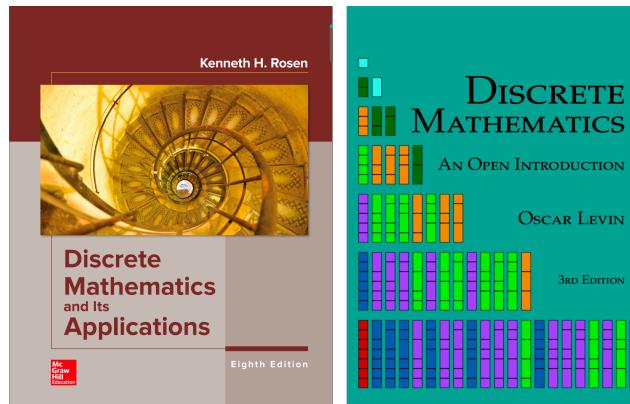




Vietnam National University of HCMC  
International University  
School of Computer Science and Engineering



## Session 12: Graph Theory

Assoc. Prof. Dr. Nguyen Van Sinh

[nvsinh@hcmiu.edu.vn](mailto:nvsinh@hcmiu.edu.vn)



# **Part 1: Graphs and Graph Models: Basics, Concepts and Terminologies**

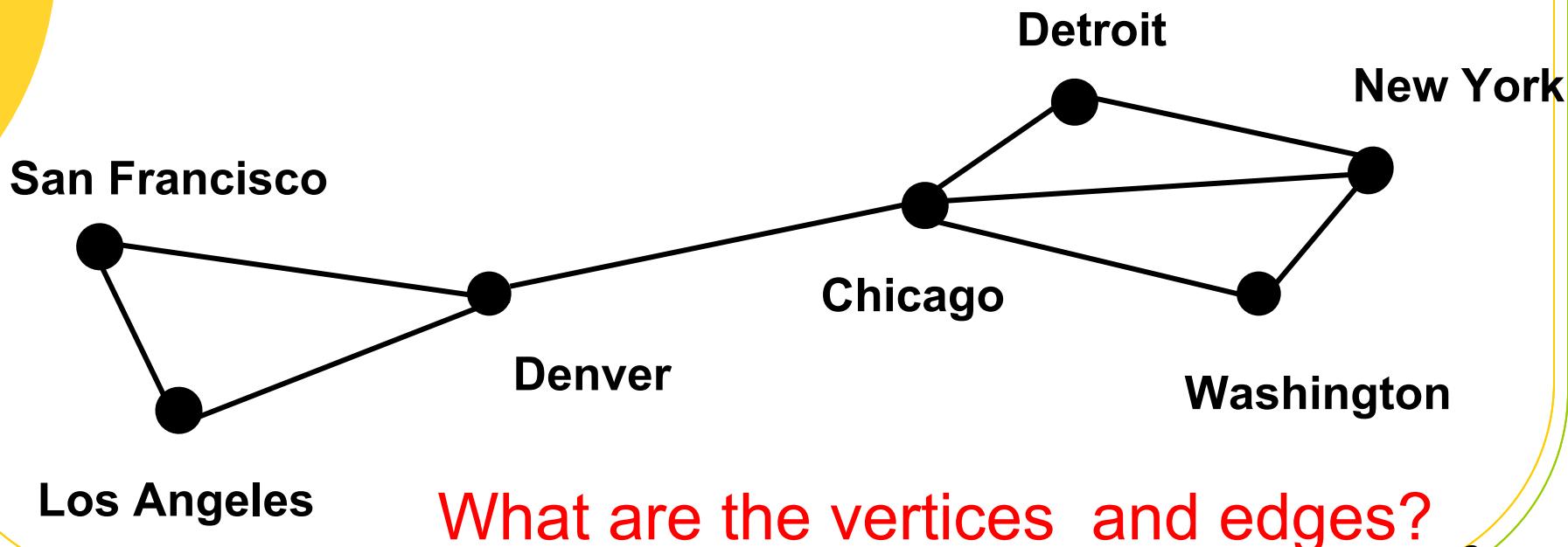
---

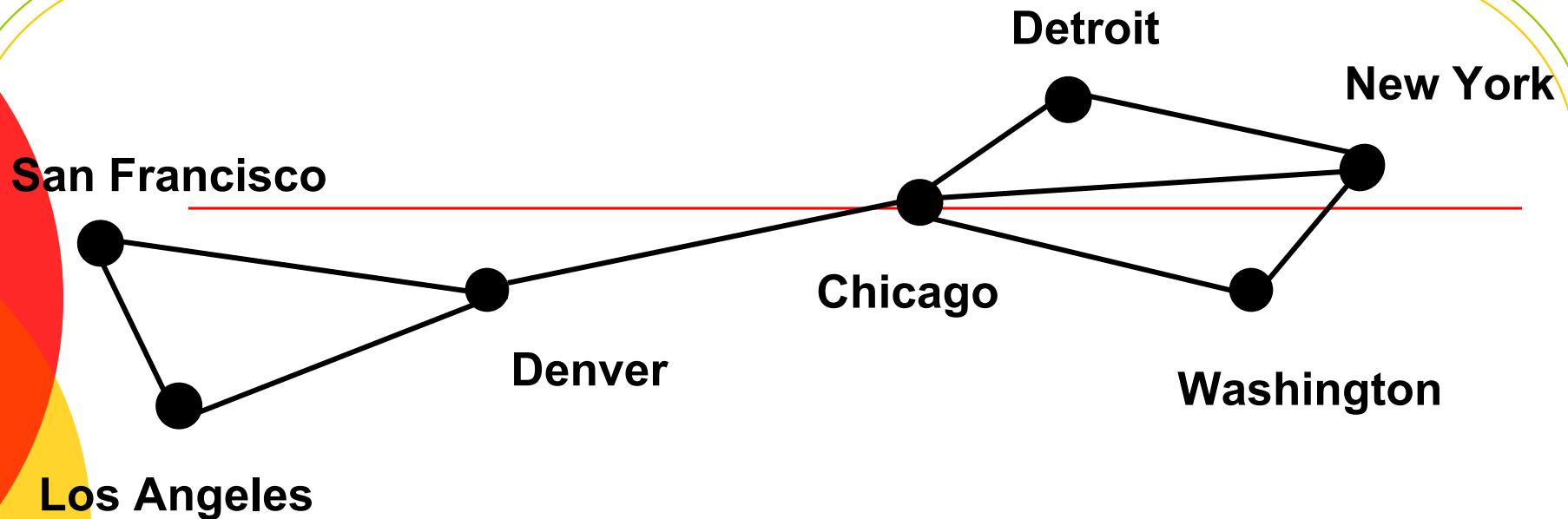
1. Types of Graphs
2. Basic Terminology
3. Special Types of Graphs
4. New Graphs from Old Graphs

# 1. Type of Graphs

## Simple Graph

**Definition 1.** A *simple graph*  $G = (V, E)$  consists of  $V$ , a nonempty set of *vertices*, and  $E$ , a set of unordered pairs of distinct elements of  $V$  called *edges*.

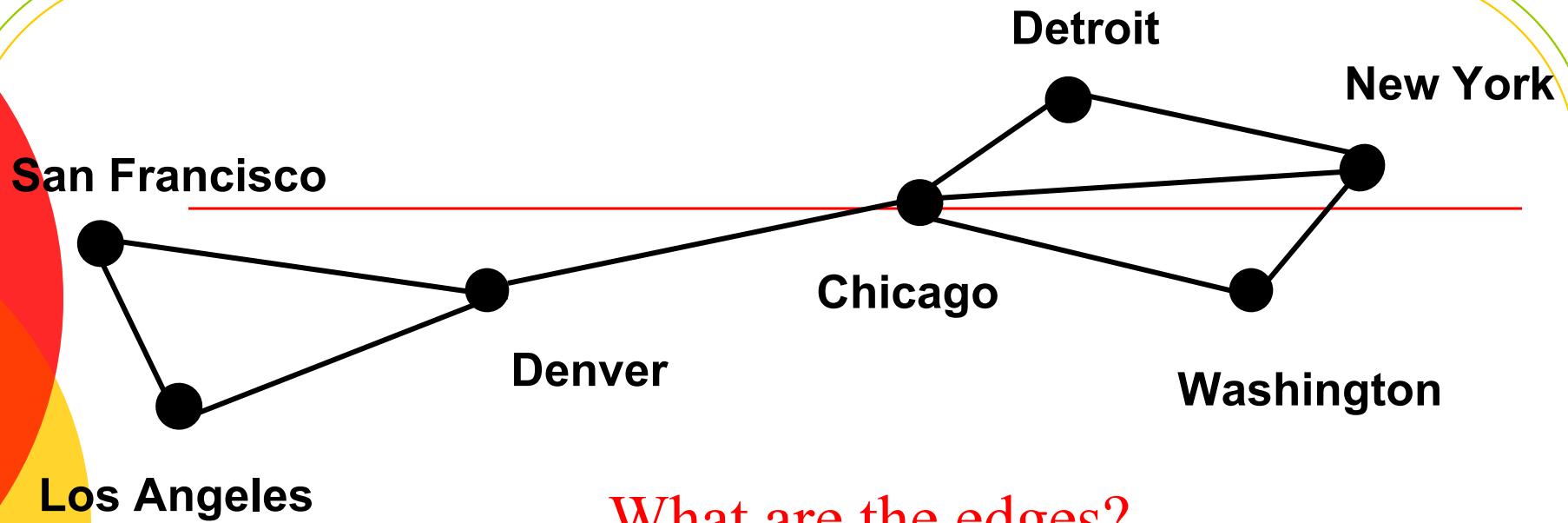




What are the vertices?

## SET OF VERTICES

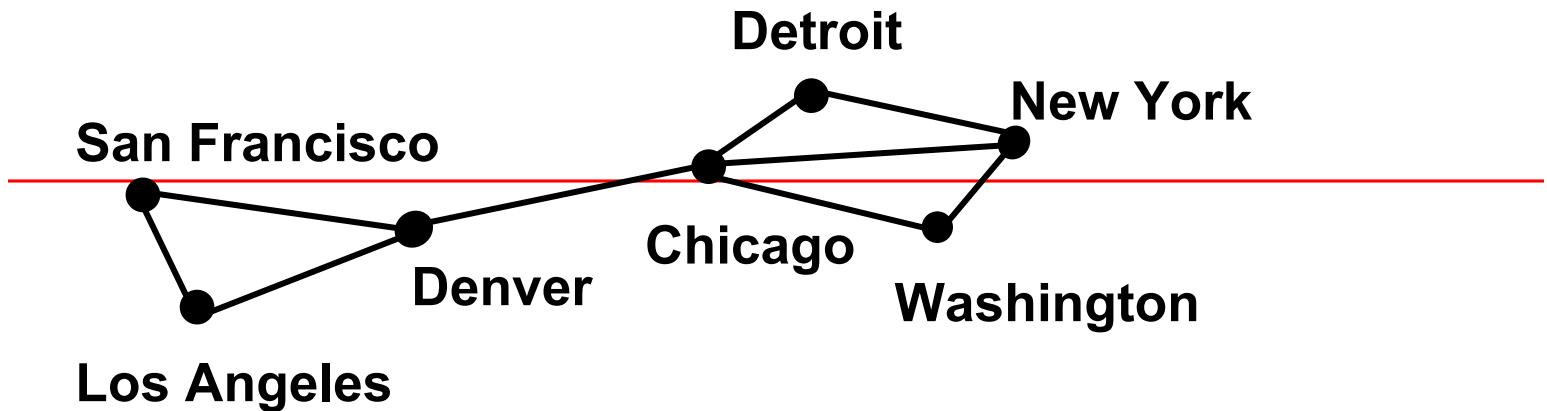
$V = \{\text{San Francisco}, \text{Los Angeles}, \text{Denver}, \text{Chicago}, \text{Detroit}, \text{New York}, \text{Washington}\}$



What are the edges?

## SET OF EDGES

$E = \{ \{S F, L A\}, \{S F, D e n v e r\}, \{L A, D e n v e r\},$   
 $\{D e n v e r, C h i c a g o\}, \{C h i c a g o, D e t r o i t\}, \{D e t r o i t, N Y\},$   
 $\{N Y, W a s h i n g t o n\}, \{C h i c a g o, W a s h i n g t o n\},$   
 $\{C h i c a g o, N Y\} \}$



The network is made up of computers and telephone lines between computers.

