MOHAMED BOUDIAF UNIVERSITY OF M'SILA

DEPARTMENT OF COMPUTER SCIENCE

Advanced Graph Theory

Chapter 2: Representation of Graphs

Dr. SAID KADRI

Lecturer, Class A

Department of Computer Science, Faculty of Mathematics and Computer Science,

Mohamed Boudiaf University of M'sila

E-mail:kadri.said28@gmail.com

Website:https://kadrisaid28.wixsite.com/sgadri

2017 - 2018

Methods of graph representation

We distinguish 02 classes of methods:

1. Static methods (use of matrices)

- Adjacency matrix (vertices – vertices)
- Incidence matrix (vertices – arcs)
- List of arcs (sorted – unsorted)
- List of successors
- List of predecessors
- Linear list of successors
- Linear list of successors with EOL
- Linear list of successors with BOL

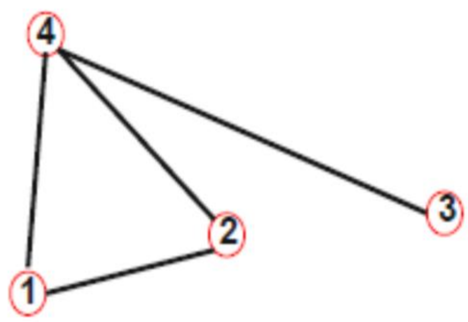
2. Dynamic methods (using linked lists)

- Dynamic list of vertices
- Dynamic list of arcs

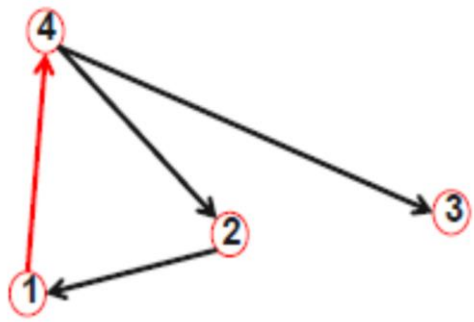
Static Methods

1. Adjacency matrix (vertices – vertices)

Graph G1



Graphe G2



$$\begin{cases} A_{ij} = 1 & \text{if there exist a link between vertices } i \text{ and } j \\ A_{ij} = 0 & \text{if vertices } i \text{ and } j \text{ are not connected} \end{cases}$$

$$A_{ij} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

A(N, N)

A_{14}

$$A_{ij} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

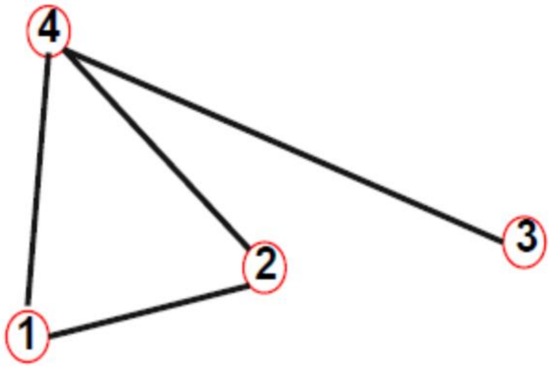
A(N, N)

NB: Note that for a directed graph (right) the matrix is not symmetrical.

Adjacency matrix and vertex degrees

Undirected graph

$$A_{ij} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

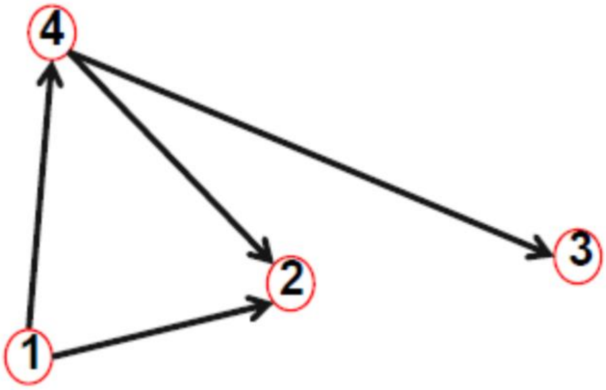


$$\begin{cases} A_{ij} = A_{ji} \\ A_{ii} = 0 \end{cases}$$

$$\boxed{\mathbf{k_i} = \sum_{j=1}^{N} \mathbf{A_{ij}} = \mathbf{k_j} = \sum_{i}^{n} \mathbf{A_{ij}}}$$

$$\boxed{L = \frac{1}{2}\sum_{i=1}^{N} k_i = \frac{1}{2}\sum_{ij}^{N} A_{ij}}$$

Directed graph



$$A_{ij} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

$$\begin{cases} A_{ij} \neq A_{ji} \\ A_{ii} = 0 \end{cases}$$

$$\boxed{k_i^{ex} = \sum_{j=1}^{N} A_{ij} \quad ; \quad k_j^{in} = \sum_{i=1}^{N} A_{ij}}$$

$$\boxed{L = \sum_{i=1}^{N} k_i^{in} = \sum_{j=1}^{N} k_j^{out} = \sum_{i,j}^{N} A_{ij}}$$

