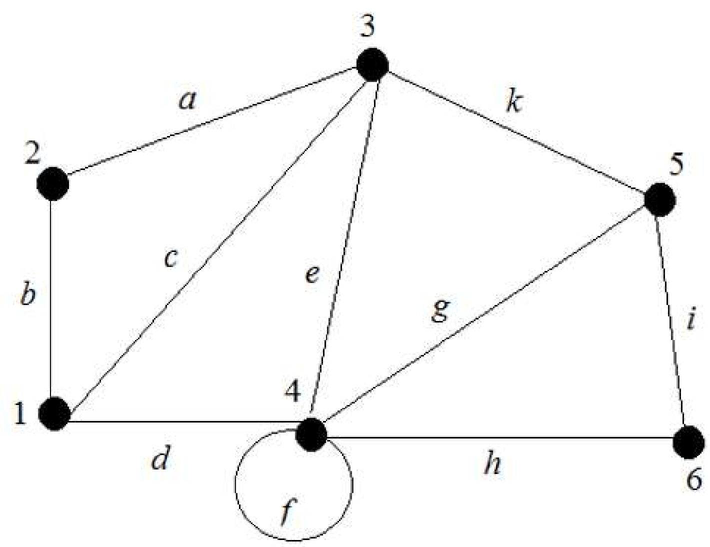


Figure 3

Hamilton Circuits

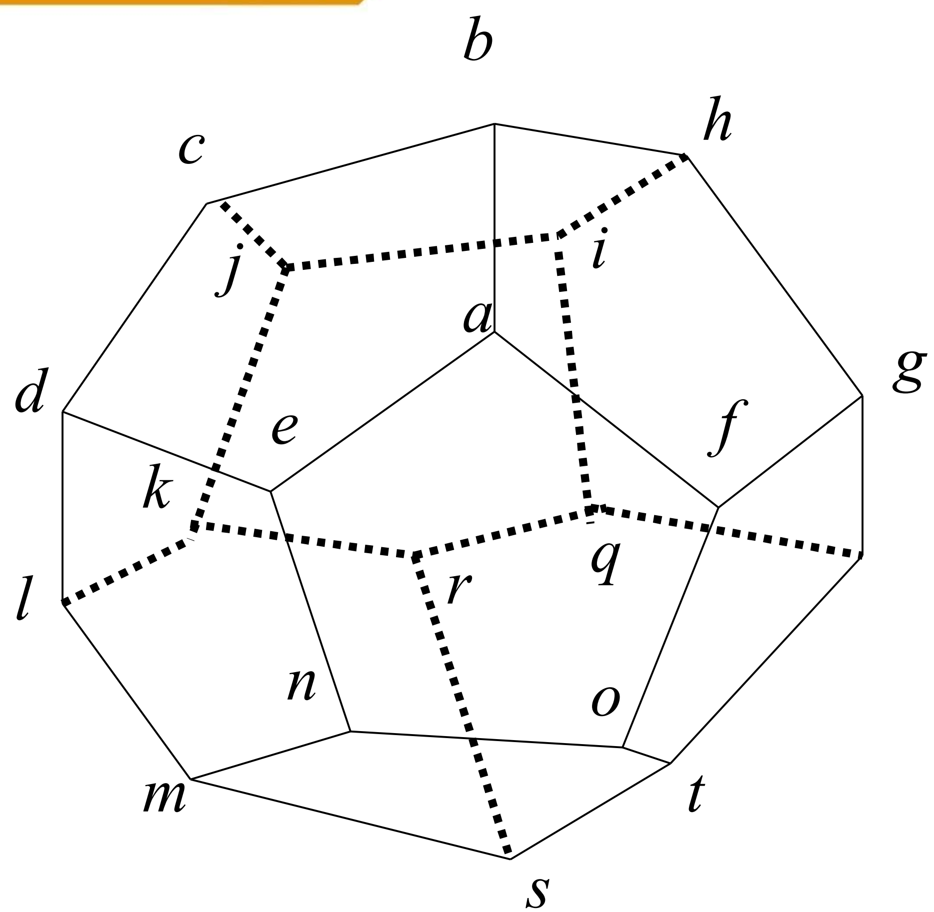
Hamiltonian Circuit

- A circuit in a graph *G* is called a Hamiltonian circuit if it contains each vertex of *G*.

- Given a graph G, a Hamiltonian circuit for G is **a simple circuit that includes every vertex of G** (but doesn't need to include all edges). That is, a Hamiltonian circuit for G is a sequence of adjacent vertices and distinct edges in which every vertex of G appears exactly once, except for the first and the last, which are the same.

