Graphs Theory

Basics (1)

University of M’sila
Faculty of Mathematics and informatics
Mathematics department

Specialty:

Applied Mathematics

L3

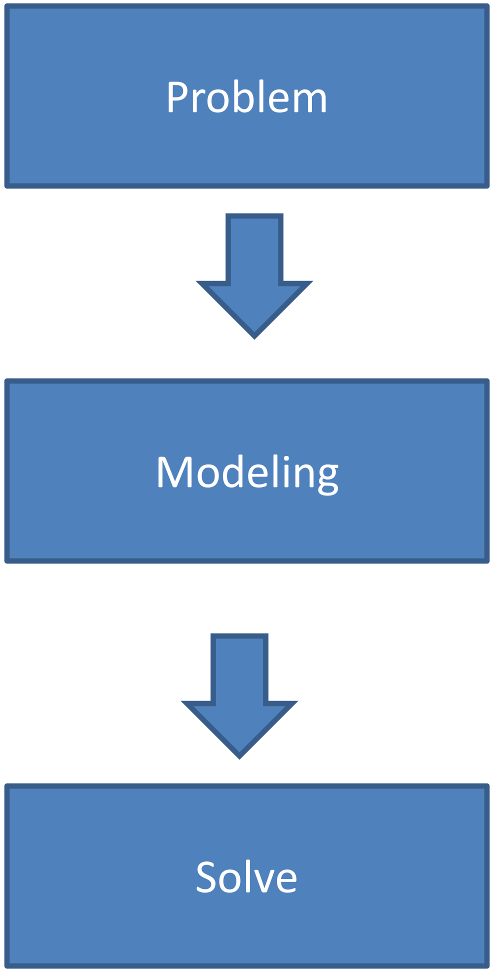
2024/2025

Dr.MOHAMED BAHACHE

Plan

- ❑ Introduction
- ❑ Definition of graph
- ❑ Order, size, and degree
- ❑ Successor and predecessor
- ❑ Type of graph
- ❑ Path, chain, cycle, and circuit
- ❑ Eulerian graph
- ❑ Hamiltonian graph
- ❑ Connectivity

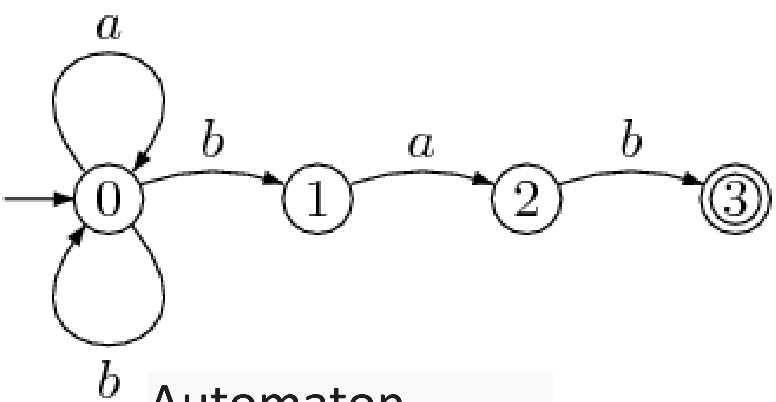
Introduction (motivation)



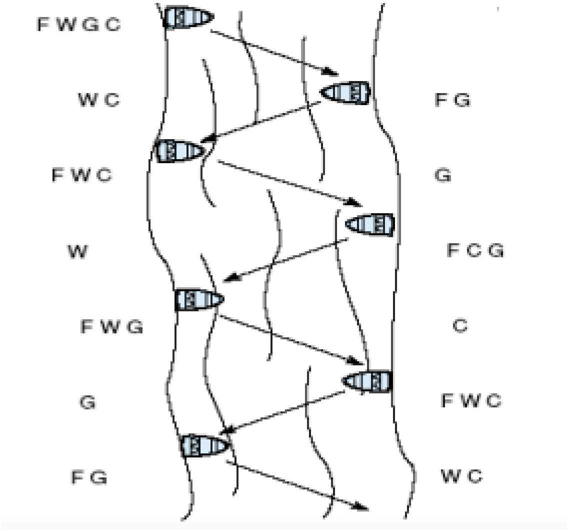
Introduction (motivation)

Road networks

Pipeline process



Automaton



Definition of graph

- A **graph** G=(V, E) is a pair where :
  - V is a set of vertices (or points or nodes)
  - E is a set of edges(or lines)
  - Each edge or lines has (joins) two incident ends.
  - These two incident ends are called adjacent (neighbors) vertices.
  - If e=(u,v); we say that the vertex u and the edge e are incident with each other.
  - If two edges are incident with a common vertex then they are called adjacent lines.
  - A graph with p vertices and q edges is called a (p, q) graph.
  - Edge joining vertex to itself is called a loop

Definition of graph

Example: for the following graph

V11

V={1,2,3,4,5,6,7}

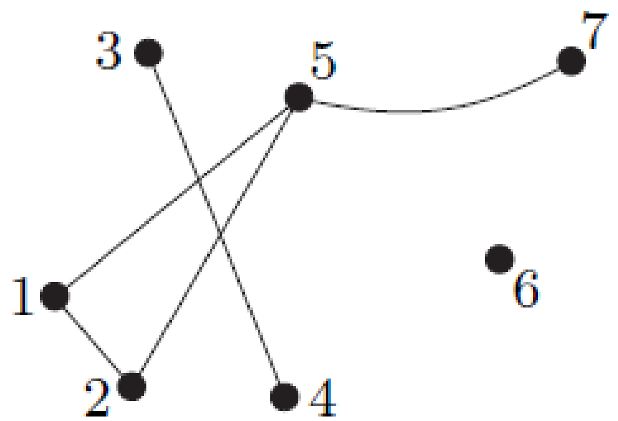
E={(1,2), (1,5), (2,5),
(3,4), (5,7),}

The vertices 1 and 2 are adjacent

We observe that:

$$E \subseteq V^2$$

$$E \cap V = \phi$$



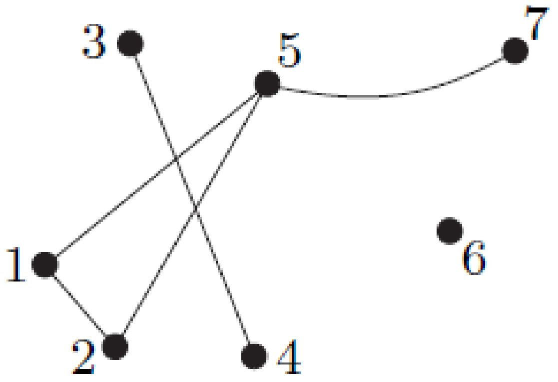
Definition of graph

The line (1,2) and the vertex 1 are incident with each other.

the lines (1,5), (2,5), are adjacent.

This graph is a (7,5) graph.

- **Non adjacent vertices or edges are called independent**



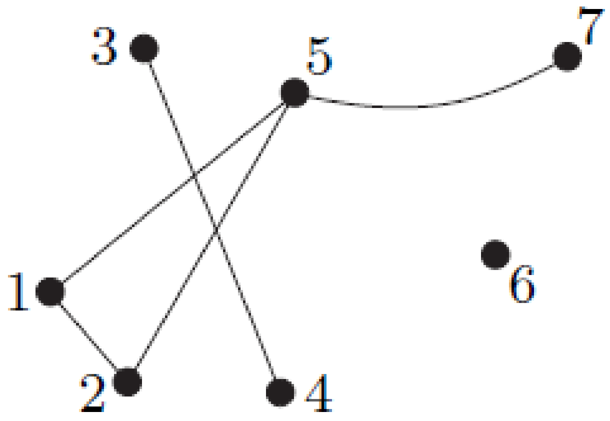
Order, size and Degree

- The **order** of a graph is its number of vertices, denoted by $|V|$
- The **size** of a graph is its number of edges, denoted by $|E|$

Example:

$$n = |V| = 7$$

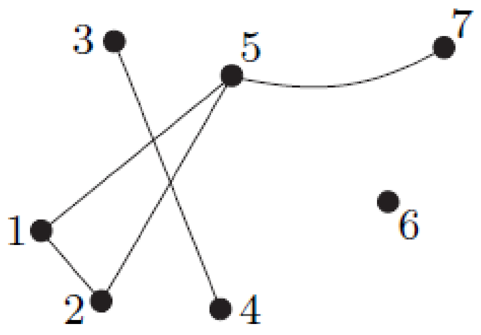
$$m = |E| = 5$$



Order, size and Degree

- Let G=(V,E) is a non empty graph
- We denote by $N_G(v)$ the set of Neighbors of a vertex v .
- We denote by $d_G(v)$, or simply $d(v)$ the set of neighbors of a vertex v .
- If $d(v) = 0$ the

  Vertex v is isolated .



d(1)=2

d(6)=0

Note: Loops are counted twice.

Order, size and Degree

- We denote by $\delta(G) = \min\{d(v) | v \in V\}$
The minimum degree of G .
- We denote by $\Delta(G) = \max\{d(v) | v \in V\}$
The maximum degree of G .

- The average degree of G: $d(G) = \frac{1}{|V|}\sum_{v \in V} d(v)$

$$\delta(G) \le d(G) \le \Delta(G)$$

Order, size and Degree

$$|E| = \frac{1}{2}\sum_{v\in V} d(v) = \frac{1}{2} d(G).|V|$$

❑**Proposition :** the number of vertices of odd degree in a graph is always even

Note : in the case of the oriented graph we split $d(v)$ in

$d^{+}(v)$: Number of edges having v as initial-endpoint

$d^{-}(v)$: Number of edges having v as terminal endpoint

$$d(v) = d^{+}(v) + d^{-}(v)$$

Order, size and Degree

- **If all the vertices of a graph G have the same degree K then G called K-regular or simply regular.**

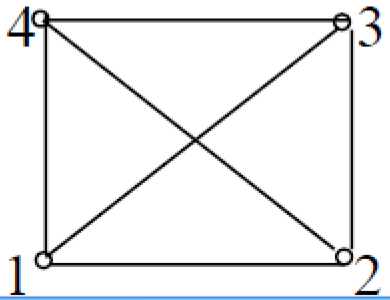
A 3-regular graph is called cubic

In a regular graph always we have :

$$\delta(G) = \Delta(G)$$

Example:

$$\delta(G) = \Delta(G) = 3$$



Theorem : Every cubic graph has an even number of points.