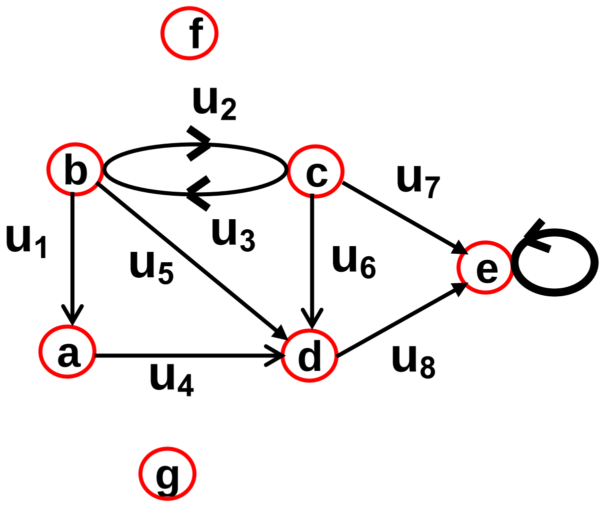
2. Incidence matrix (vertices – arcs)

Let the following graph G(X, U):

X = {a, b, c, d, e, f, g}

U = {(a, d), (b, a), (b, c), (b, d), (c, b), (c, d), (c, e), (d, e) , (e, e)}

	u1	u2	u3	u4	u5	u6	u7	u8	u9
a	-1	0	0	+1	0	0	0	0	0
b	+1	+1	-1	0	+1	0	0	0	0
c	0	-1	+1	0	0	+1	+1	0	0
d	0	0	0	-1	-1	-1	0	+1	0
e	0	0	0	0	0	0	-1	-1	2
f	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	0	0	0	0



A(N, M)

Remarks:

- **Each column in the matrix contains a single (+1) corresponding to the initial endpoint of the arc, and a single (-1) corresponding to its terminal endpoint**
- **The number of (+1) on the line gives the external 1/2 degree of the vertex, although the number of (-1) gives the internal 1/2 degree of the same vertex.**

3. List of sorted arcs (by initial end)

U4	U1	U2	U5	U3	U6	U7	U8	U9
a	b	b	b	c	c	c	d	e
d	a	c	d	b	d	e	e	e

f	g	…..

Isolated vertices

A(2, M)/**A(3, M)**

4. List of unsorted arcs (taken arbitrarily)

U4	U1	U3	U2	U8	U9	U5	U6	U7
a	b	c	b	d	e	b	c	c
d	a	b	c	e	e	d	d	e

f	g	…..

Isolated vertices

A(2, M)/**A(3, M)**

5. List of successors

1	a	d		
2	b	a	c	d
3	c	b	d	e
4	d	e		
5	e	e		
6	f			
7	g			

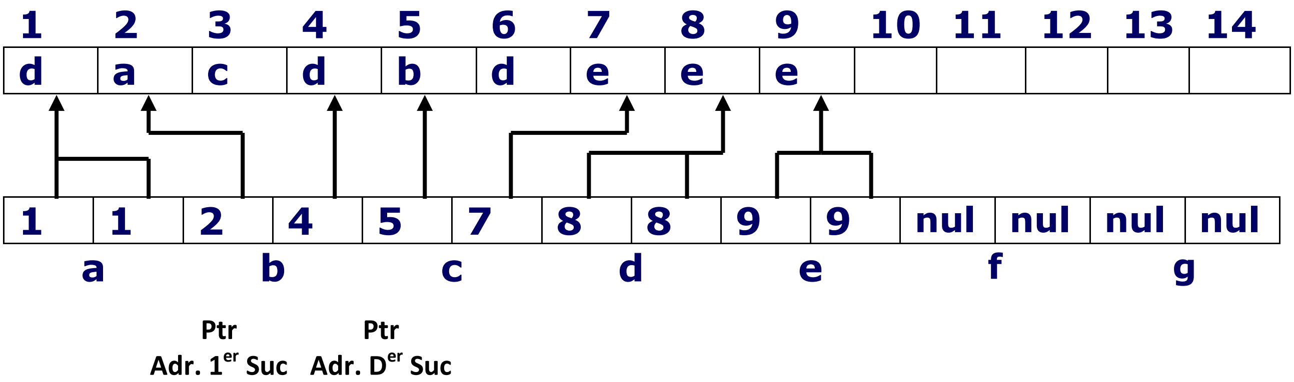
A(N, d^{+}_{max})