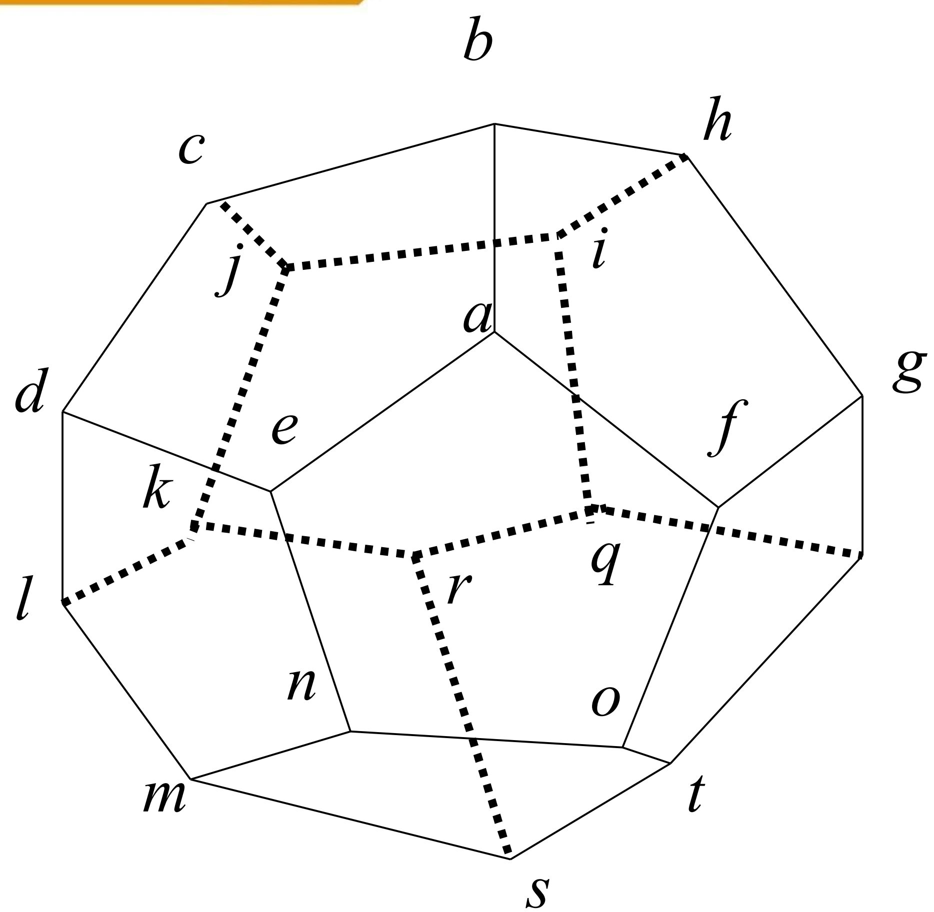
prepared by Razana Alwee

Around the world game



Sir William Rowan Hamilton marketed a puzzle in the mid-1800s in the form of dedocahedron