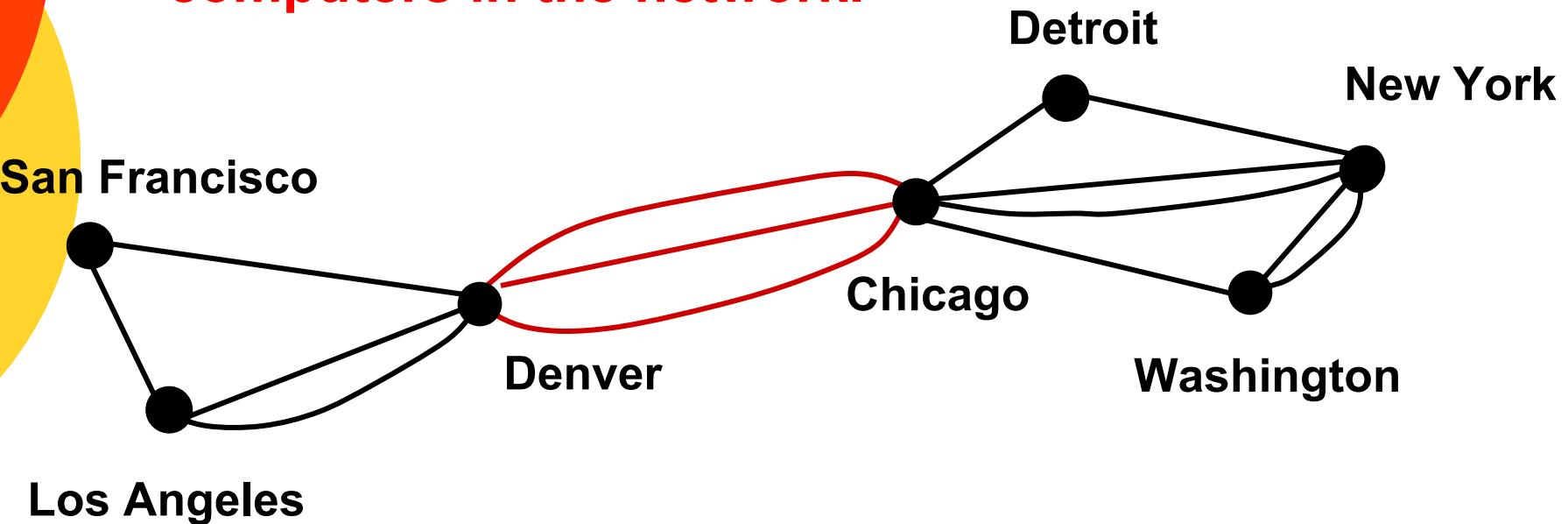
There is at most 1 telephone line between 2 computers in the network. Each line operates in both directions.

No computer has a telephone line to itself.

There are undirected edges, each of which connects two distinct vertices, and no two edges connect the same pair of vertices.

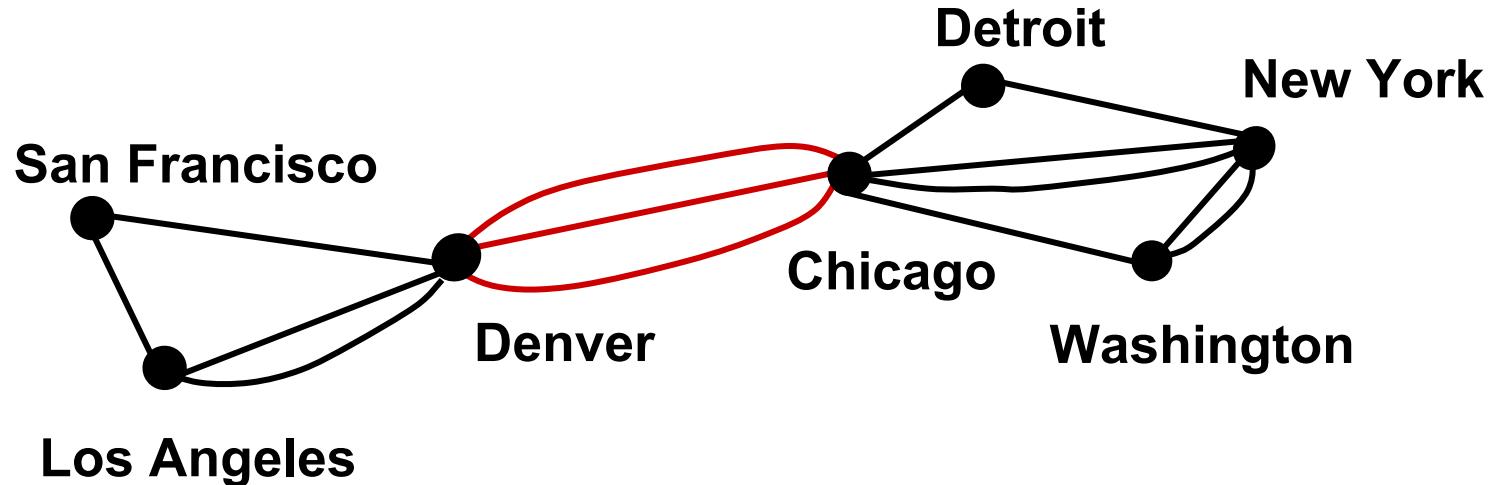
# Multigraph, A Non-Simple Graph

There can be multiple telephone lines between two computers in the network.



- In a *multigraph*  $G = (V, E)$  two or more edges may connect the same pair of vertices.

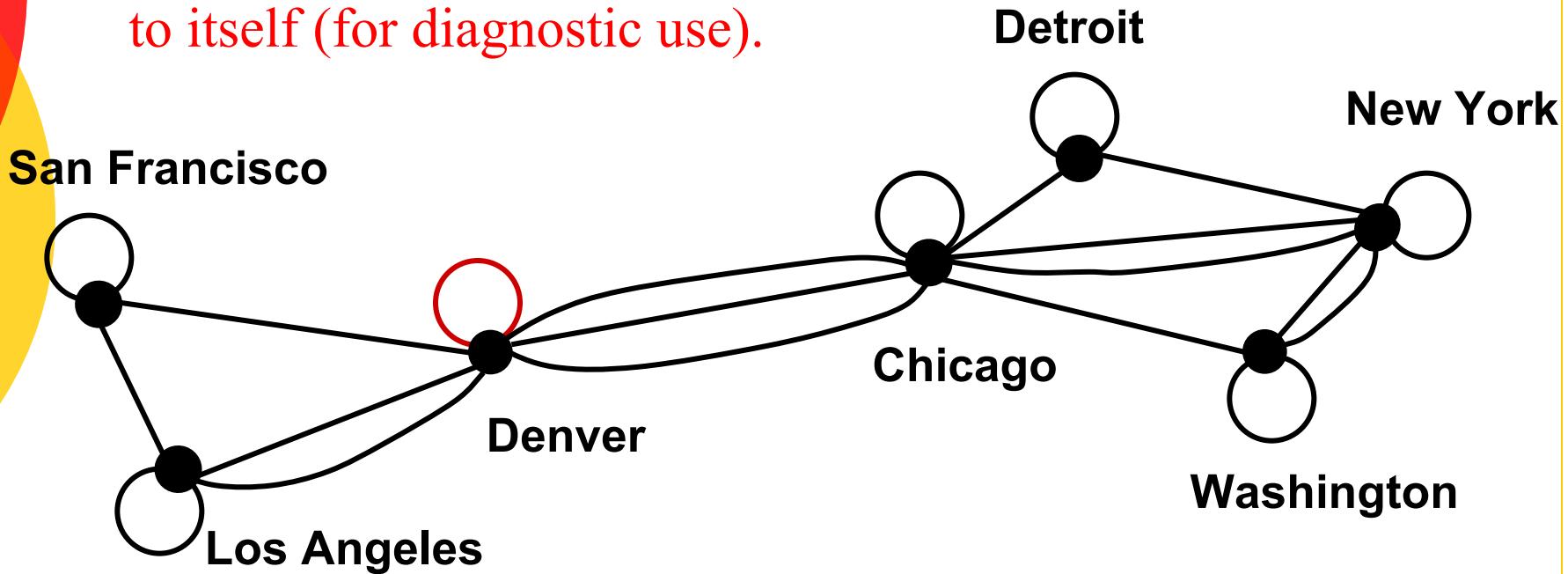
# Multiple Edges



Two edges are called ***multiple*** or ***parallel edges*** if they connect the same two distinct vertices.

# Pseudograph, A Non-Simple Graph

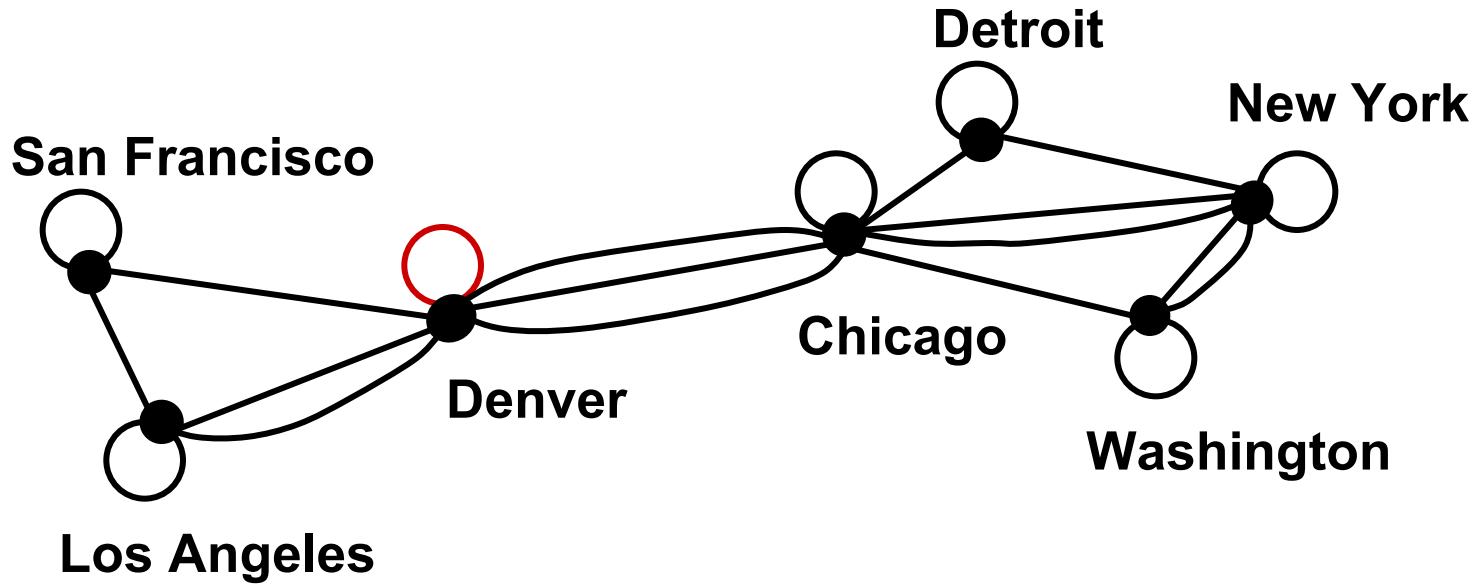
There can be telephone lines in the network from a computer to itself (for diagnostic use).



- In a **pseudograph**  $G = (V, E)$  two or more edges may connect the same pair of vertices,
- and in addition, an edge may connect a vertex to itself.

# Loops

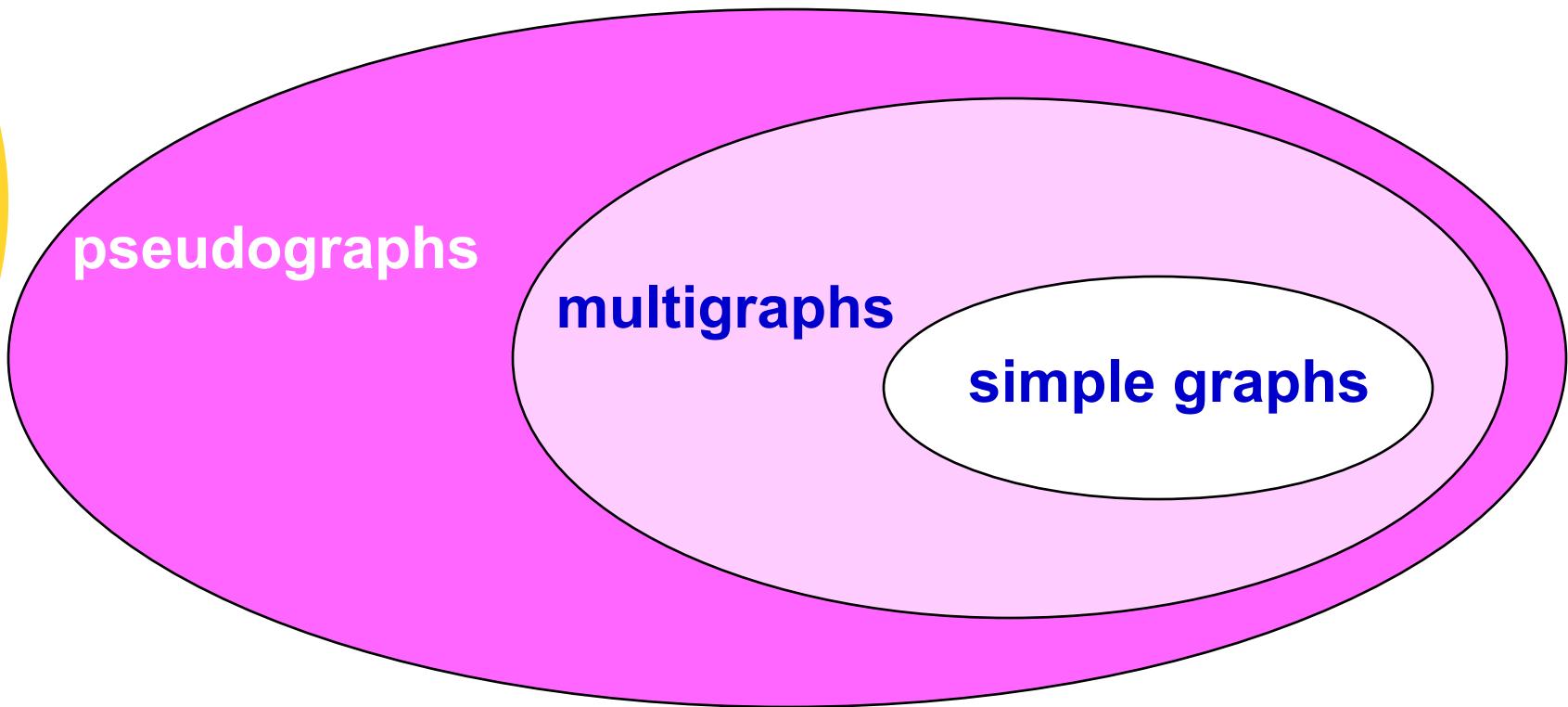
---



An edge is called a *loop*  
if it connects a vertex to itself.