6. List of predecessors

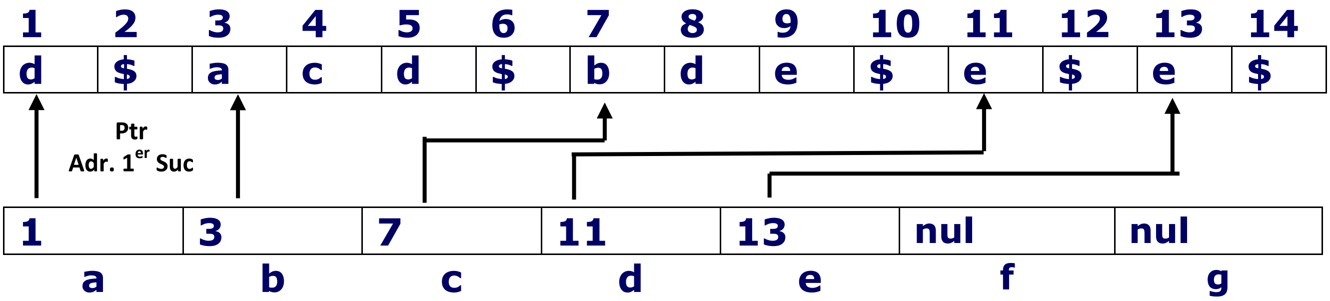
1	a	b		
2	b	c		
3	c	b		
4	d	a	b	c
5	e	e		
6	f			
7	g			

A(N, d^{-}_{max})