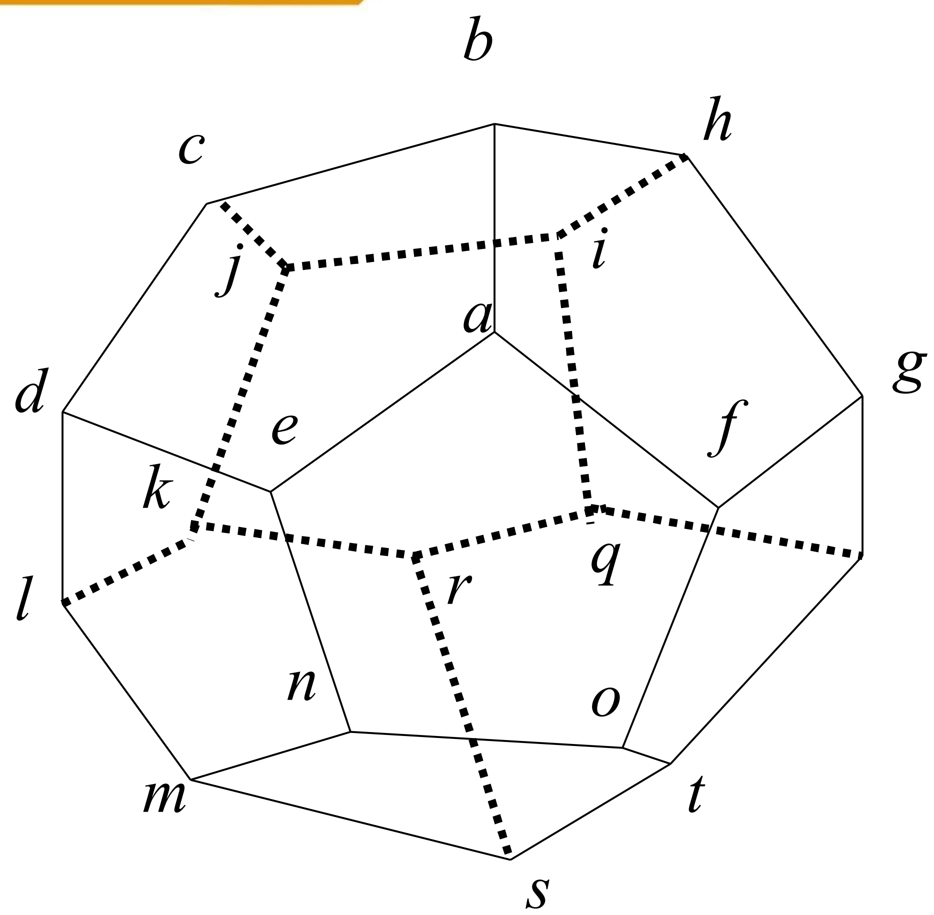
Around the world game



Each corner bore the name of a city

prepared by Razana Alwee

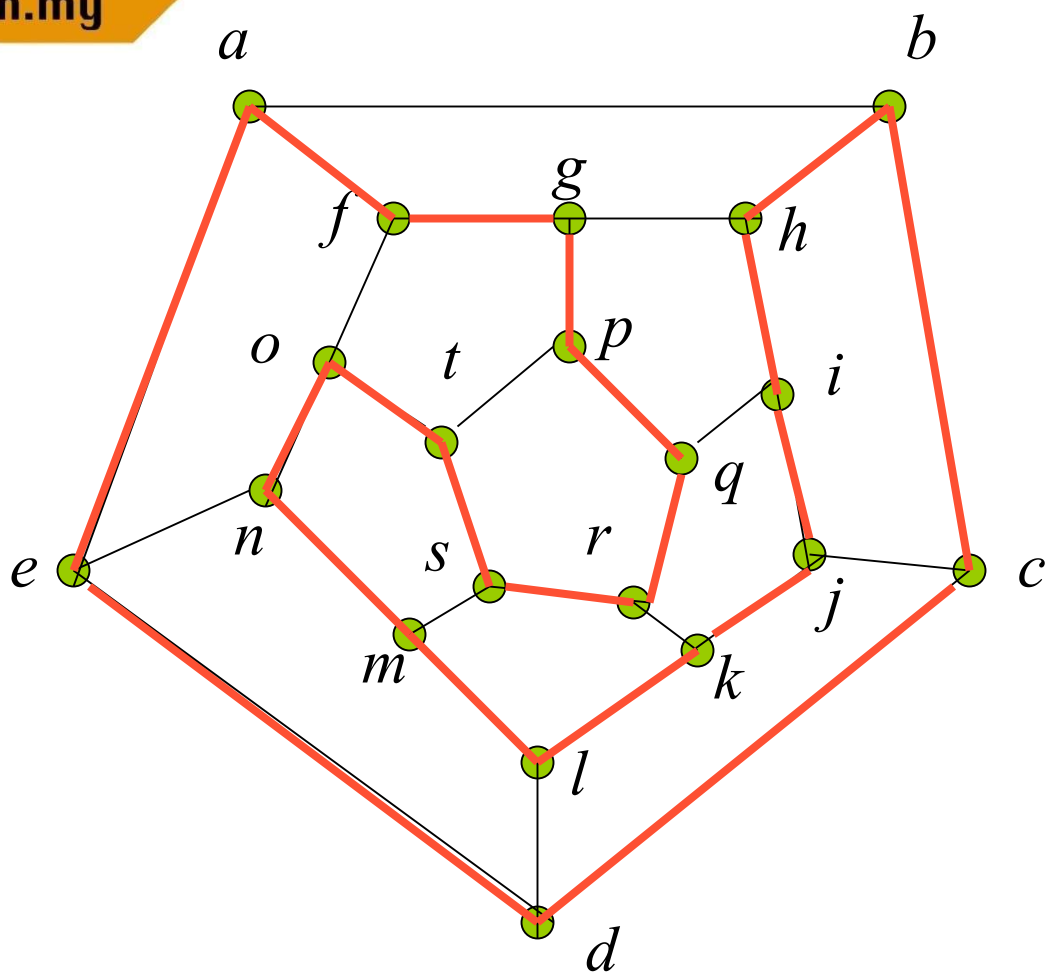
Around the world game



The problem was to start at any city, travel along the edges, visit each city exactly one time and return to the initial city