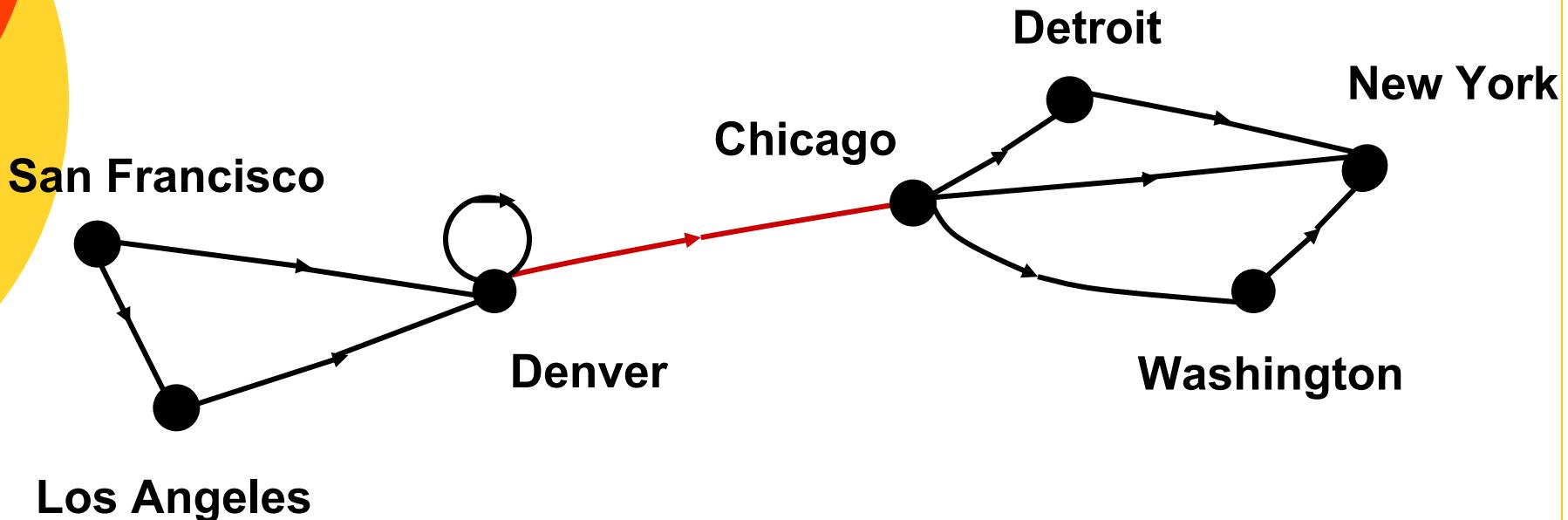
# Undirected Graphs

---



# A Directed Graph

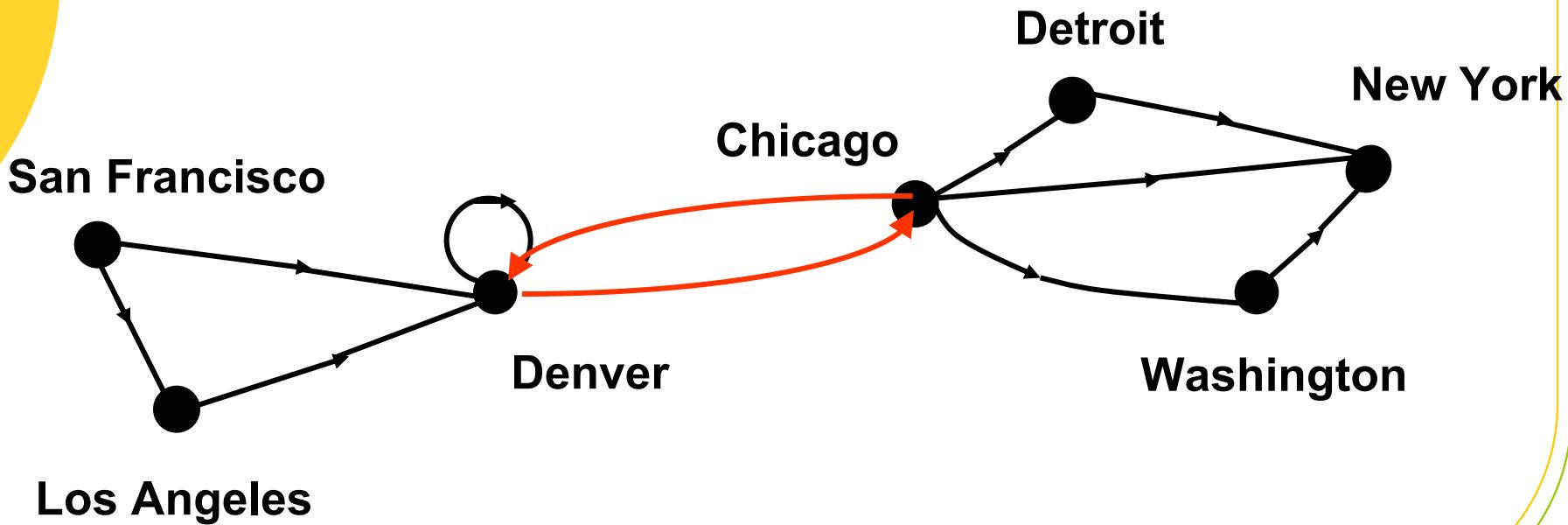
- In a ***directed graph***  $G = (V, E)$  the edges are ordered pairs of (not necessarily distinct) vertices.



Some telephone lines in the network may operate in only one direction .

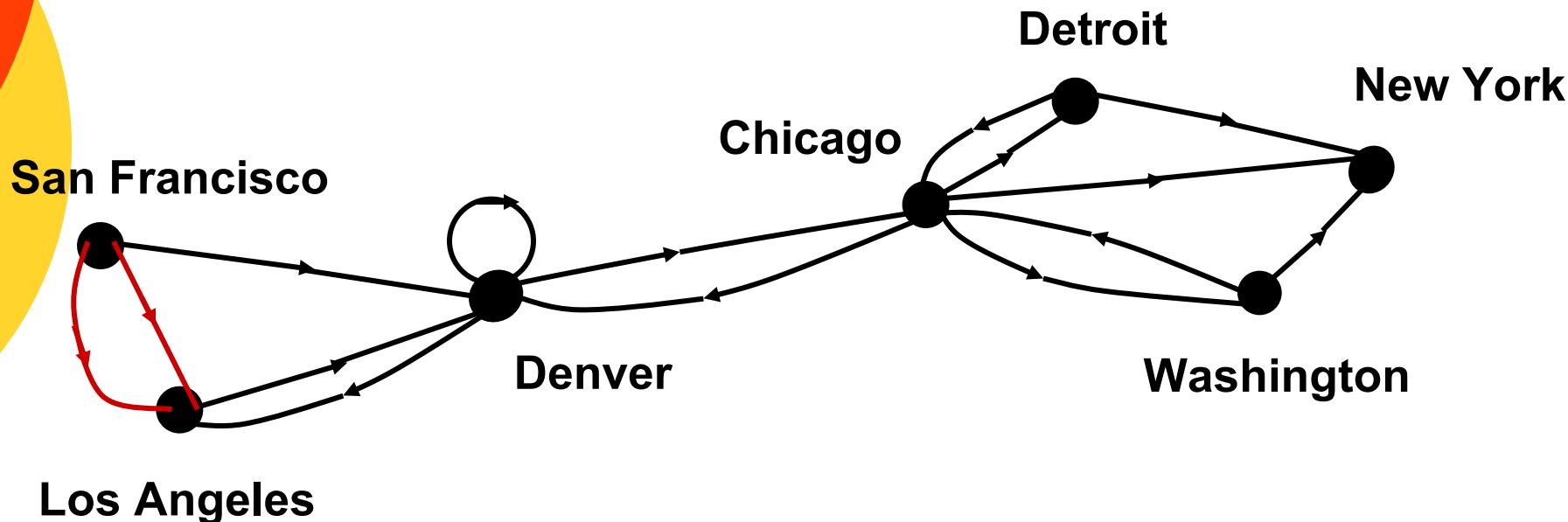
# A Directed Graph

The telephone lines in the network that operate in two directions are represented by pairs of edges in opposite directions.



# A Directed Multigraph

In a ***directed multigraph***  $G = (V, E)$  the edges are ordered pairs of (not necessarily distinct) vertices, and in addition there may be multiple edges.



There may be several one-way lines in the same direction from one computer to another in the network.

# Types of Graphs

Graph Terminology			
Type	Edges	Multiple Edges?	Loops Allowed?
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed and undirected	Yes	Yes

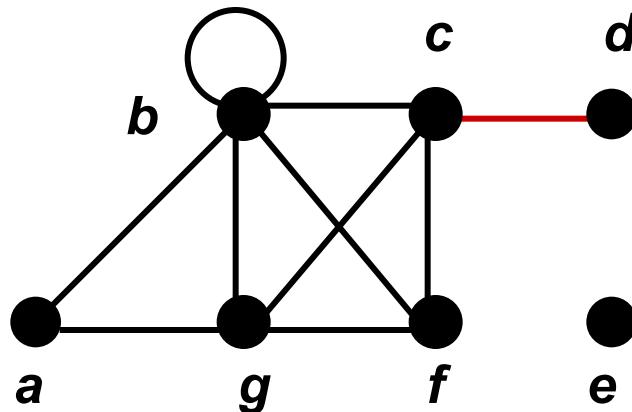
## 2. Basic Terminology Adjacent Vertices

---

**Definition 1.** Two vertices,  $u$  and  $v$  in an undirected graph  $G$  are called ***adjacent*** (or ***neighbors***) in  $G$ , if  $\{u, v\}$  is an edge of  $G$ . An edge  $e$  connecting  $u$  and  $v$  is called ***incident with vertices  $u$  and  $v$*** , or is said to connect  $u$  and  $v$ . The vertices  $u$  and  $v$  are called ***endpoints*** of the edge  $\{u, v\}$ .

# Degree of a vertex

**Definition 1.** The *degree of a vertex* in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex.



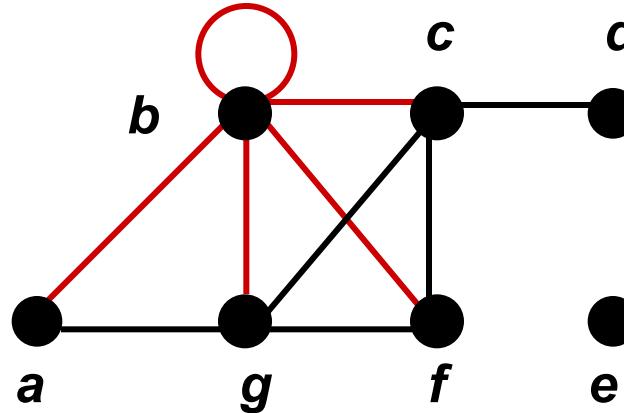
$$\deg(d) = 1$$

$$\deg(e) = 0$$

## Degree of a vertex

**Definition 1.** The *degree of a vertex* in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex.

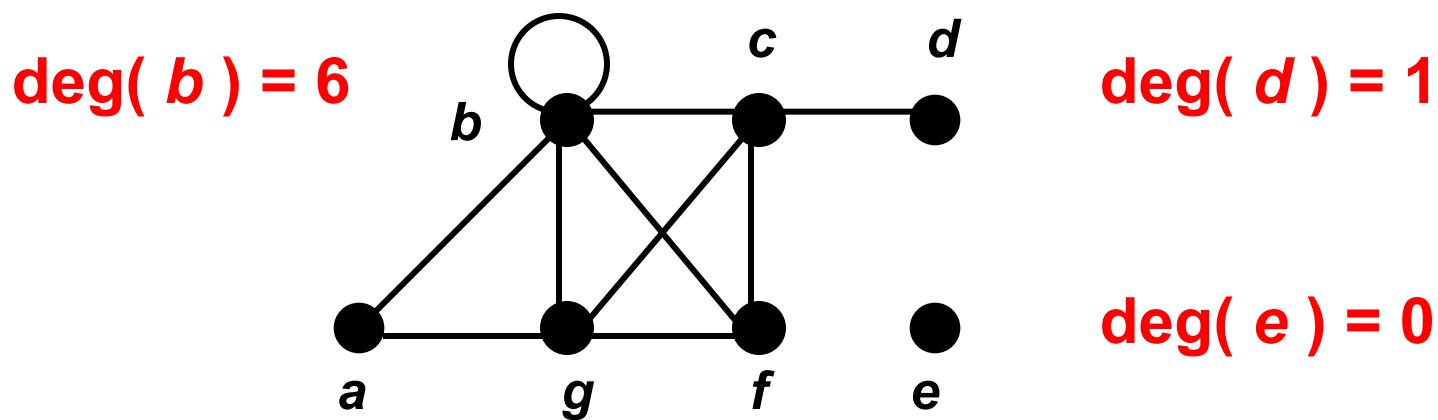
$$\deg(b) = 6$$



## Degree of a vertex

Find the degree of all the other vertices.

$$\deg(a) \quad \deg(c) \quad \deg(f) \quad \deg(g)$$



## Degree of a vertex

Find the degree of all the other vertices.