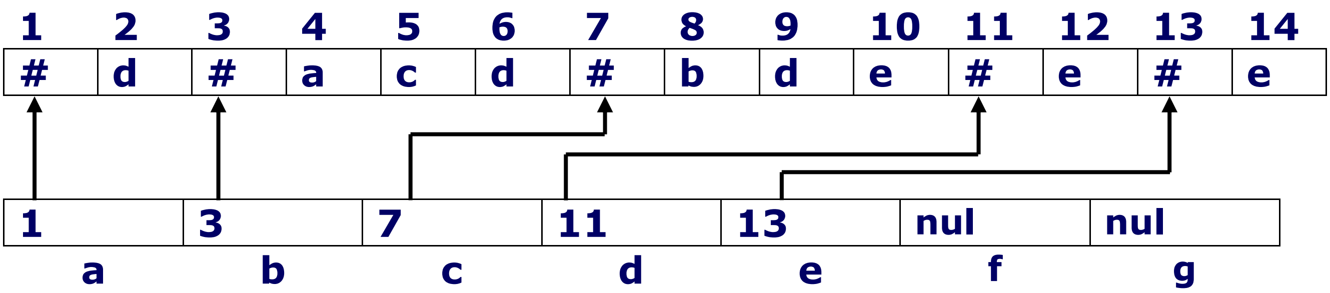
7. Linear List of successors



8. Linear List of successors with end of list (EOL)



9. Linear list of successors with begining of list (BOL)

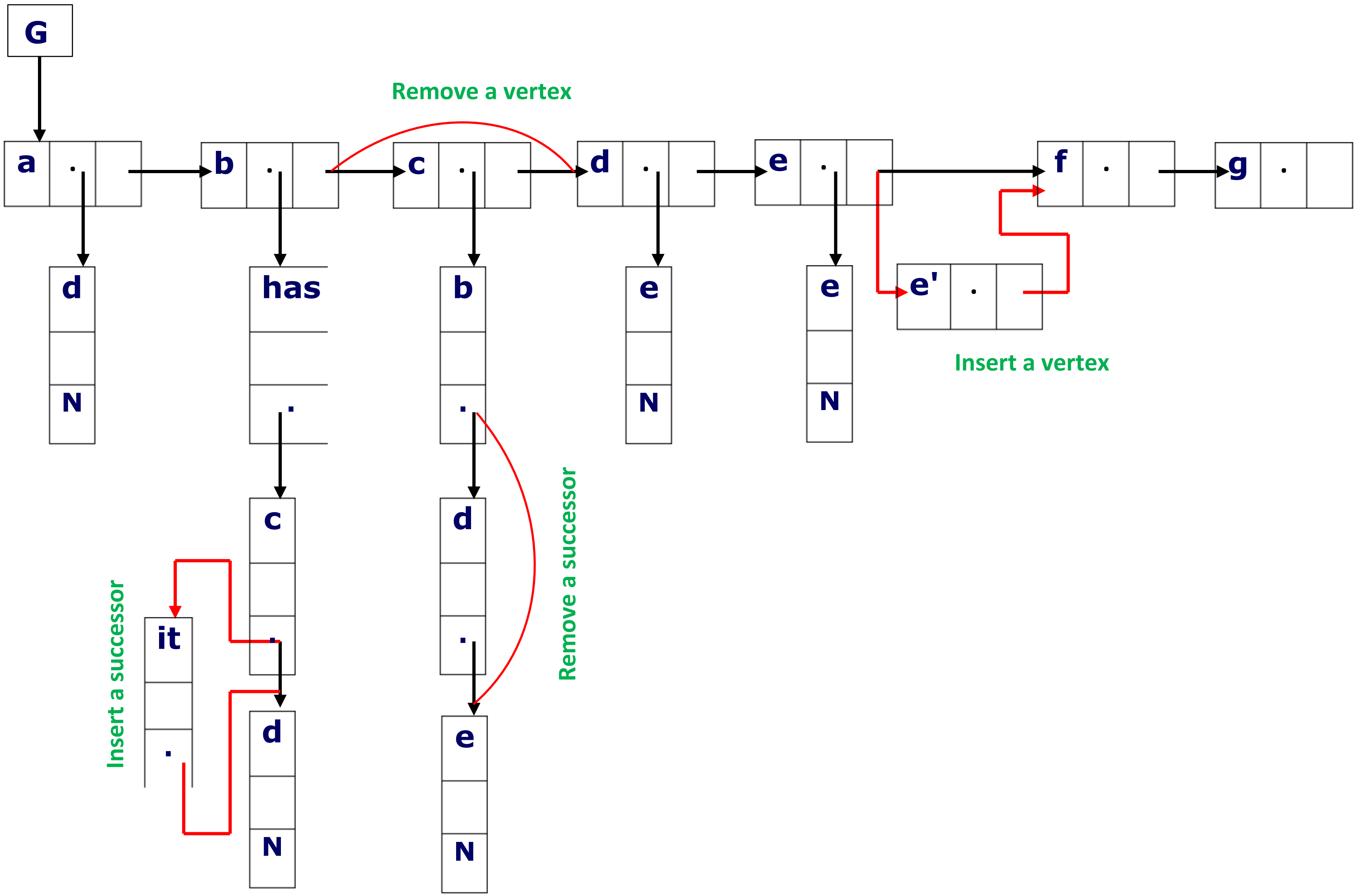


Dynamic methods

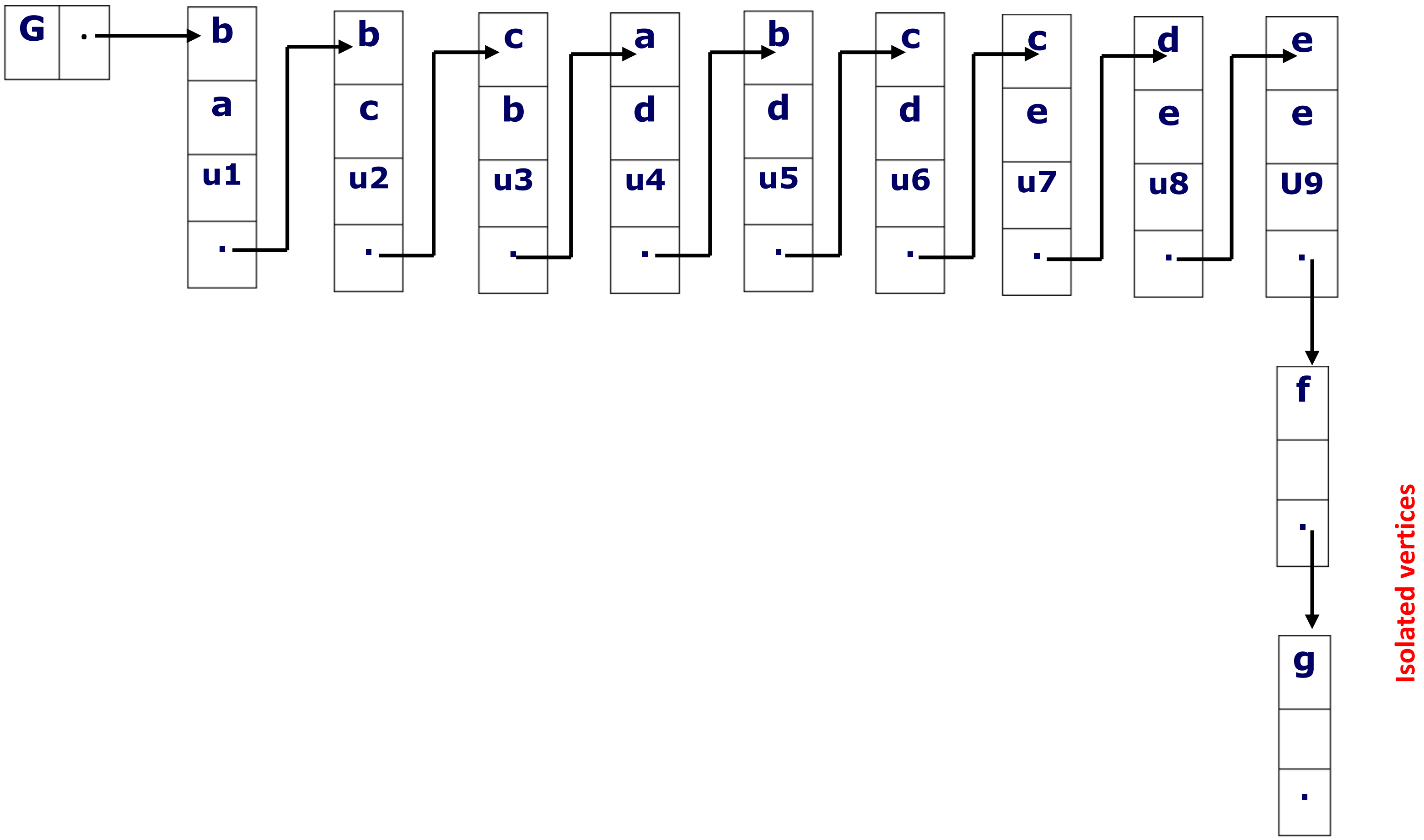
For each vertex we associate a sequence of boxes (boxes/nodes), each box contains an elementary information, such as:

- **Name of the vertex (a, b, c, 1, 2, …)**