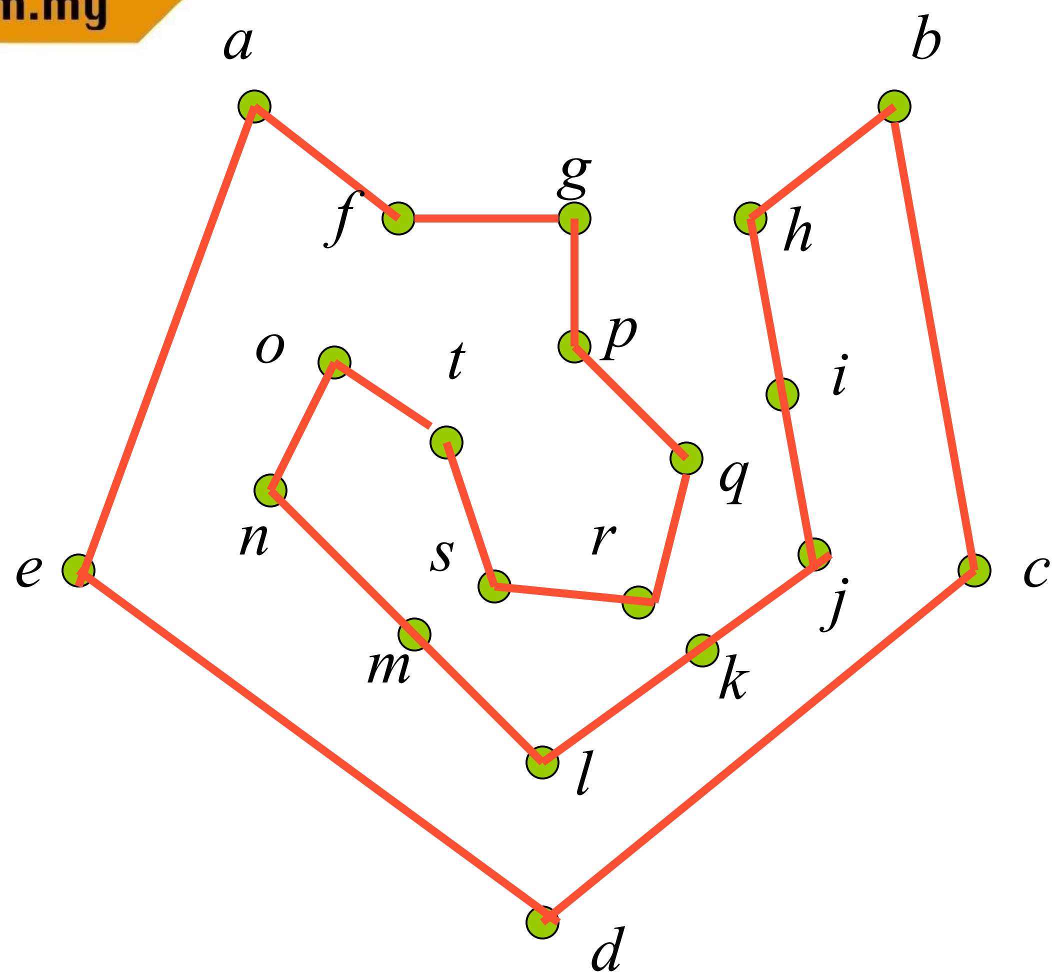
prepared by Razana Alwee

The graph



prepared by Razana Alwee

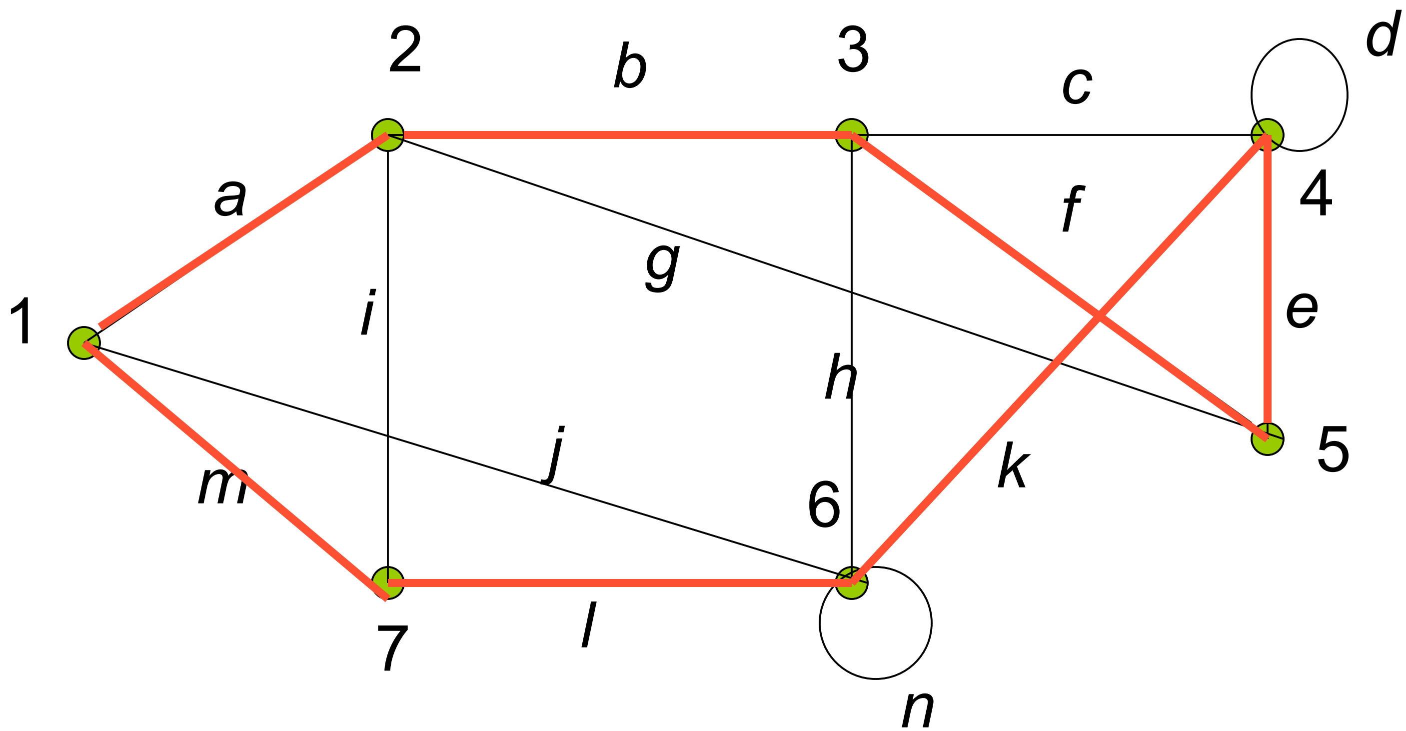
Hamiltonian Circuit



a-f-g-p-q-r-s-t-o-n-m-l-k-j-i-h-b-c-d-e-a

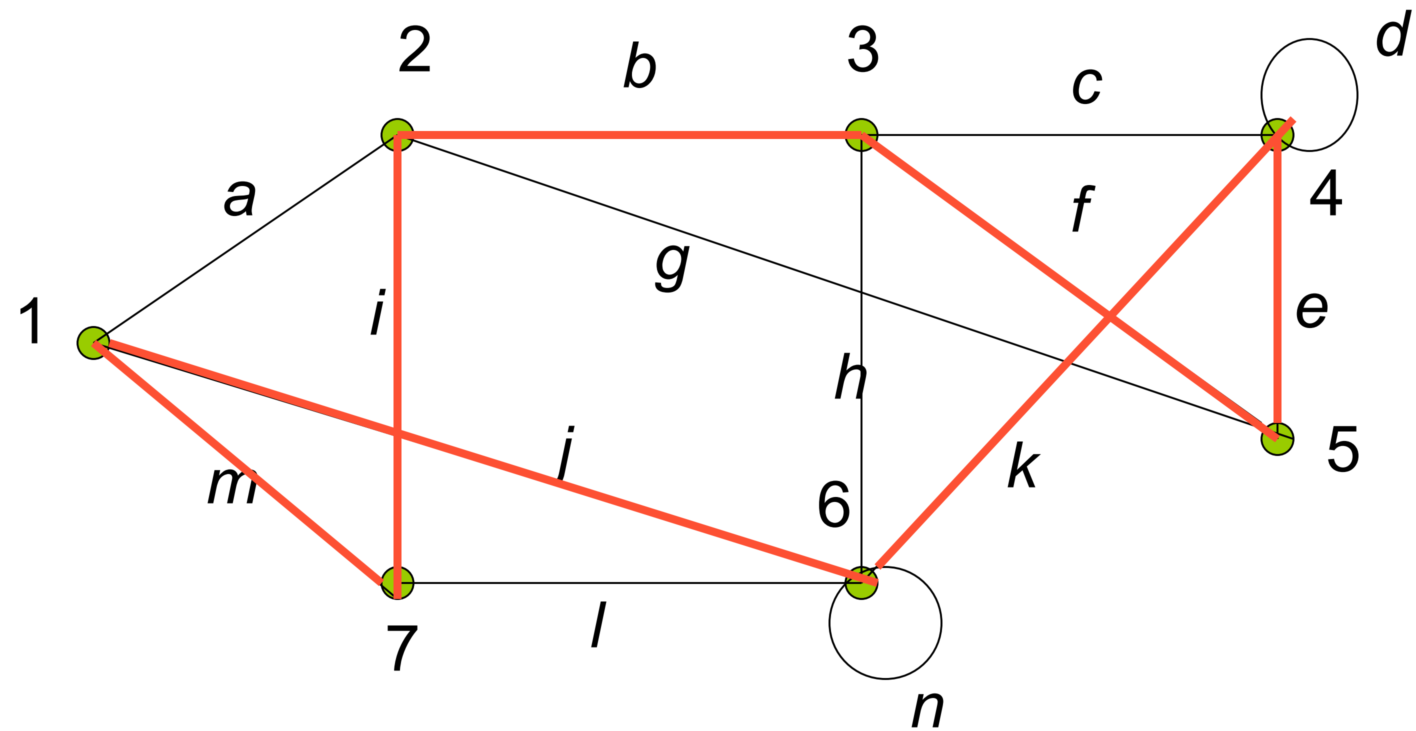
example

This graph has a Hamiltonian circuit