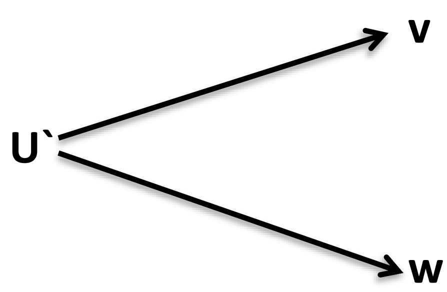
Set of successors,predecessors

- We denote by :
  - $\Gamma^{+}(v)$: set of successor of vertex v
  - $\Gamma^{-}(v)$: set of predecessor of vertex v
  - $\Gamma(v) = \Gamma^{-}(v) \cup \Gamma^{+}(v)$

Example:

v is a successor of u

u is a predecessor of v



$$\Gamma^{+}(u) = \{v, w\}$$

$$\Gamma^{-}(v) = \{u\}$$

Set of successors, predecessors

- if
  - $\Gamma^{+}(v) = \phi$ then: v is a **sink** vertex
  - $\Gamma^{-}(v) = \phi$ then: v is a **source** vertex
  - $\Gamma(v) = \phi$ then : v is an **isolated** vertex

Graph
Directed Vs undirected

Oriented	Non oriented
Edge	Line
Path	chain
Loop	Loop
Circuit	Cycle

Types of graphs

- **Valuated graph : Let G(V,E) be a graph (directed or undirected) we define an application**

$$f : E \to \Re$$

- **f is called valuation of the graph G.**

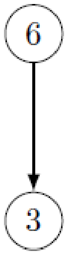
$$\forall (u,v) \in V^2 : (u,v) = +\infty$$

if $$(u,v) \notin E$$

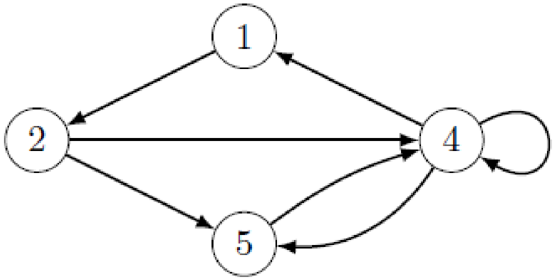
Types of graphs

- **Simple : Let G(V,E) be a graph, G is a simple graph if no loops and each pair of vertices are joined at most by one edge.**

$$\forall\left(u,v\right)\in V^{2}\exists at\ \mathrm{most}\left(u,v\right)\in E$$



- **Otherwise G is called multi-graph.**



Types of graphs

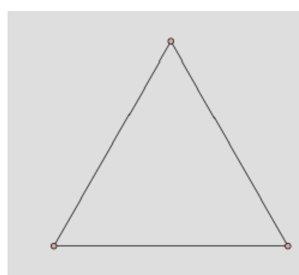
- **Complete graph : Let G(V,E) is a simple graph, G is** **complete** **iff**

$$\forall (u,v) \in V^2 : (u,v) \in E$$

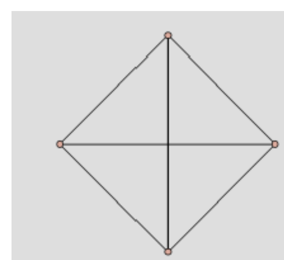
- **A complete graph with p vertices is denoted by**

$$K_p$$

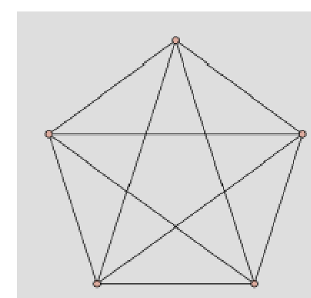
- **Example:**



K_3



K_4

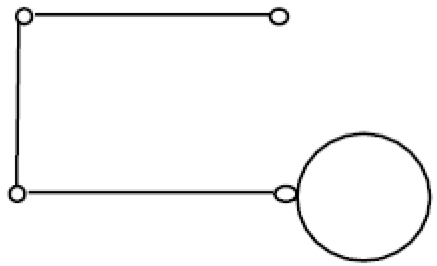


K_5

Types of graphs

- **If a vertex has multiple lines and loops then the graph is called PSEUDO GRAPH**

- **Example:**

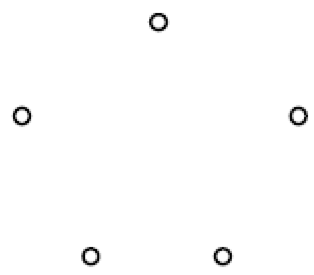


Types of graphs

- **Null graph : Let G(V,E) is a graph, G is a null graph iff:**

$$E = \phi$$

- **Example:**



G

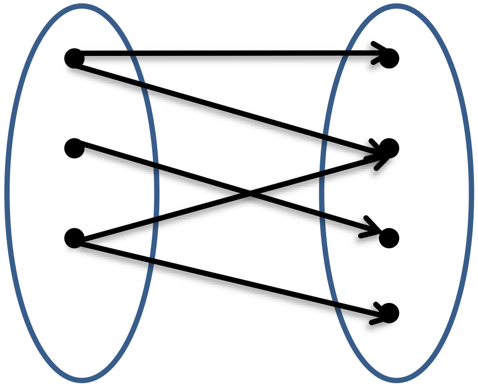
Types of graphs

- **BIPARTITE GRAPH : Let G(V,E) is a graph, G is** called a ***bigraph or bipartite graph*** ***if the vertex set V can be partitioned into two disjoint subsets V1 and V2 such that every line of G joins a point of V1 to a point of V2. (V1, V2) is called a bipartition of G.***
- ***Formaly*** $V_1 \cap V_2 = \phi$ ***and*** $V_1 \cup V_2 = V$
- ***If*** $(u, v) \in E \text{ and if } u \in V_1 \Rightarrow v \in V_2$

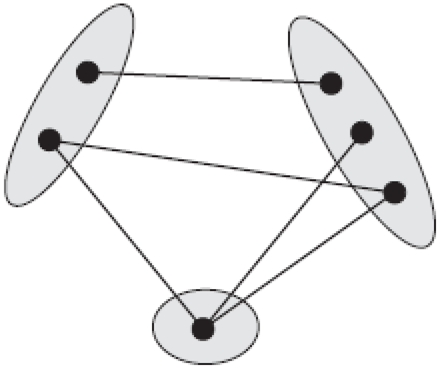
and vice versa

- **Example:**

Types of graphs



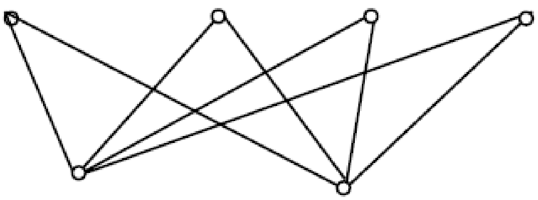
2-partite graph
Or simply bipartite graph