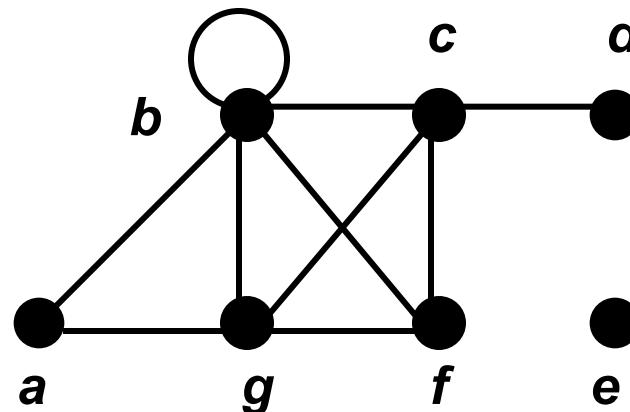
$$\deg(a) = 2$$

$$\deg(c) = 4$$

$$\deg(f) = 3$$

$$\deg(g) = 4$$

$$\deg(b) = 6$$



$$\deg(d) = 1$$

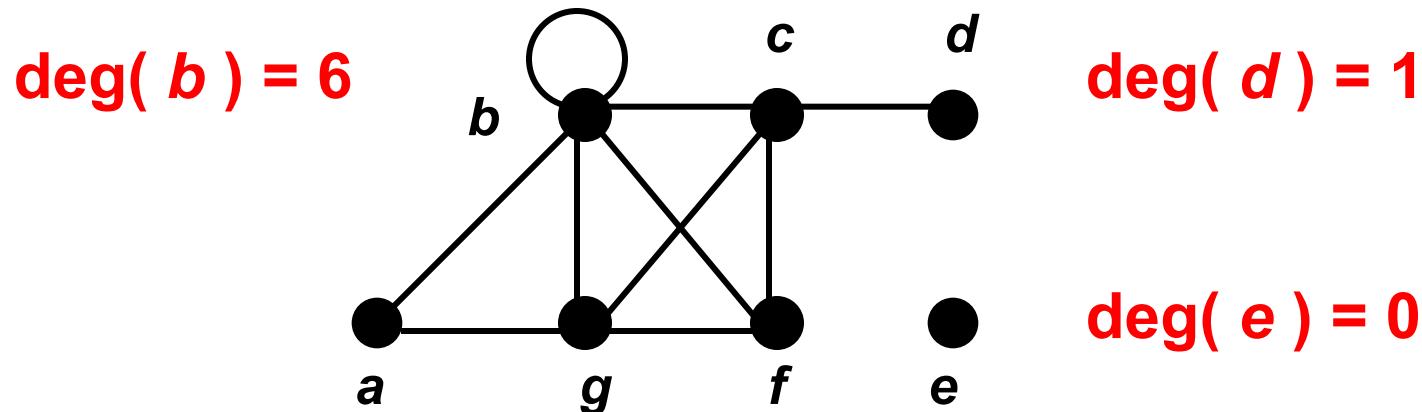
$$\deg(e) = 0$$

## Degree of a vertex

Find the degree of all the other vertices.

$$\deg(a) = 2 \quad \deg(c) = 4 \quad \deg(f) = 3 \quad \deg(g) = 4$$

$$\text{TOTAL of degrees} = 2 + 4 + 3 + 4 + 6 + 1 + 0 = 20$$



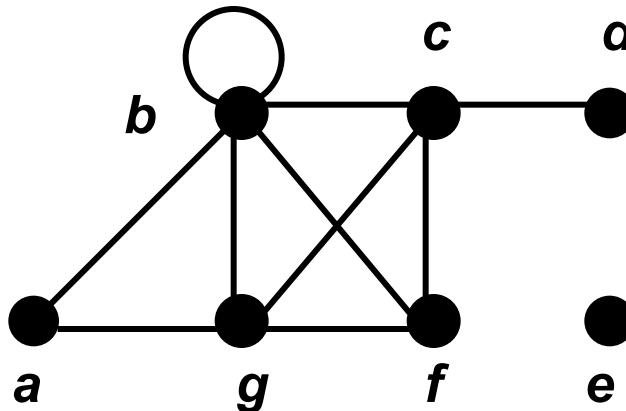
## Degree of a vertex

Find the degree of all the other vertices.

$$\deg(a) = 2 \quad \deg(c) = 4 \quad \deg(f) = 3 \quad \deg(g) = 4$$

$$\text{TOTAL of degrees} = 2 + 4 + 3 + 4 + 6 + 1 + 0 = 20$$

TOTAL NUMBER OF EDGES = 10



$$\deg(d) = 1$$

$$\deg(e) = 0$$

# Handshaking Theorem

**Theorem 1.** Let  $G = (V, E)$  be an undirected graph  $G$  with  $e$  edges. Then

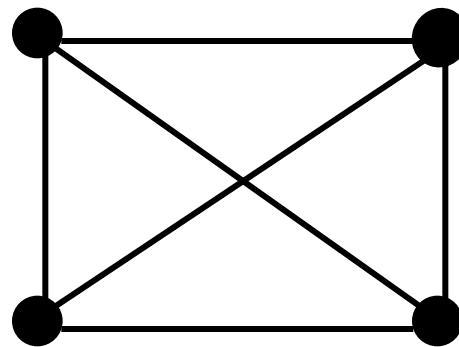
$$\sum_{v \in V} \deg(v) = 2e$$

*“The sum of the degrees over all the vertices equals twice the number of edges.”*

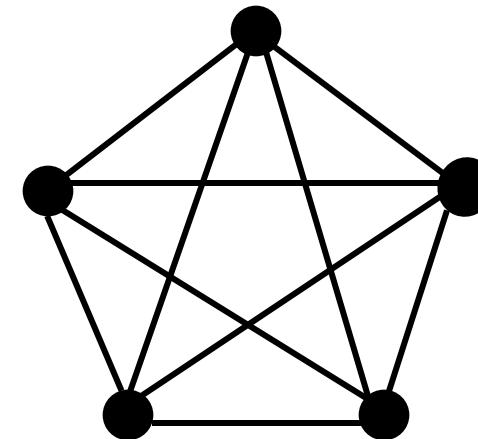
**NOTE:** This applies even if multiple edges and loops are present.

### 3. Some Special Simple Graphs

---



$K_4$

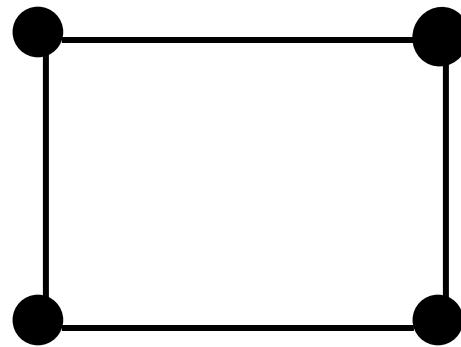


$K_5$

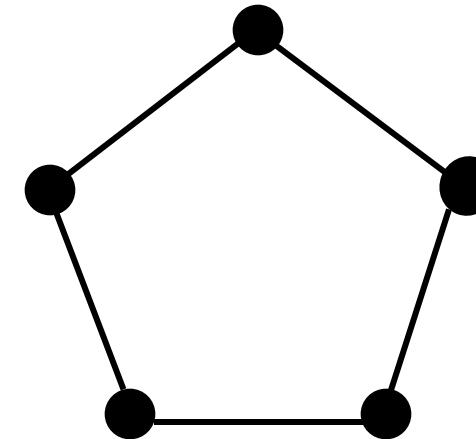
Complete graph  $K_n$

### 3. Some Special Simple Graphs

---



$C_4$

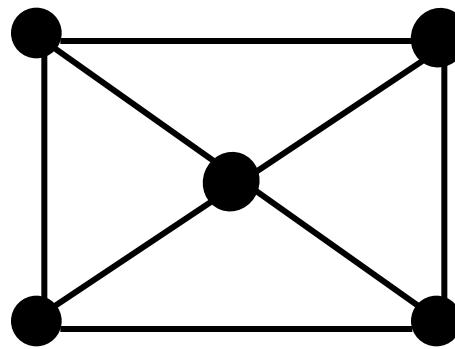


$C_5$

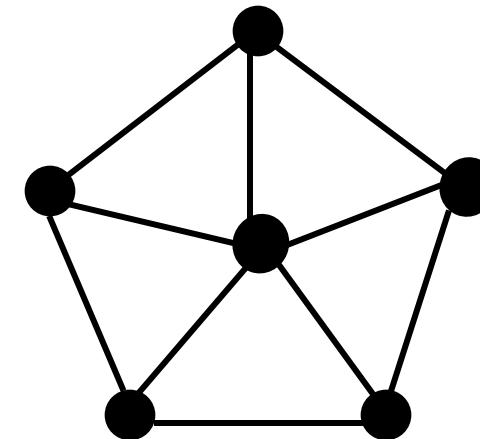
**Cycle  $C_n$**

### 3. Some Special Simple Graphs

---



$W_4$

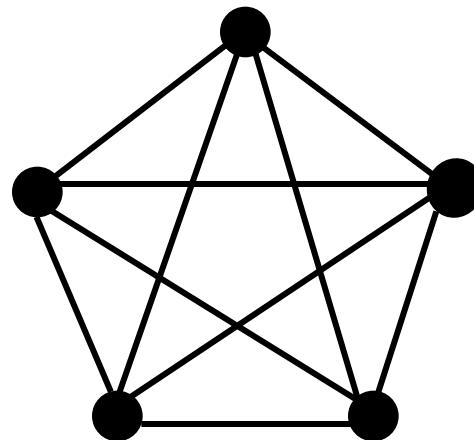


$W_5$

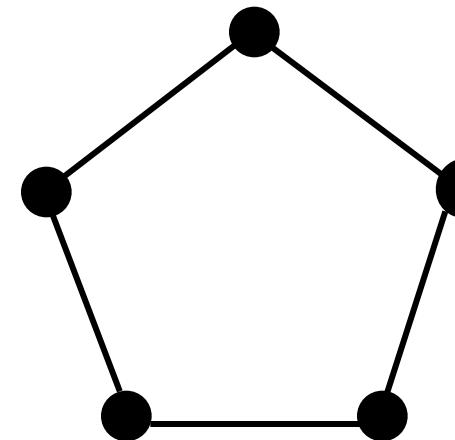
**Wheele  $W_n$**

## 4. New Graphs from Old Graphs

**Definition.** A *subgraph* of a graph  $G = (V, E)$  is a graph  $H = (W, F)$  where  $W \subseteq V$  and  $F \subseteq E$ .