- **Address of the 1st successor**
- **Address of the last successor.**
- **Number of successors**
- **Value of the arc.**
- **Others**

1. Dynamic list of vertices

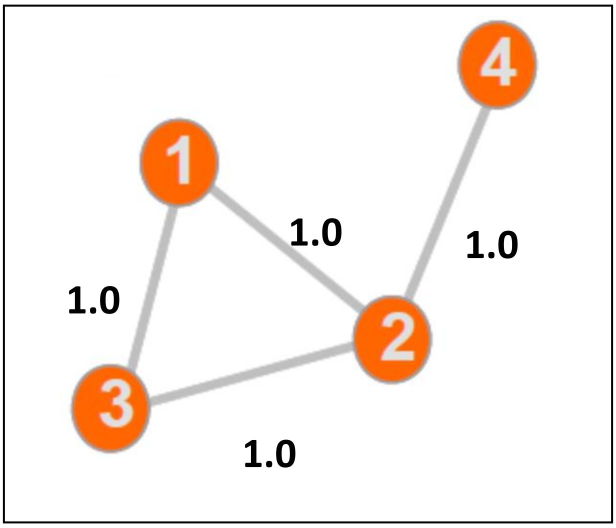


2. Dynamic list of arcs



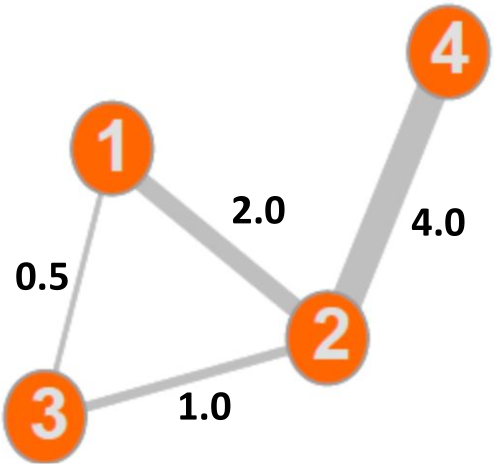
Methods to represent particular graphs

1. Unvalued Graph/Simple (Undirected)



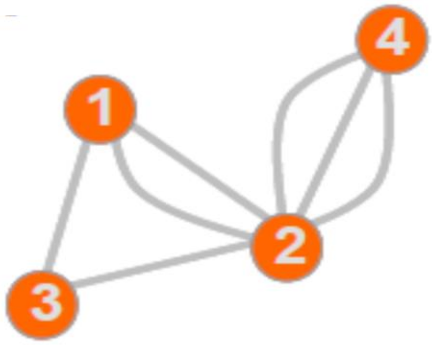
$$A_{ij} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