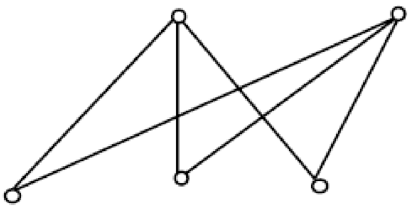
3-partite graph

Types of graphs

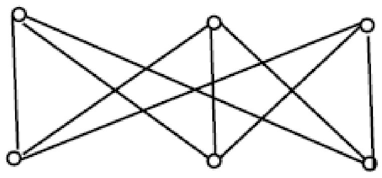
$K_{4,2}$



$K_{2,3}$



$K_{3,3}$



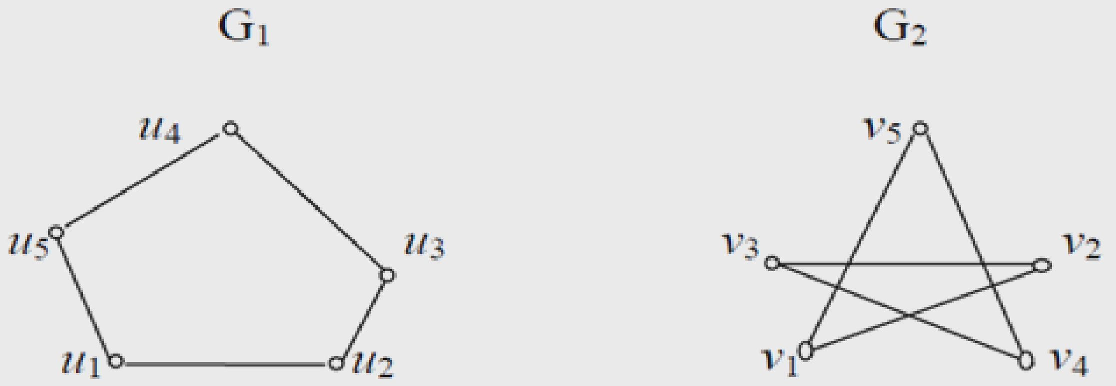
Complete bipartite graph

Proposition : a graph is a bipartite iff it contains no odd cycle

Types of graphs

❑ISOMORPHISM

Definition: Two graphs G1 = (V1, X1) and G2 = (V2, X2) are said to be *isomorphic* *if there exists a bijection f :V1 → V2 such that u, v ∈ V1 are adjacents in G1 iff f(u) , f(v) ∈ V2 are adjacents in G2.*

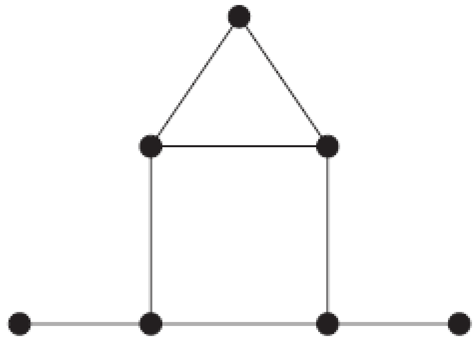


G1 and G2 are isomorphic

$f(u_1) = v_1$, $f(u_2) = v_2$, $f(u_3) = v_3$, $f(u_4) = v_4$, $f(u_5) = v_5$

Types of graphs

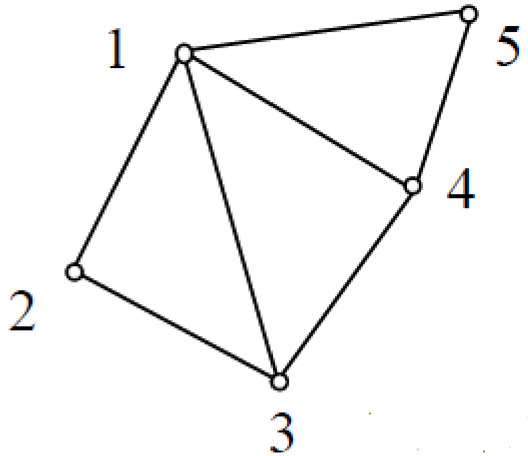
- ***Crossing number of a graph G is the minimum number*** of pair wise intersections of the edges when G is drawn in the plane.
- A planar graph is called ***outer planar if it can be embedded*** in the plane so that all its vertices lie on the same face.
- Note: The crossing number of a planar graph is zero.

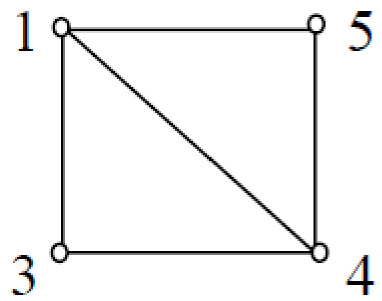


Types of graphs

❑ ***Sub graph***

Definition: A graph H = (V1, X1) is called a *sub graph of* G (V, X) if V1 ⊆ V and X1 ⊆ X.