$K_5$

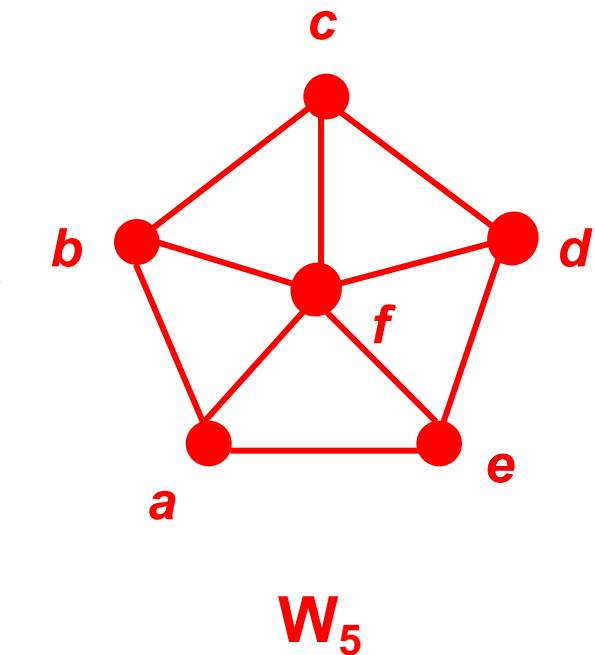
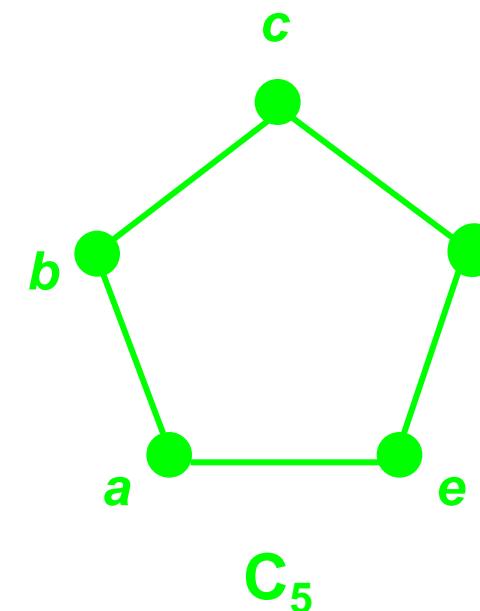
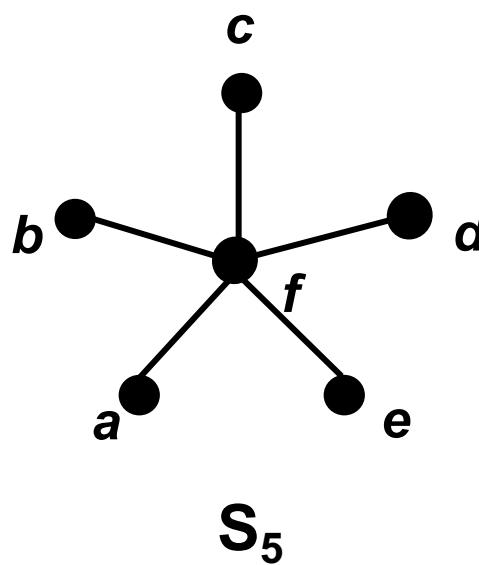


$C_5$

$C_5$  is a subgraph of  $K_5$

## 4. New Graphs from Old Graphs

**Definition 7.** The *union* of 2 simple graphs  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ , denoted by  $G_1 \cup G_2$ , is the simple graph with vertex set  $V = V_1 \cup V_2$  and edge set  $E = E_1 \cup E_2$ .



$W_5$  is the union of  $S_5$  and  $C_5$



## **Part 2: Representing Graphs, Connectivity, and Euler Paths**

---

1. Representing Graphs
2. Adjacency Matrices
3. Incidence Matrices
4. Isomorphism of Graphs
5. Paths
6. Connectedness
7. Euler Paths and Circuits

# 1. Representing Graphs

---

- One way to represent a graph without multiple edges is to list all the edges of this graph.
- Another way to represent a graph with no multiple edges is to use **adjacency lists**, which specify the vertices that are adjacent to each vertex of the graph.

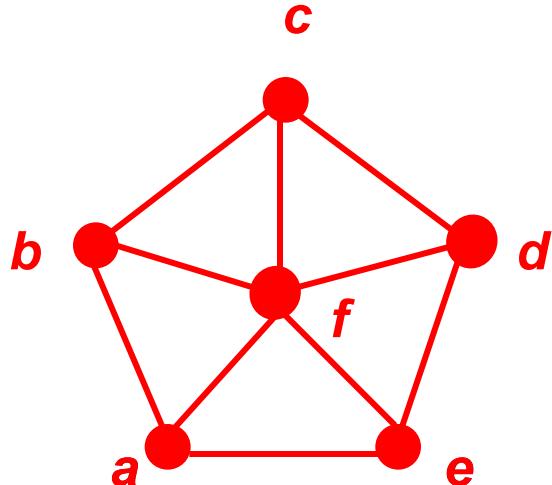
## 2. Adjacency Matrix

---

A simple graph  $G = (V, E)$  with  $n$  vertices can be represented by its **adjacency matrix**,  $\mathbf{A}$ , where entry  $a_{ij}$  in row  $i$  and column  $j$  is

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge in } G, \\ 0 & \text{otherwise.} \end{cases}$$

## Finding the adjacency matrix

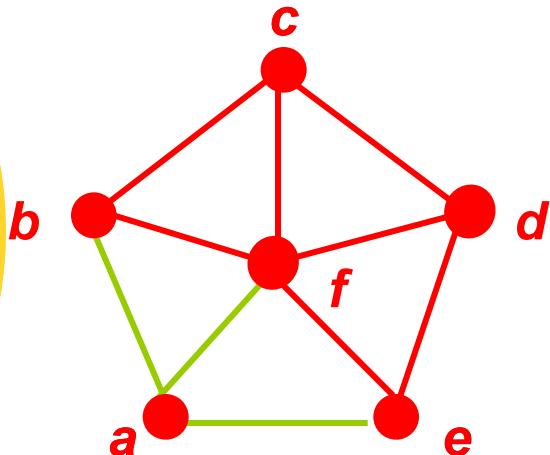


**$W_5$**

This graph has 6 vertices  
a, b, c, d, e, f.

We can arrange them in that order.

# Finding the adjacency matrix

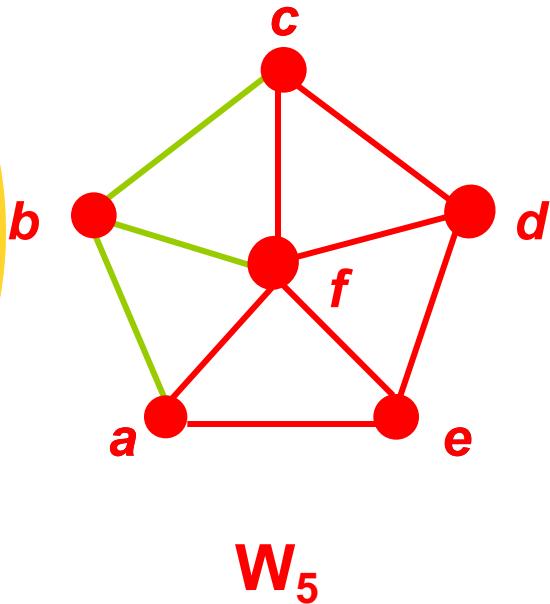


$W_5$

	TO	a	b	c	d	e	f
FROM	a	0	1	0	0	1	1
	b						
	c						
	d						
	e						
	f						

There are edges from a to b, e, and f

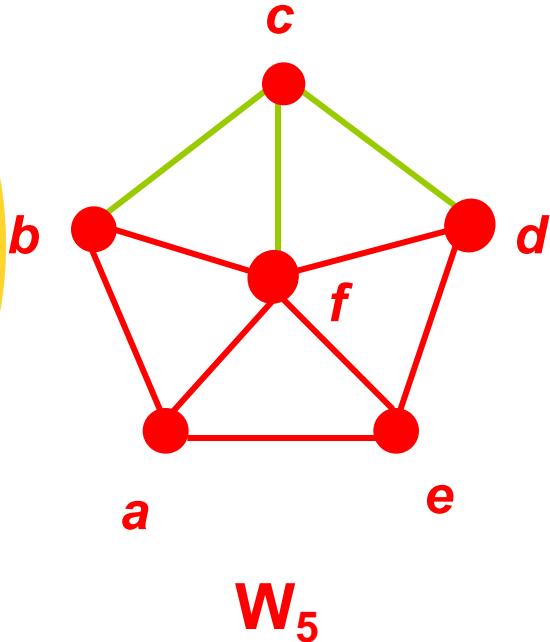
# Finding the adjacency matrix



	TO					
	a	b	c	d	e	f
FROM	0	1	0	0	1	1
a	1	0	1	0	0	1
b						
c						
d						
e						
f						

There are edges from b to a, c, and f

# Finding the adjacency matrix

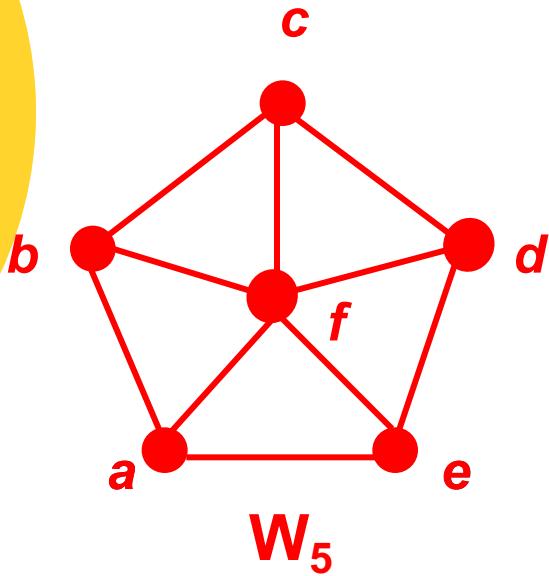


		TO					
		a	b	c	d	e	f
FROM	a	0	1	0	0	1	1
	b	1	0	1	0	0	1
c	0	1	0	1	0	1	
d	1	0	1	0	0	1	
e	0	1	0	1	0	1	
f	1	1	0	0	1	0	

There are edges from c to b, d, and f

# Finding the adjacency matrix

COMPLETE THE ADJACENCY MATRIX . . .



TO  
FROM

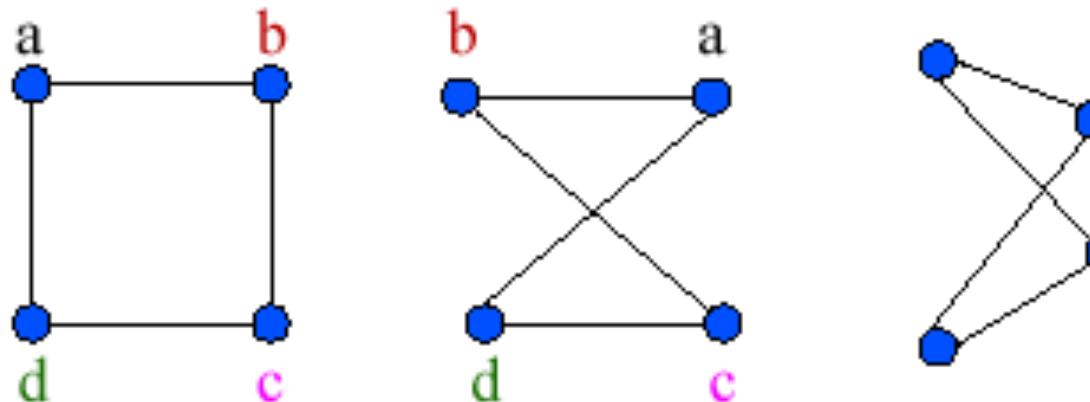
a  
b  
c  
d  
e  
f

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

Notice that this matrix is symmetric. i.e.  $a_{ij} = a_{ji}$

## 4. Graph Isomorphism

**Definition 1.** An *isomorphism* between simple graphs  $G_1$  and  $G_2$  is a bijection between the set of vertices that preserves the adjacency list

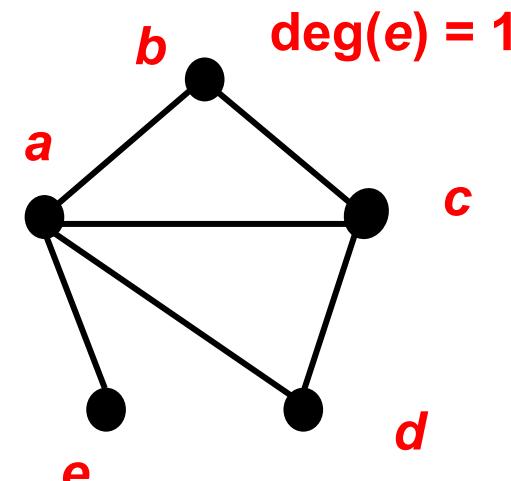
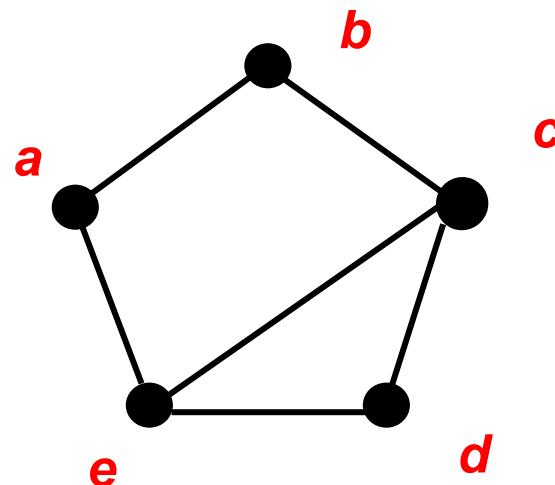


## 4. Graph Isomorphism

**Note.** *Isomorphic* simple graphs must have the same *invariants*:

- ✓ The number of vertices
- ✓ The number of edges
- ✓ The degrees of the vertices

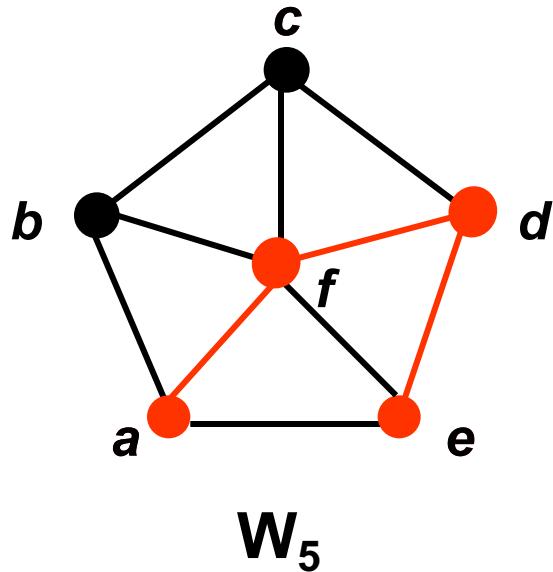
No vertex of deg 1



Non-isomorphic graphs

## 5. Connectivity-Paths

**Definition 1.** A ***path of length n*** from  $u$  to  $v$  in an undirected graph is a sequence of edges  $e_1 = \{u, x_1\}$ ,  $e_2 = \{x_1, x_2\}$ , . . . ,  $e_n = \{x_{n-1}, v\}$ . We also denote this path by:  $ux_1x_2 \dots x_{n-1}v$

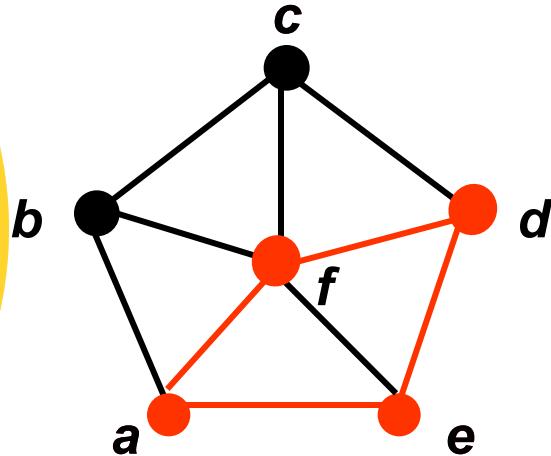


**One path from a to e**

This path passes through vertices  $f$  and  $d$  in that order.