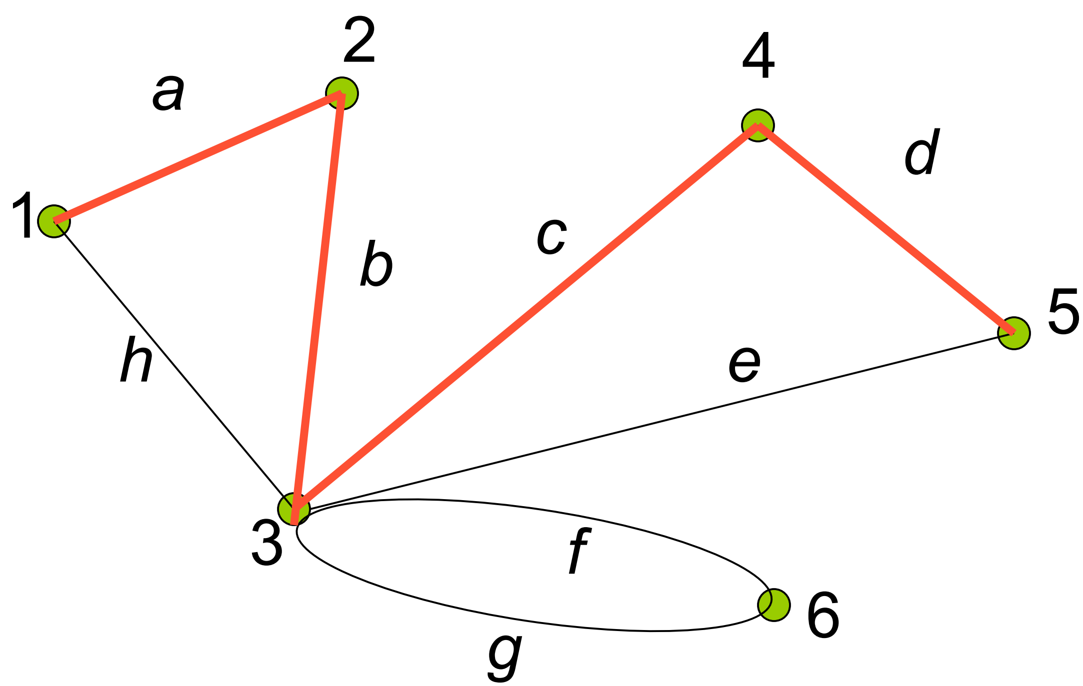
1-a-2-b-3-f-5-e-4-k-6-l-7-m-1

example



1-j-6-k-4-e-5-f-3-b-2-i-7-m-1

example

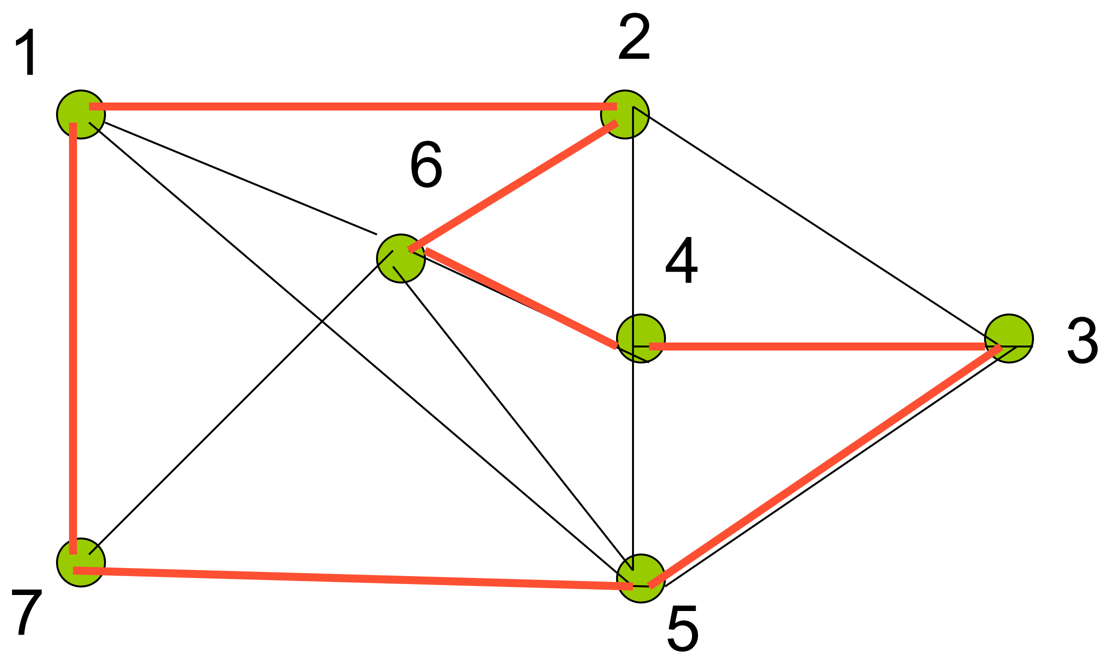


no Hamiltonion circuit

prepared by Razana Alwee

example

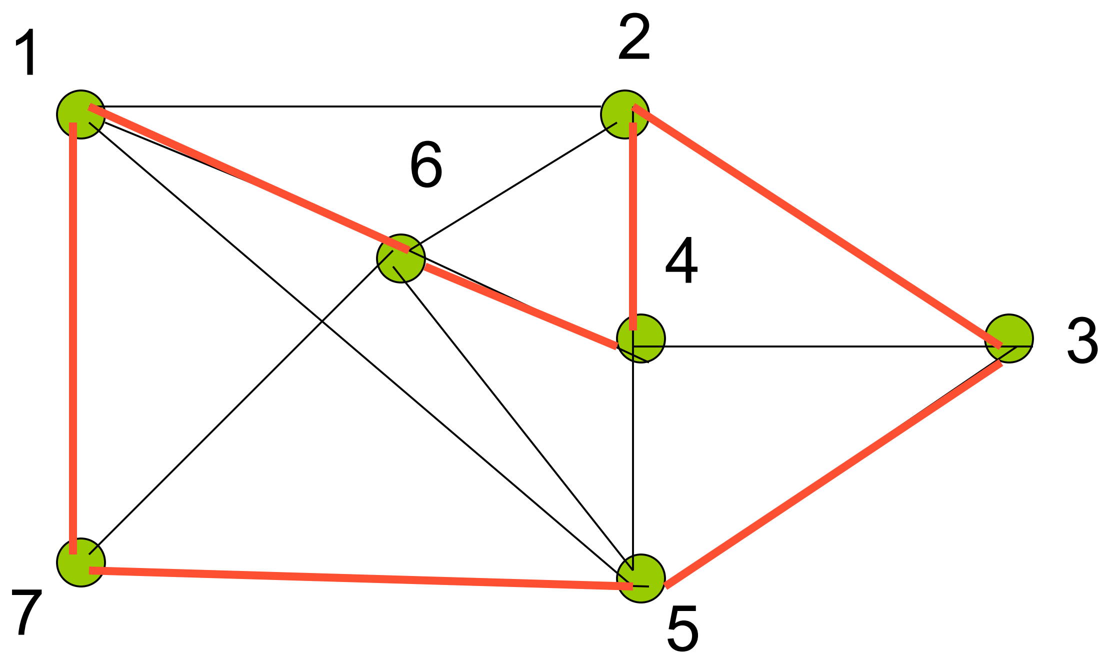