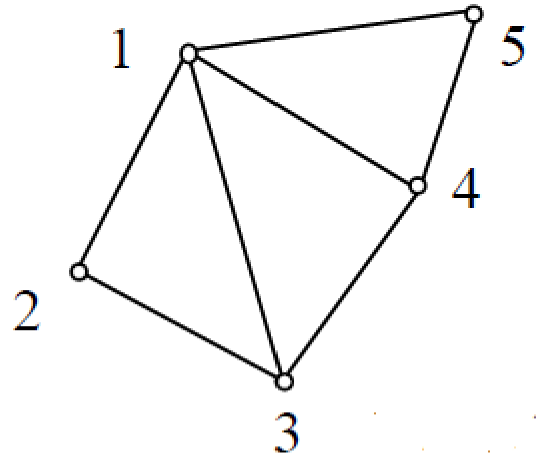


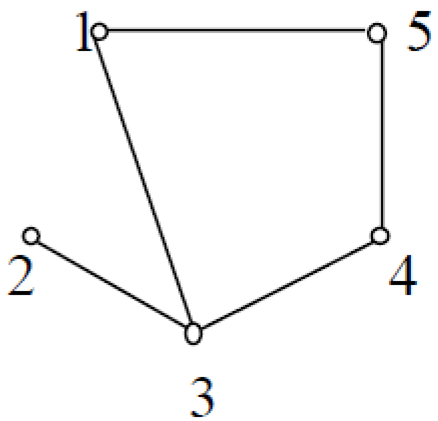


Types of graphs

❑PARTIAL GRAPH

Definition: Let Two graphs G1 = (V1, X1) and G2 = (V2, X2) , G2 is a partial graph of G1 If V1=V2, and $E_2 \subset E_1$





Types of graphs

❑**REMOVAL OF A POINT :**

▪**Definition: Let G = (V, X) be a graph and *v ∈ V. The subgraph of G obtained by removing the point v and all the lines incident with v is called the subgraph obtained by the removal of the point v and is denoted by G – v.***

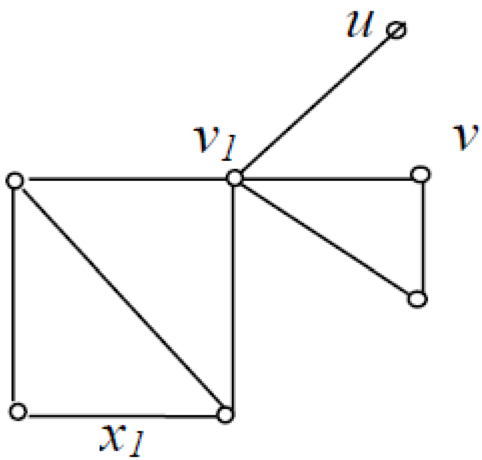
(G – *v is an* *induced subgraph* *of G.)*

❑**REMOVAL OF A LINE**

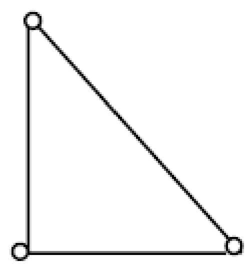
▪**Definition: Let G = (V, X) be a graph and *x ∈ X. Then G – x = (V, X –{x}) is called the* *spanning subgraph* *of G obtained by the removal of the line x .***

(G – *x is a spanning subgraph of G **)***

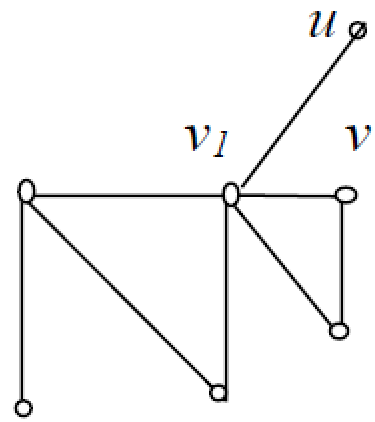
Types of graphs



G



$\mathbf{G} - v_1$



$\mathbf{G} - x_1$

Path, chain,circuit,cycle

❑Oriented graph

Definition: Let G = (V, E) is an oriented graph

A path from vertex u to vertex v is a sequence

<s0,s1,…,sk> where

S0=u and sk=v and $\left(s_i, s_j\right) \in E \; \forall i \in \{0,1,\ldots,k\}$

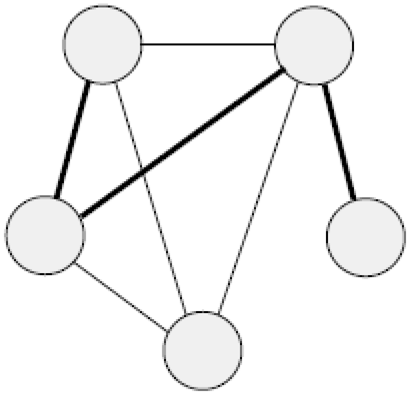
We can express this path either by:

<s0,s1,…,sk> or by <(s0,s1),…,(sk-1,sk)>

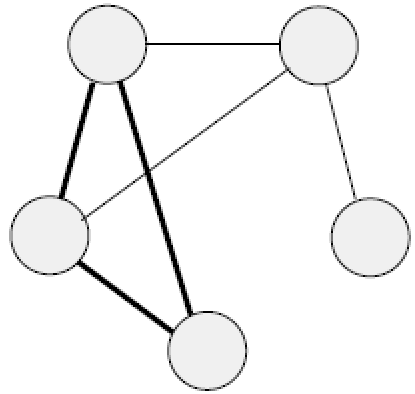
The length of this path is the number of edges that contains

Path, chain, circuit, cycle

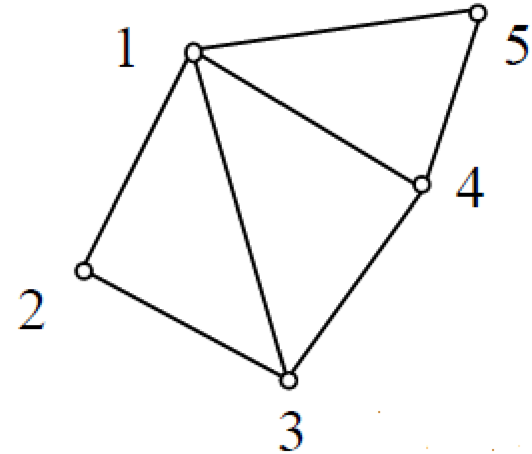
- Simple path: a path where all the edges that contains are different
- Elementary path: a path where all the vertices that contains are different.
- A path <s0,s1,…,sk> is a circuit if s0=sk and if k>=1, where k is the number of edges.
- This circuit is simple if all the edges that contains are different.
- This circuit is elementary if s1,…sk are all distinct.
- A loop is a circuit whose length 1.