The length of this path is 3

## One path from a to a



**W<sub>5</sub>**

This path passes through vertices f, d, e, in that order. It has **length** 4.

It is a **circuit** because it begins and ends at the same vertex.

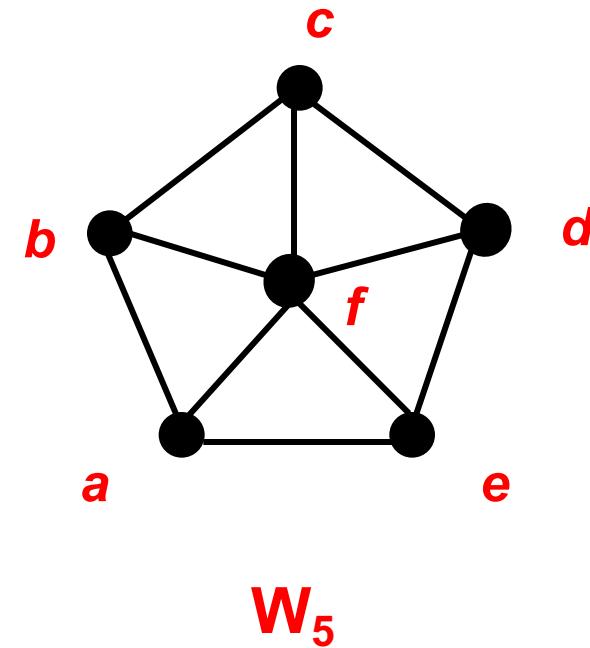
It is called **simple** because it does not contain the same edge more than once.

## 6. Connectedness in Undirected Graphs

**Definition 3.** An undirected graph is called ***connected*** if there is a path between every pair of distinct vertices of the graph.

IS THIS GRAPH CONNECTED?

YES



## 6. Connectedness in Undirected Graphs

**Theorem .** There is a simple path between every pair of distinct vertices of a connected undirected graph.

**Proof.** Let  $x_0 x_1 \dots x_n$  be a path of shortest length joining the vertices  $x_0 = u$  and  $x_1 = v$ .

If this path is not simple, then there are  $0 \leq i < j$  such that  $x_i = x_j$ .

Now deleting the edges  $x_i x_{i+1}, x_{i+1} x_{i+2}, \dots, x_{j-1} x_j$ , we will obtain a path of shorter length joining  $u$  and  $v$ :

***contradiction.***

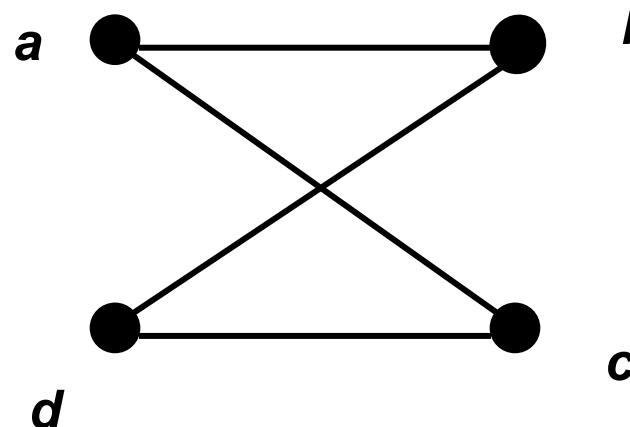
# Counting Paths between Vertices

**Theorem.** Let  $G$  be a graph with adjacency matrix  $\mathbf{A}$  with respect to the ordering  $v_1, v_2, \dots, v_n$ .

The number of different paths of length  $r$  from  $v_i$  to  $v_j$ , where  $r$  is a positive integer, equals the entry in row  $i$  and column  $j$  of  $\mathbf{A}^r$ .

**NOTE.** This applies with directed or undirected edges, with multiple edges and loops allowed.

## Example



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

$$A^4 = \begin{bmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{bmatrix}$$

Thus there are 8 different paths of length 4 joining  $a$  and  $d$

See: Example 15, page 723

## 7. Euler Paths

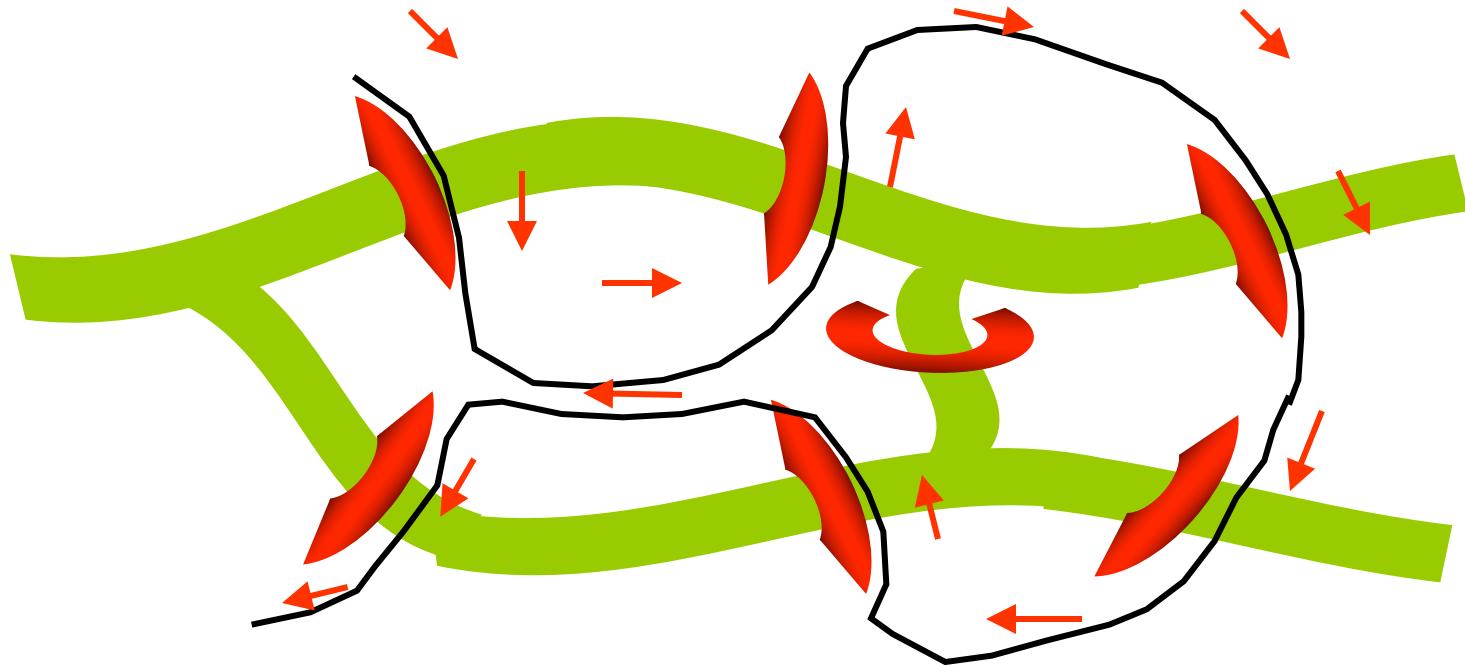
---

**Problem.** The town of Königsberg was divided into four sections by the branch of the Pregel River



These four sections are connected by seven bridges

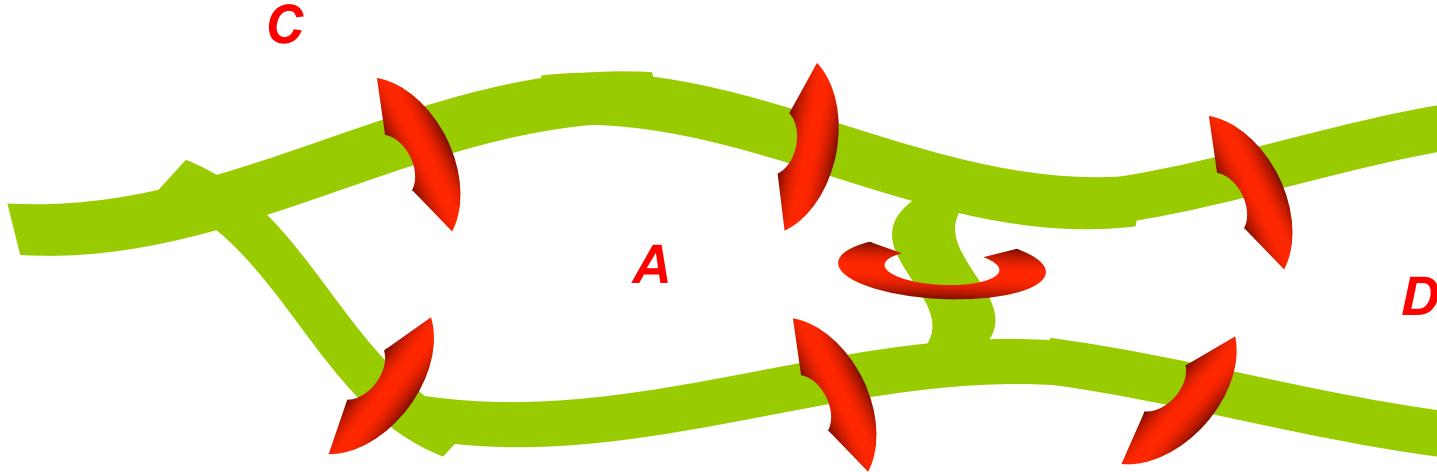
## 7. Euler Paths



**Question.** Can one cross seven bridges and return to the starting point without crossing any bridge twice?

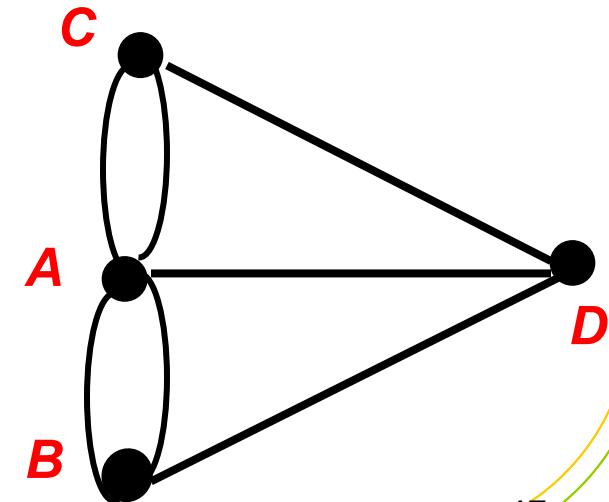
In the eighteen<sup>th</sup> century, Euler solved this problem using Graph Theory

## 7. Euler Paths



Euler modeled this problem using the multigraph:

- ✓ four sections correspond to four vertices  $A, B, C, D$ .
- ✓ each bridge corresponds to an edge

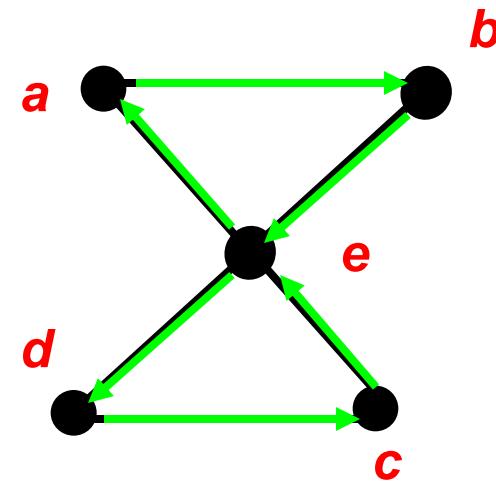


## 7. Euler Paths

**Definition.** In a graph  $G$  an *Euler path* is a simple path containing every edge of  $G$ . A simple circuit which is also an Euler path is called an *Euler circuit*.