This graph has a Hamiltonian circuit



1-2-6-4-3-5-7-1

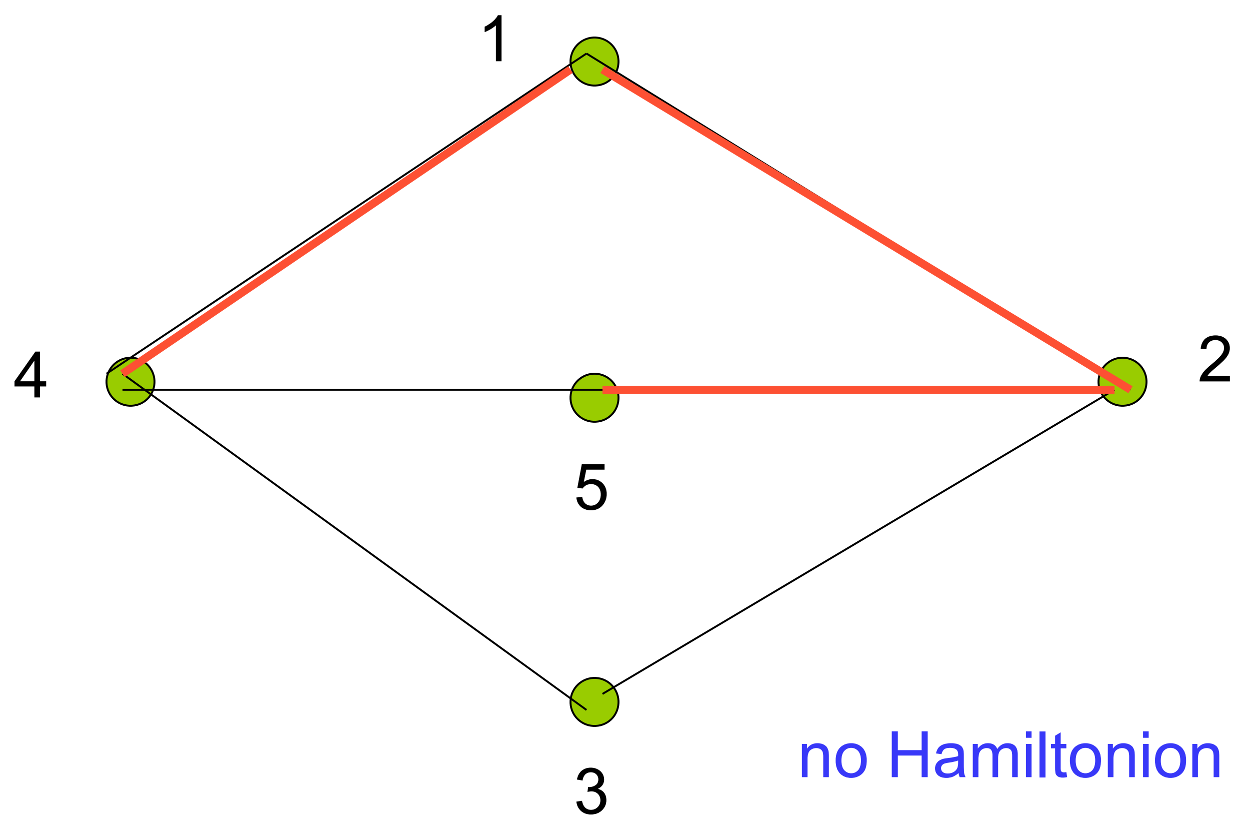
prepared by Razana Alwee

example



1-6-4-2-3-5-7-1

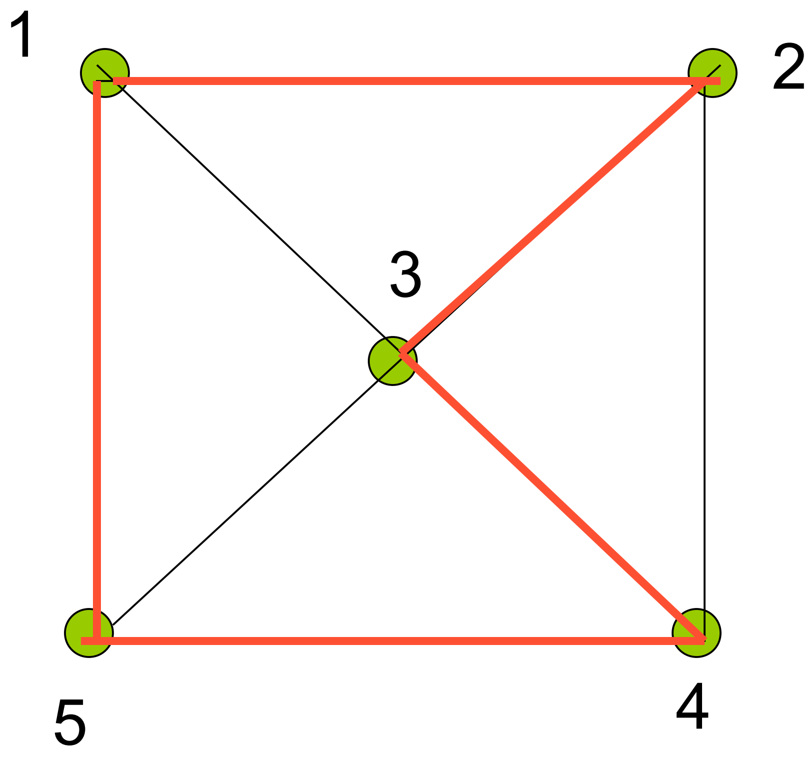
example



no Hamiltonion circuit

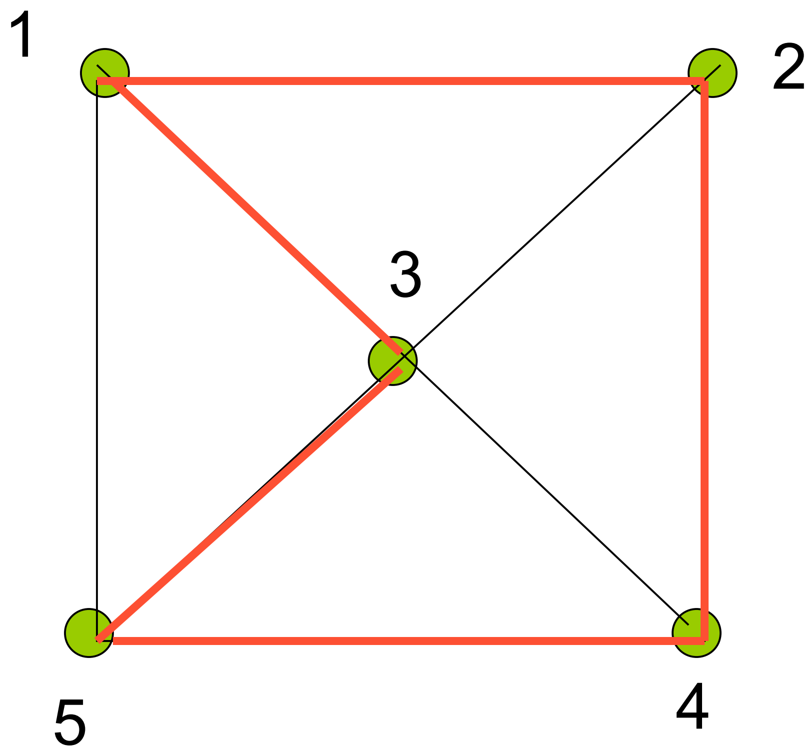
example