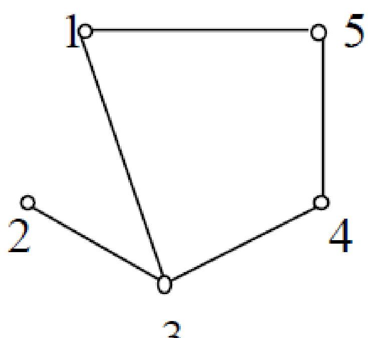
Path in connected graph



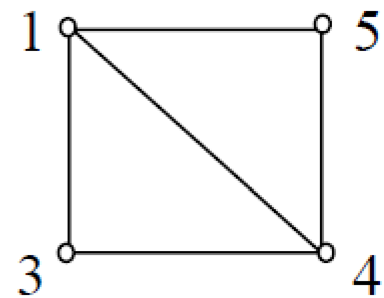
Cycle in connected graph



G1

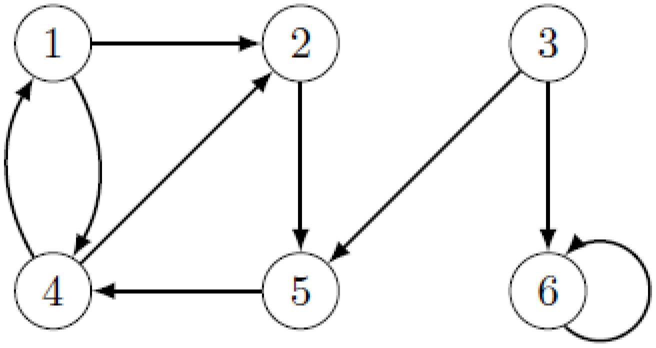


Partial graph of G1



Sub graph of G1

Path, chain, circuit, cycle



Elementary path is < 2,5,4,1 >.

no elementary path is < 2,5,4,1,4 >..

elememtary circuit is < 1; 2; 5; 4; 1 >.

no elemetary circuit < 1; 2; 5; 4; 2; 5; 4; 1 >.

Simple path is <1,2,5,4>

No Simple path is <1,2,5,4,1,2>

Simple circuit is <1,2,5,4,1>

No simple circuit is <1,2,5,4,1,4,1>

Path, chain, circuit, cycle

- In the case of no oriented graph we talk chain rather than path, cycle rather than circuit.

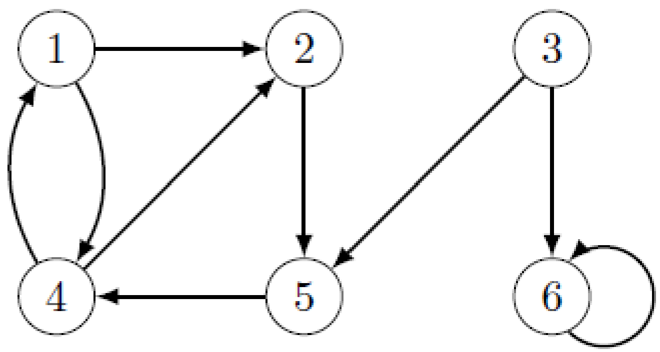
- **Theorem(cycle)** : **If G is a graph in which the degree of every vertex is at least two then G contains a cycle.**

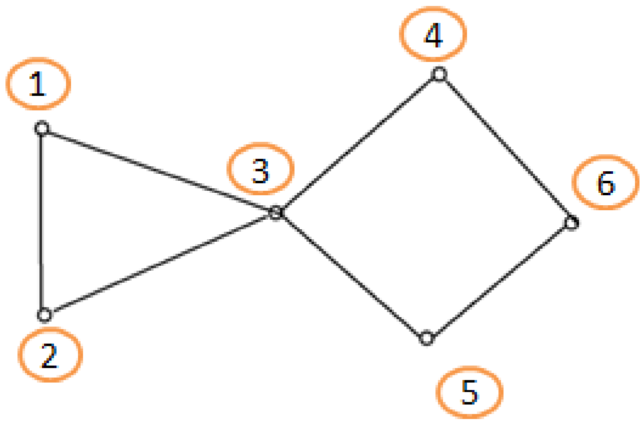
Path, chain, circuit, cycle

- **Hamiltonian path**: elementary path containing all the vertices of the graph.
- **Eulerian path**: simple path containing all the edges of the graph.
- **Hamiltonian graph**: graph containing an Hamiltonian cycle.
- **Semi Hamiltonian graph**: graph containing an Hamiltonian chain.
- **Eulerian graph**: graph containing an Eulerian cycle.
- **Semi Eulerian graph**: graph containing an Eulerian chain.

Path, chain, circuit, cycle

Example:





No Hamiltonian path

No Eulerian path

Hamiltonian path <1,2,3,4,6,5>

Eulerian cycle <1,3,5,6,4,3,2,1>

Eulerian graph

❑Theorem (cycle) : A connected graph is Eulerian if and only if every vertex has even degree.