**Example.** Does this graph have an Euler circuit?

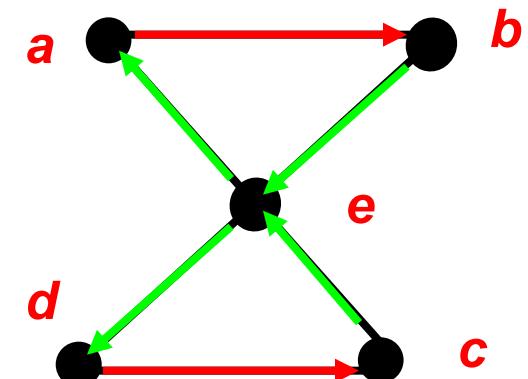
**Solution.** An Euler circuit is *abedcea*



## 7. Euler Paths

**Theorem.** A connected multigraph has an Euler circuit if and only if each vertex has even degree.

**Proof.** When an Euler circuit passes by a vertex, it arrives by an edge and leaves by a different edge. Therefore the degree of the vertex increases by 2 units.

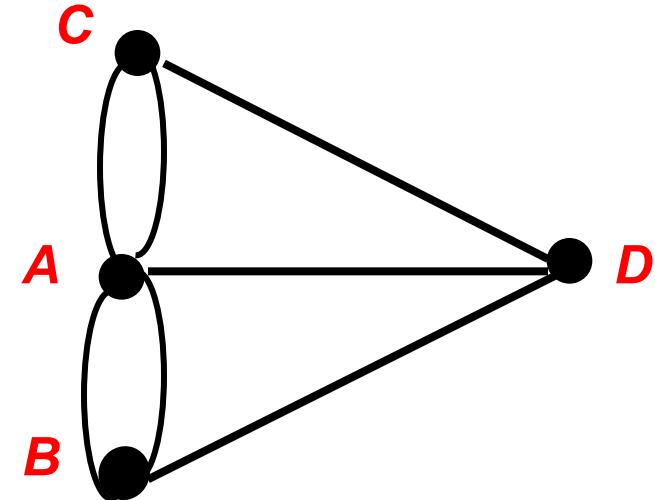


$$\deg(e) = 0 + 2 + \dots$$

$$= 2 + 2 = 4$$

## 7. Euler Paths

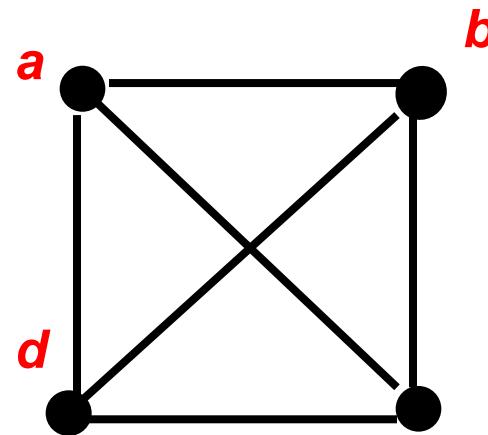
**Example.** All vertices of the multigraph corresponding to the Königsberg problem have odd degree so there is no Euler circuit. Thus the Problem has no solution



**Example.** All four vertices of this graph have dgree 3

So there is no Euler circuit.

Thus we cannot draw it without removing the pen head



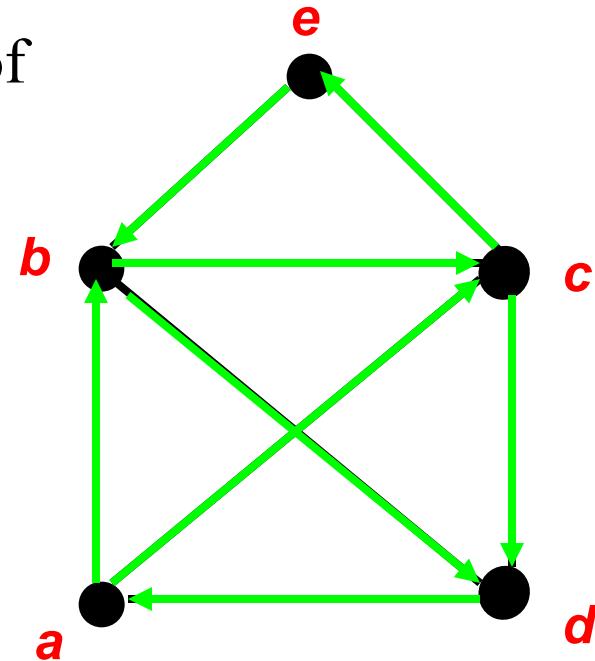
## 7. Euler Paths

**Example.** This graph has two vertices of degree 3, namely **a** and **d**

So there is no Euler circuit

However there is an Euler path

*abcdaceb*



Therefore we can draw this without removing the pen head

The fact that this graph has exactly two vertices of odd degree is not an incidence. In fact we have

## 7. Euler Paths

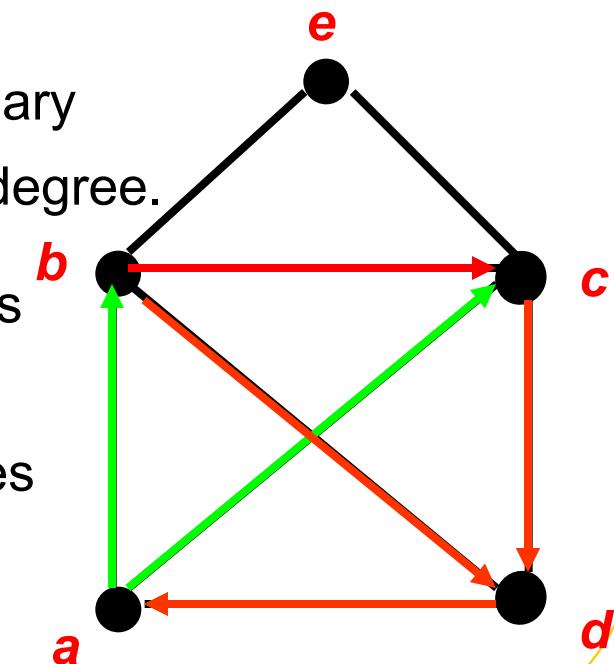
**Theorem.** A connected multigraph has an Euler path that is not an Euler circuit if and only if there are exactly two vertices of odd degree.

**Proof.** As in the last Theorem, the intermediary vertices passed by an Euler path has even degree.

At the starting vertex **a**, the degree increases by 1 units when it leaves.

In any subsequent visit to **a**,  $\deg(a)$  increases by 2. So  $\deg(a)$  is odd.

Similarly degree of ending vertex **d** is odd.



## 7. Algorithm for finding Euler Circuit

---

**Step 1:** Start from any  $v$  to find Circuit  $C$

**Step 2:** Delete the edges in  $C$  out of  $G$  and alone vertices.

**Step 3:** If  $G = \emptyset$  then  $C$  is Euler Circuit of  $G$ .

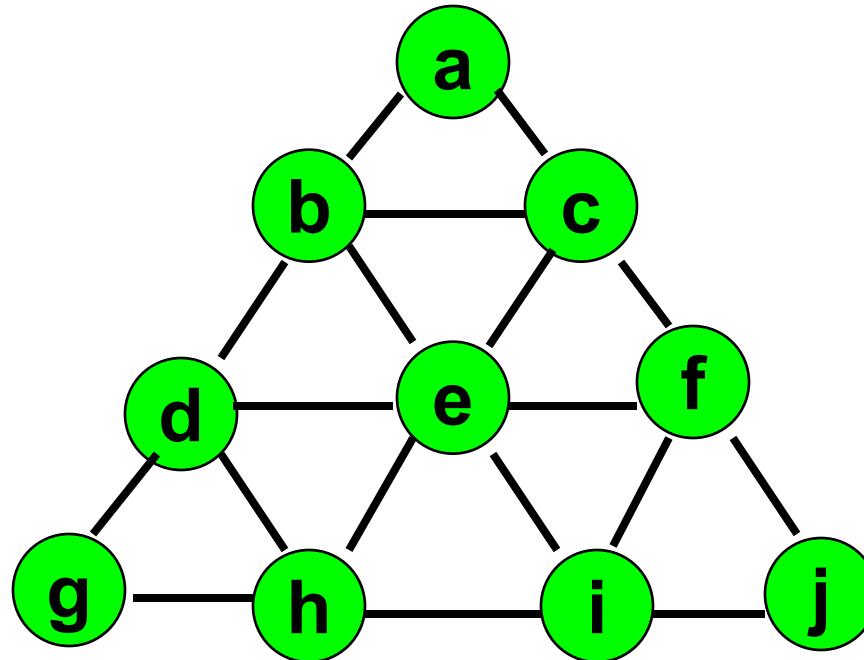
**Step 4:** Find the sharing vertices  $v$  of  $C$  and  $G$ :

- Sort  $C$  starting from  $v$ .
- Start from  $v$ , find Circuit  $C'$ .
- Connect  $C'$  into  $C$ .
- Return Step 2.

## 7. Algorithm for finding Euler Circuit

Example: Given  $G=(V, E)$  as follows:

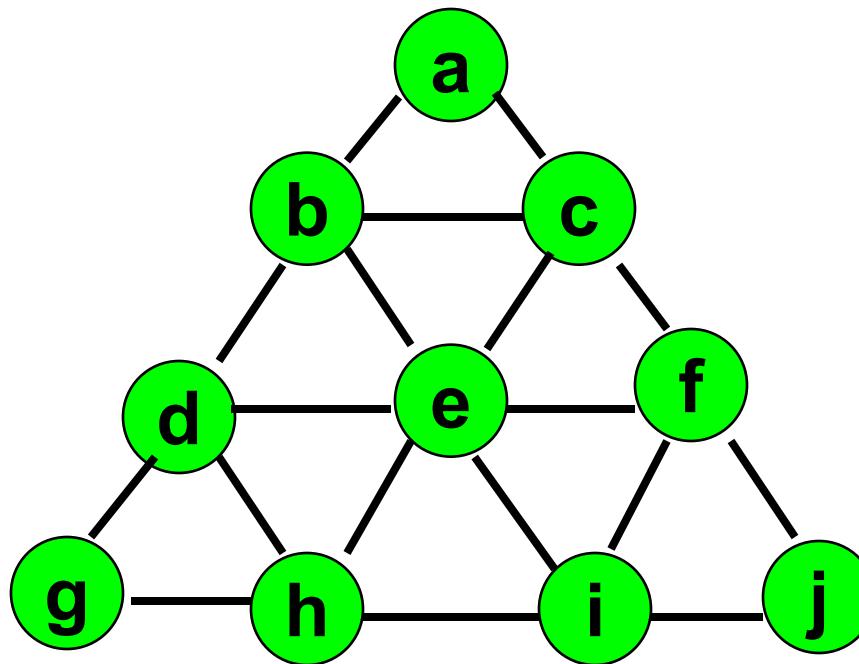
- a) Prove  $G$  is an Euler graph.
- b) Find Euler Circuit  $C$  of  $G$



## 7. Algorithm for finding Euler Circuit

*Solution:*

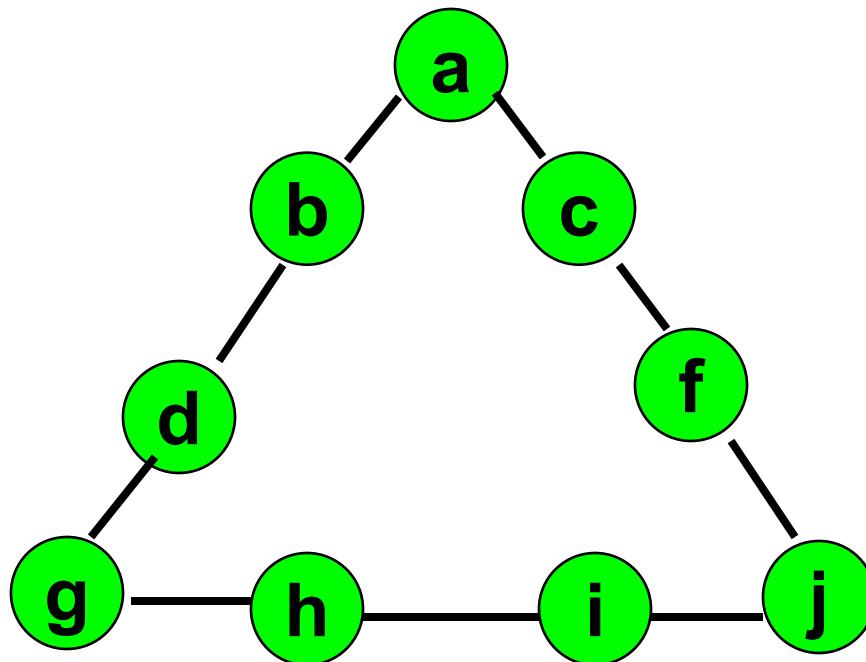
- a) G is connected and all vertices have even degree.