This graph has a Hamiltonian circuit



1-2-3-4-5-1

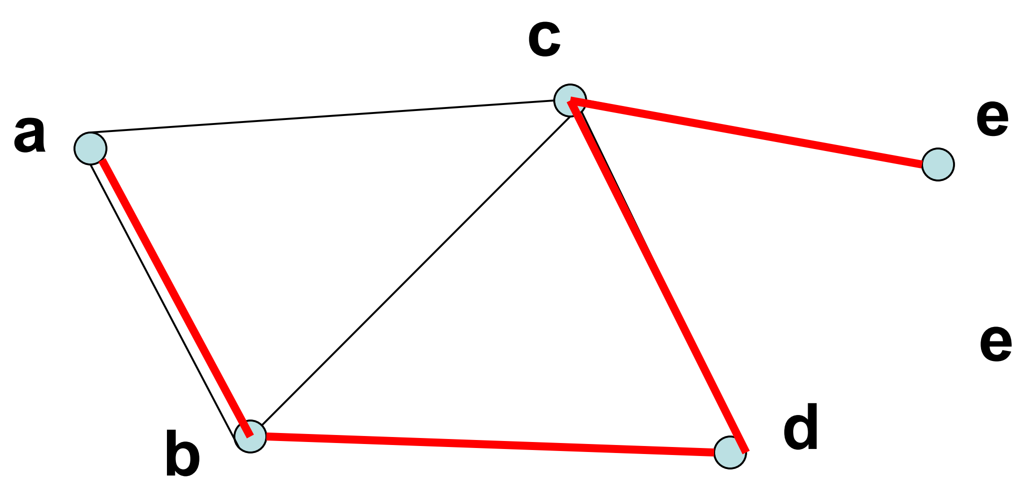
example



3-5-4-2-1-3

Hamiltonian Path

- A path in a graph *G* is called a Hamiltonian path if it contains each vertex of *G.*
- Example:

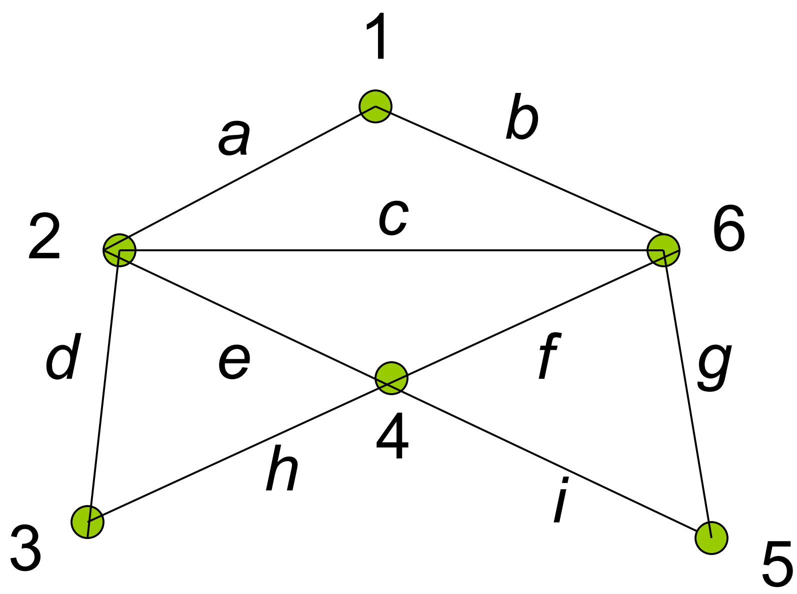


e-c-d-b-a