❑Theorem(chain): A connected simple graph is semi-Eulerian iff It has 0 or exactly 2vetices have an odd degree

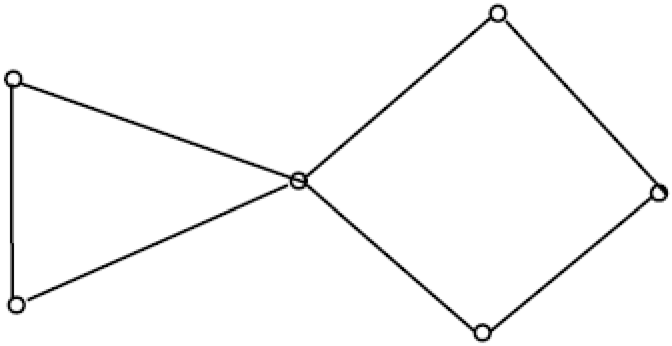
❑Theorem(Eulerian circuit): An oriented multigraph strongly connected contains an eulerian circuit iff: $d^{+}(s) = d^{-}(s)\ \forall s \in V$

❑Theorem(Eulerian path): An oriented and connected multigraph contains an Euleurian path from the vertex u to the vertex v iff:

$$d^{+}(u) = d^{-}(u) + 1 \qquad d^{+}(v) = d^{-}(v) - 1 \qquad d^{+}(s) = d^{-}(s)$$

Eulerian graph

Example:

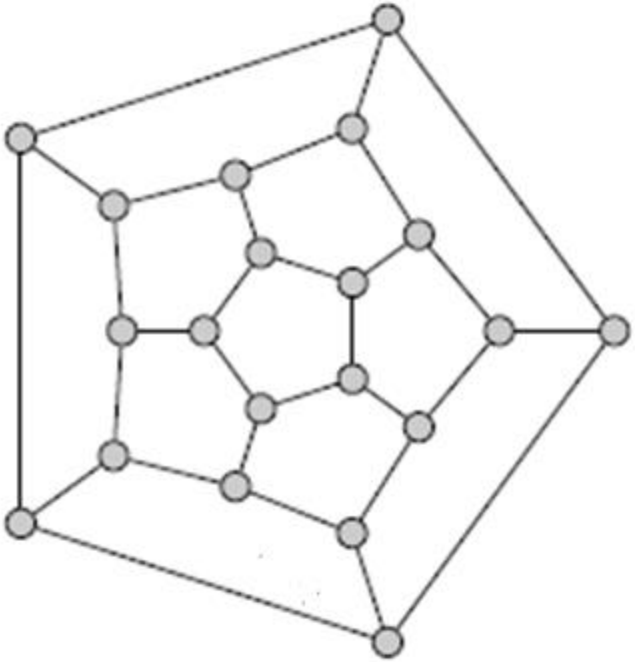


Hamiltonian graph

- **Simple sufficient conditions:**
  - A graph with vertex of degree 1 not Hamiltonian
  - If a vertex in a graph has degree 2 , then the two edges Hold to Hamiltonian cycle.
  - The K_n graphs are Hamiltonians.
  - Theorem (Ore): Let G be a simple graph with *p vertices , p≥3, if for all non adjacent two vertices u and v such that d(u) + d(v) ≥ p. Then G is Hamiltonian.*
  - Theorem (Dirac):
  - If G is a graph with *p ≥ 3 vertices and δ ≥ p / 2 then G is Hamiltonian.*

Hamiltonian graph

- Example:



Connectivity

- A non empty graph G is connected if :

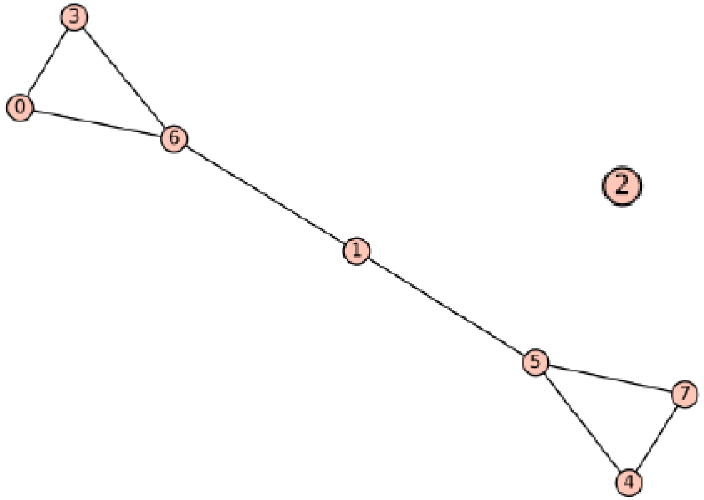
$$\forall u, v \in V \Rightarrow \exists \text{ chain from } u \text{ to v}$$

- A non empty graph G is strongly connected if :

$$\forall u, v \in V \Rightarrow \exists \text{ path from } u \text{ to } v$$

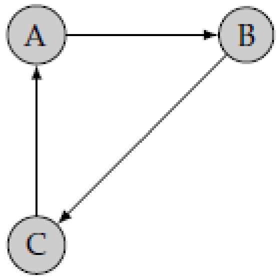
$$\text{and } \exists \text{path from } v \text{ to } u$$

Connectivity

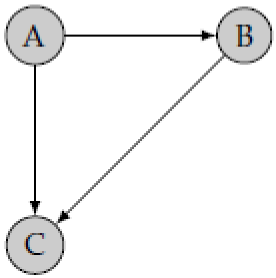


Disconnected graph
But it contains two connected component(sub graphs)

Connectivity



Strongly connected graph



Simply connected graph

Simply connected component

Simply connected component research Algorithm

Let G=(V,E) be a graph, s∈ V

SCC(s)=ϕ;