2. Valued Graph/Simple (Undirected)



$$A_{ij} = \begin{pmatrix} 0 & 2 & 0.5 & 0 \\ 2 & 0 & 1 & 4 \\ 0.5 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 \end{pmatrix}$$

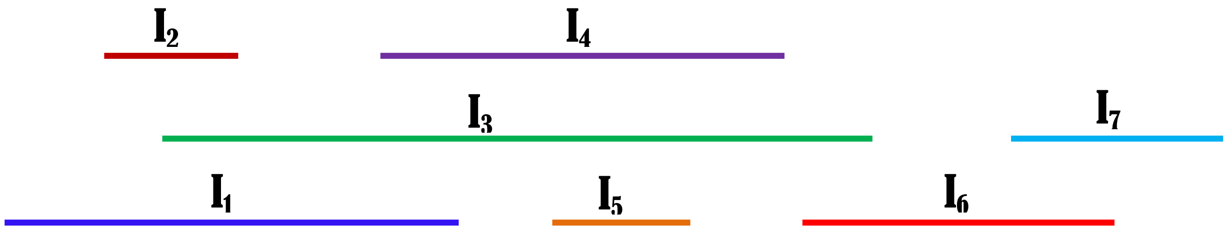
3. Multigraph (Undirected)



$$A_{ij} = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 3 \\ 1 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{pmatrix}$$

Interval Graphs

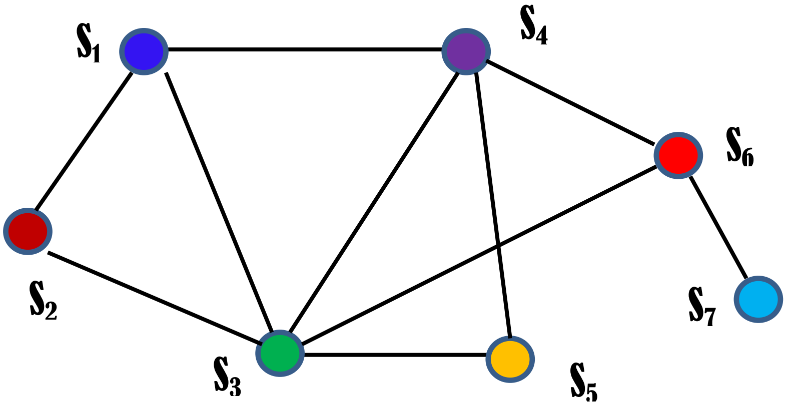
Let the intervals of the real line $I_1, I_2, I_3, \ldots, I_n$ be represented as follows:



We construct an undirected graph from the intervals $I_1, I_2, I_3, \ldots, I_n$ where the vertices of G are numbered as follows: $S_1, S_2, S_3, \ldots, S_n$.

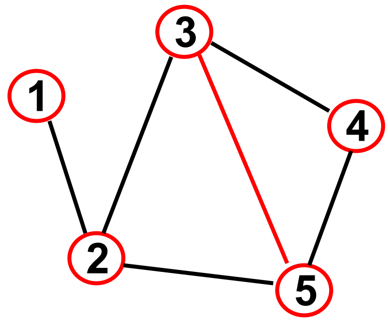
In the graph G, there exists an edge between the S_i and S_j $(i \neq j)$ if only if $I_i \cap I_j \neq \phi$

i.e., two vertices are connected if their associated intervals overlap.

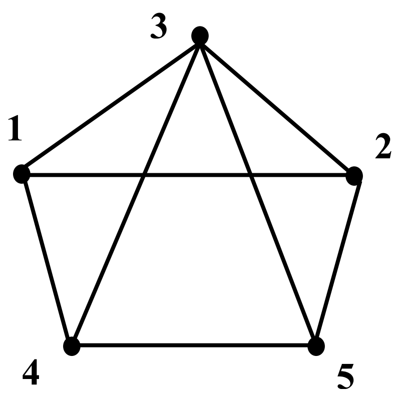


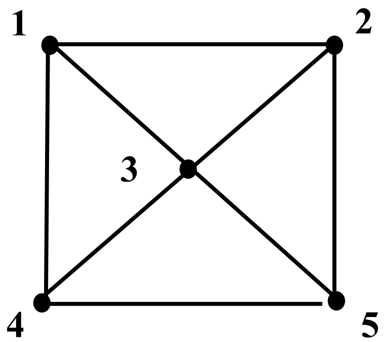
Triangulated Graphs

A graph is triangulated if all its cycles containing more than 3 vertices contain at least one chord (edge connecting two non-adjacent vertices of a cycle).