prepared by Razana Alwee

exercise

- Find a Hamiltonian circuit in this graph.



prepared by Razana Alwee

Exercise
Past Year 2015/2016

Determine whether the graph in Figure 4 has an Hamiltonian cycle. If yes, exhibit one.

(3 marks)

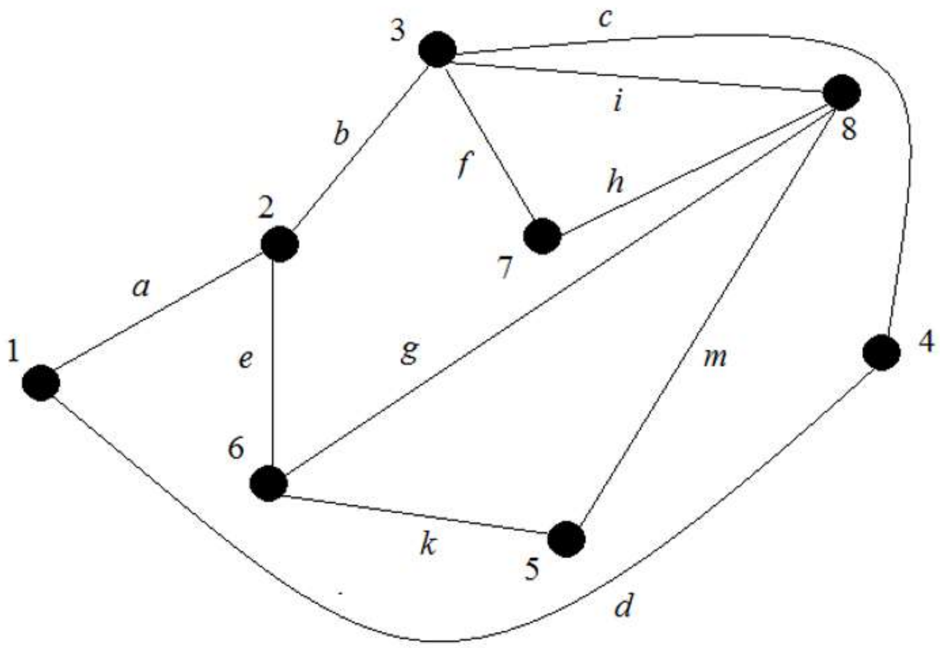


Figure 4