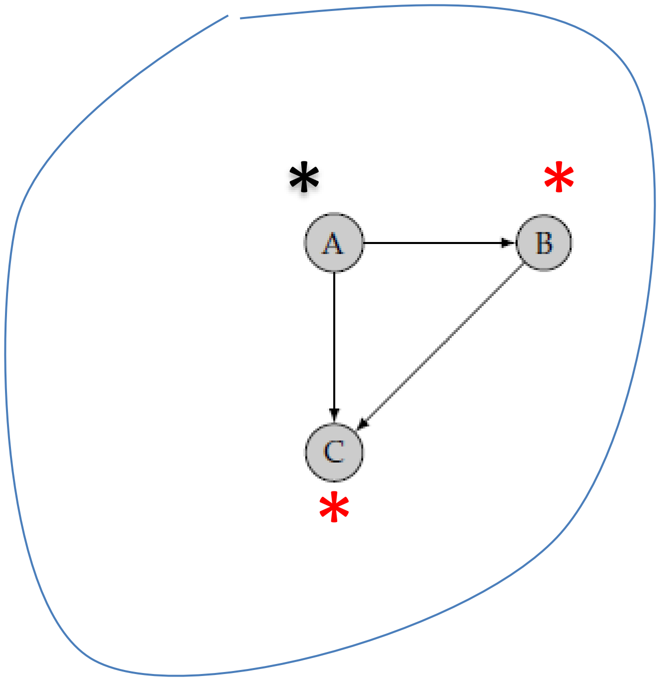
1-Put (*) on the vertex s;

2-Put (*) on all adjacent vertices of a marked vertex;

3-repeat 2 until no mark is possible;

4-all the vertices having * mark construct a SCC(S)

Simply connected component



Strongly connected component

Strongly Connected Component Research Algorithm

Let $G=(V,E)$ be a graph, $s \in V$

$StCC(s)=\phi$;

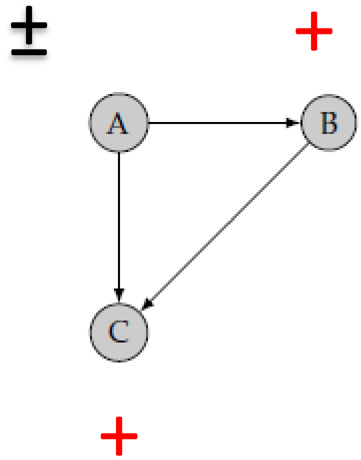
1-Put ($\pm$) on the vertex s;

2-Put (+) on all successors not yet marked of all vertices marked by (+)

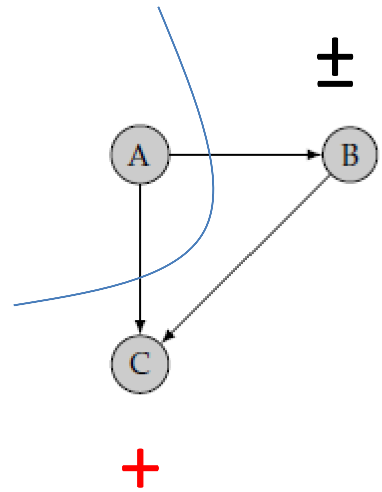
3-Put (-) on all predecessor not yet marked of all vertices marked by (-)

4-All the vertices marked by ($\pm$) construct an $StCC(s)$.

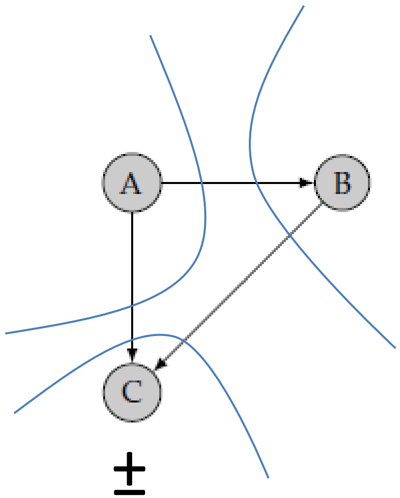
Strongly connected component



Strongly connected component



Strongly connected component



Strongly connected component

- **CUT POINT**

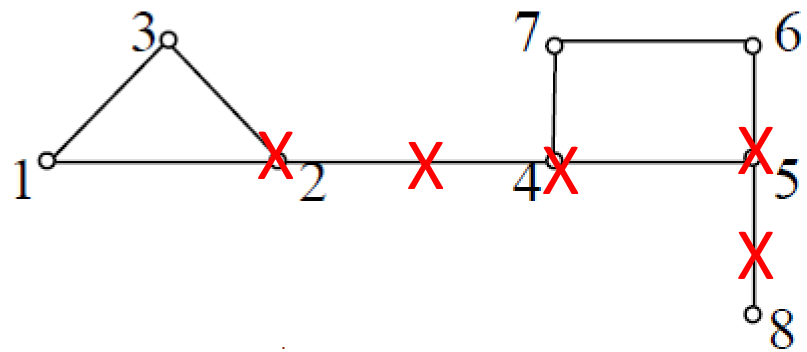
  Is ***a vertex whose removal increases the number of connected components.***

- **BRIDGE**

  Is an edge whose removal increases the number of connected components.

Note: If *v is a cut point of a connected graph then G – {v} is disconnected.*

Example:



2, 4, 5 are cut points.
{2, 4} and {5, 8} are bridges.

References

1-DIESTEL, Reinhard. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.
2- Sri Chandrasekharendra Saraswathi Viswa Mahavidhyalaya, Graph theory
3-Introduction à la théorie des graphes
Didier Müller
4- Mathieu SABLIK , Graphe et langage.