Planar graphs





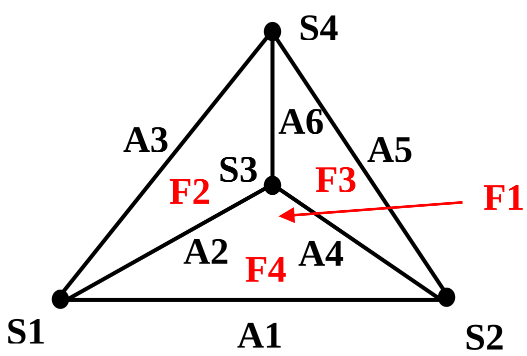
Euler's formula for a planar graph

$$V - E + F = 2$$

V: nb.vertices

E: nb.edges

F: nb.faces



Eulerian graphs and Hamiltonian graphs

Theorem 1:

- **A connected undirected graph has an Eulerian chain if only if the number of vertices of odd degree is equal to 0 or 2.**
- **It admits an Eulerian cycle if only if all its vertices have a even degree.**

  Necessary condition:

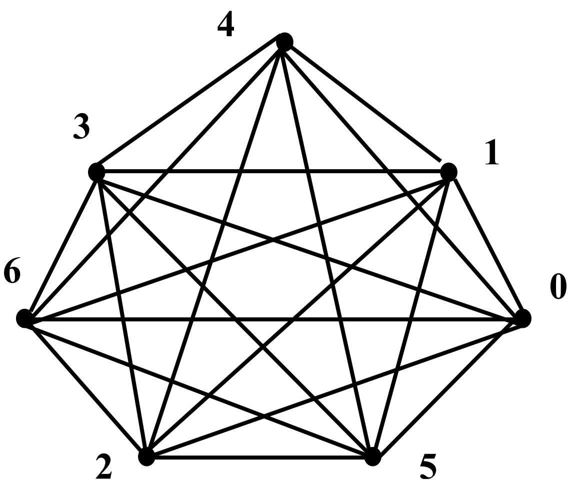
  - **For each vertex x, $d^-(x) = d^+(x)$ ==> d(x) is even**

Eulerian paths and circuits

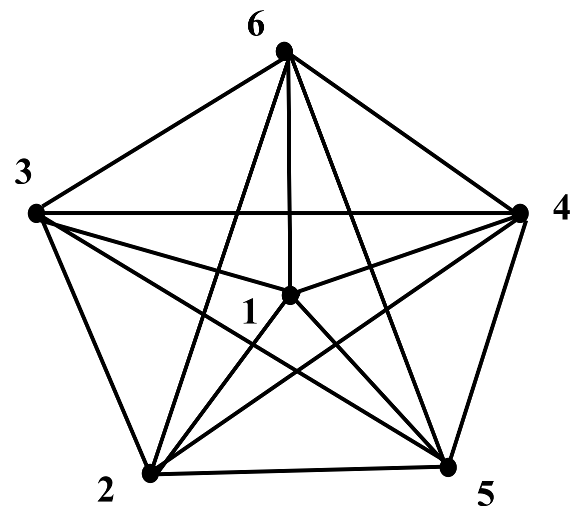
Theorem 2:

- **G admits an Eulerian circuit if only if $\forall x \in G$ $\boldsymbol{d^-(x) = d^+(x)}$**
- **A simple connected graph, G(X, U) is Eulerian if only if all its vertices are of even degree ($\forall x \in X$, d(x) is even)**