## 7. Algorithm for finding Euler Circuit

*Solution:*

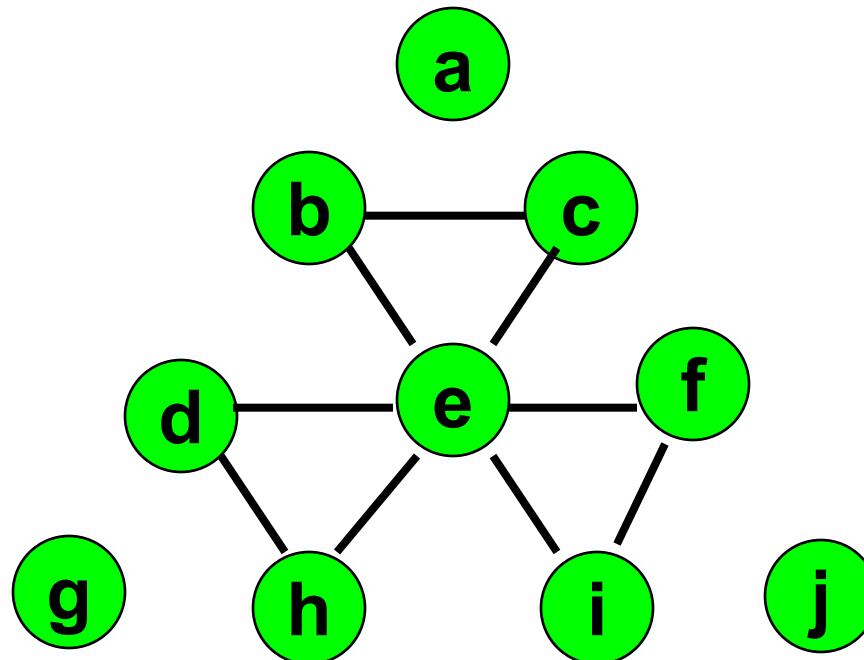
- b) Starting from vertex a,  
 $C = (a, b, d, g, h, i, j, f, c, a)$ .



## 7. Algorithm for finding Euler Circuit

*Solution:*

- Delete edge on C out of G



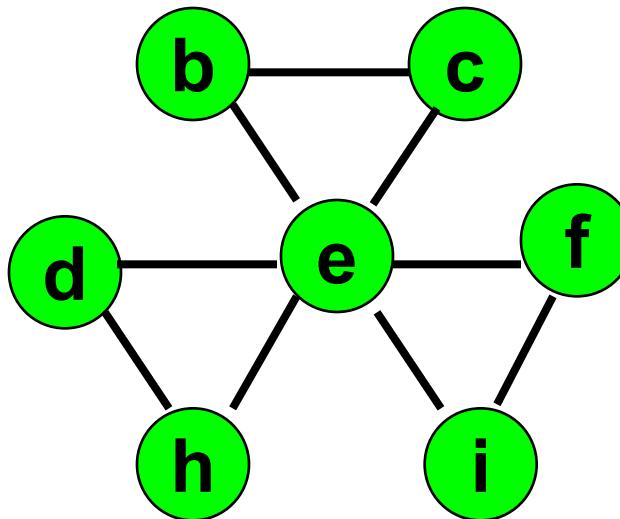
- Delete the alone vertices: *a, g, j.*

## 7. Algorithm for finding Euler Circuit

*Solution:*

After deleting the alone vertices:  $a, g, j$ .

The remaining graph is  $G_1$



## 7. Algorithm for finding Euler Circuit

C and  $G_1$  have the same vertex b.

Sort C starting from b

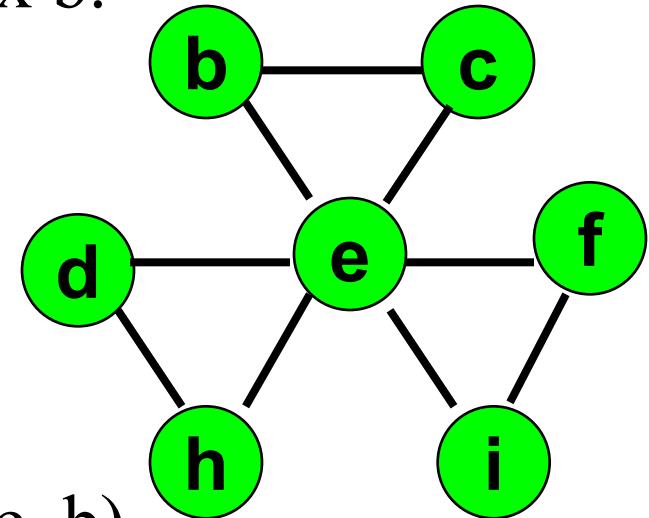
$C = (b, d, g, h, i, j, f, c, a, b)$ .

$G_1$  has circuit starting from b,

$C_1 = (b, c, e, b)$ .

Connect  $C_1$  into C,

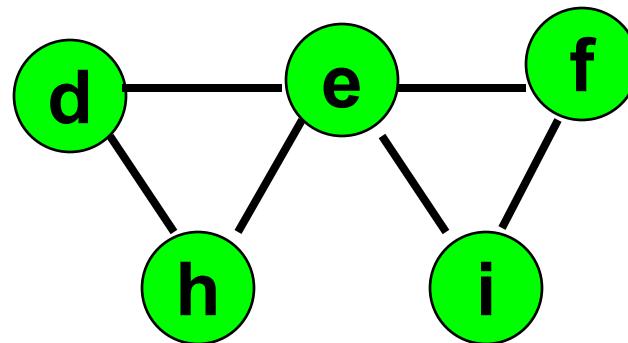
$C = (b, d, g, h, i, j, f, c, a, b, c, e, b)$ .



## 7. Algorithm for finding Euler Circuit

---

The remaining  
graph is  $G_2$



## 7. Algorithm for finding Euler Circuit

C and  $G_2$  have sharing vertex d.

Sort C starting from d,

$C = (d, g, h, i, j, f, c, a, b, c, e, b, d)$ .

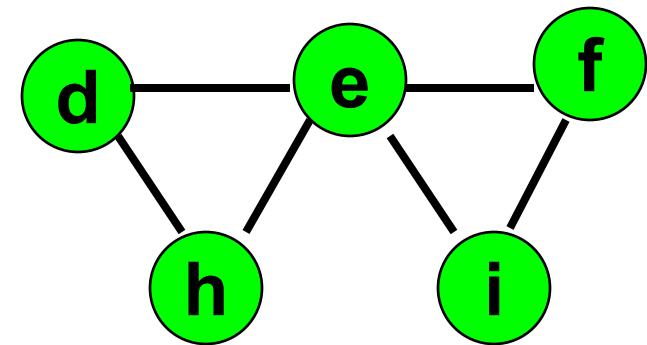
On  $G_2$  has circuit starting from d,

$C_2 = (d, e, f, i, e, h, d)$ .

Connect  $C_2$  into C,

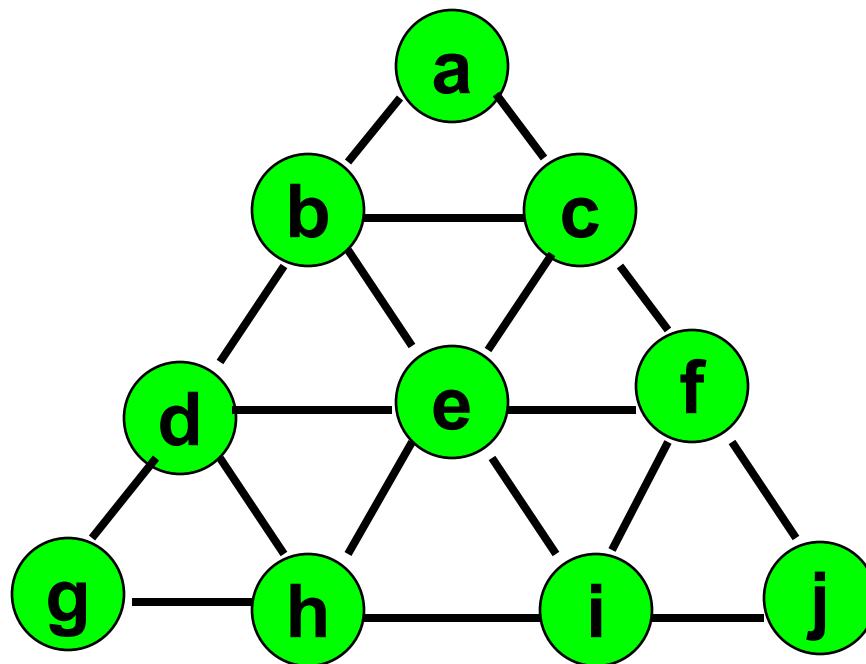
$C = (d, g, h, i, j, f, c, a, b, c, e, b, d, e, f, i, e, h, d)$ .

$G_3 = \emptyset$ , C is the final Euler circuit that we want to find.



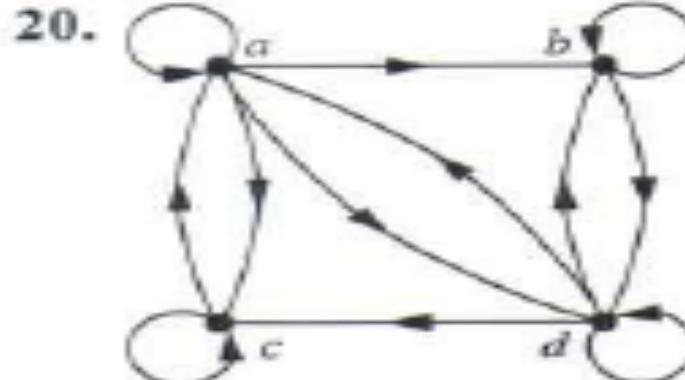
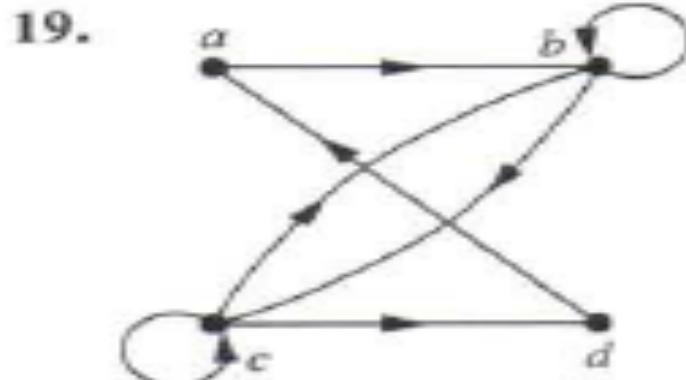
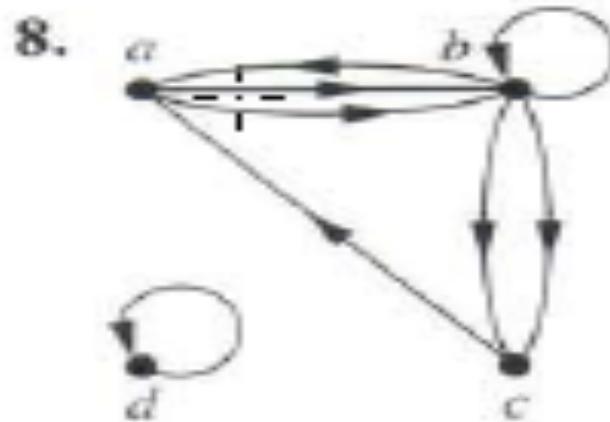
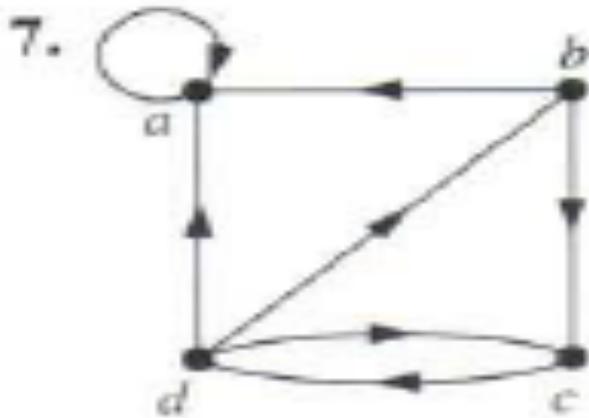
## 7. Algorithm for finding Euler Circuit

$C=(a, b, c, e, b, d, e, f, i, e, h, d, g, h, i, j, f, c, a)$ .



# CW Assignment.

- Find the number of edges, number of vertices, and degree of each vertex (both input and output vertices)
- 



# Homework.

---

- Pages 683: 10, 12, 14, 16
- Pages 699: 6, 10, 18, 20
- Pages 710, 711: 10, 16, 18, 22, 24
- Implement the algorithm  
EulerCircuit.C/C++ following the  
description in slides 53 to 62.

Deadline: before 10:00 PM December 5<sup>th</sup> 2022