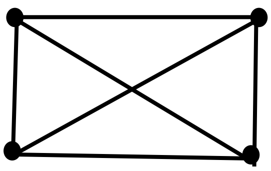
Examples



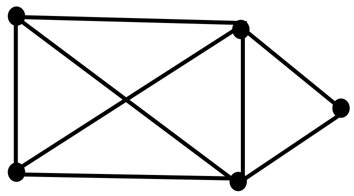
∀x∈X, d(x) = 6 even
G is Eulerian



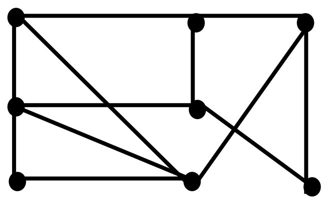
∀x∈X, d(x) = 5 odd
G is not Eulerian



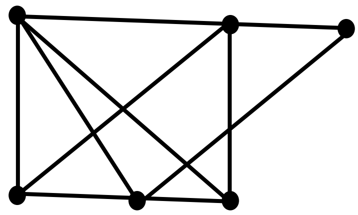
G is not Eulerian



G is Eulerian

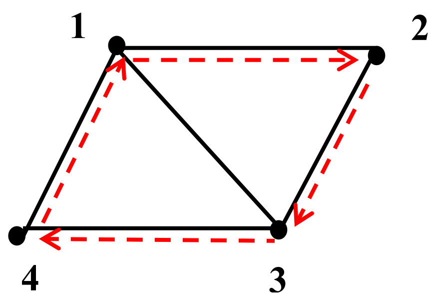


G is not Eulerian



G is Eulerian

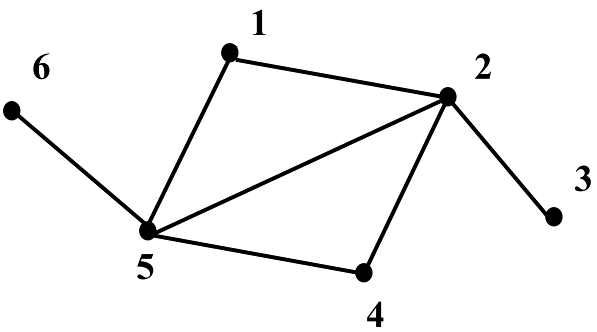
Hamiltonian cycles



∃ a Hamiltonian cycle

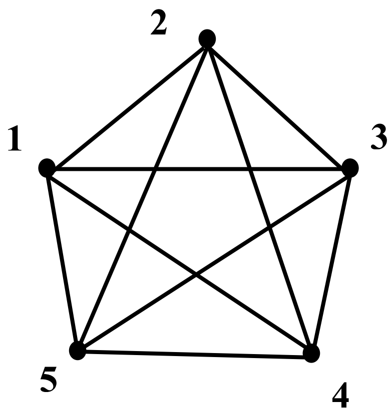
Observation

- **A graph with a vertex of degree 1 cannot be Hamiltonian**

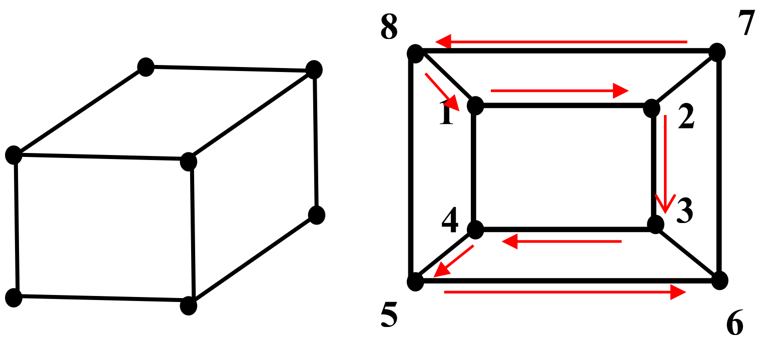


G is not Hamiltonian

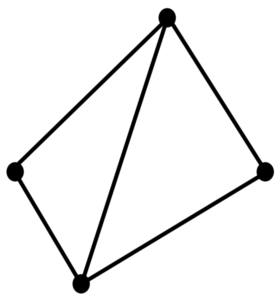
- **The complete graphs k_n are Hamiltonian if n > 3**



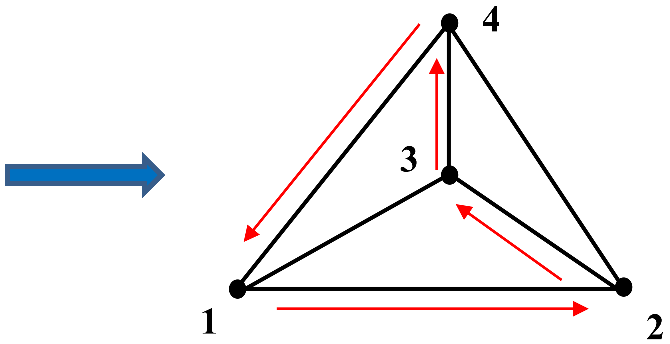
G is Hamiltonian (complete, k4)



G contains a Hamiltonian cycle



k3 type graph (pyramid)



G is Hamiltonian (complete, k3)