Week 2

Graph Theory 101

Previous Class Discussion:

Avatars of Graph!

3.1 Subgraph

A subgraph of a graph **G** is a graph **H** such that $\mathbf{V(H) \subseteq V(G)}$ and **E(H) ⊆ E(G)**. Also **H ⊆ G**.

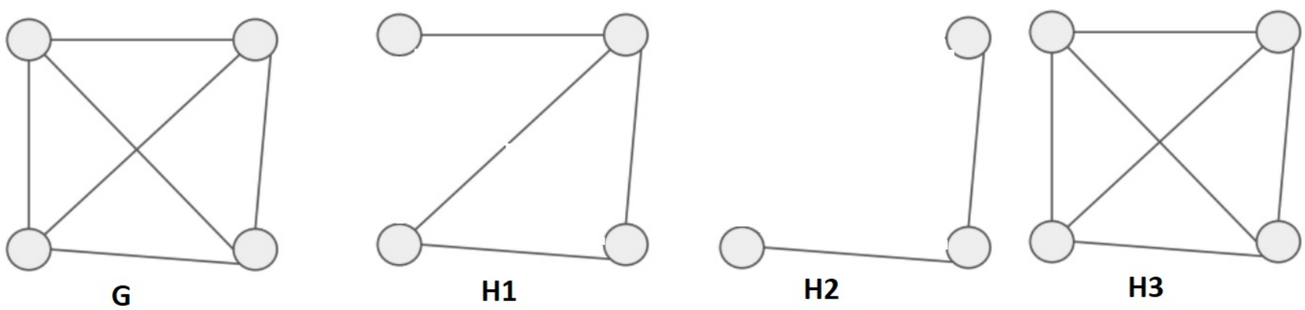
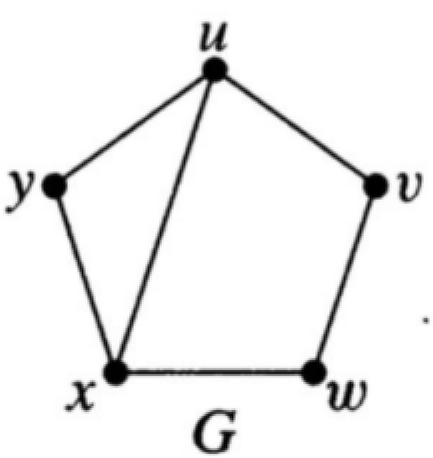


Figure: **H1 ⊂ G, H2 ⊂ G, H3 ⊆ G**

3.2 Independent Set and Clique

→ A set of vertices is called **independent** if no two vertices in the set are adjacent.

→ A set of vertices is called a **clique** if every two vertices in the set are adjacent.

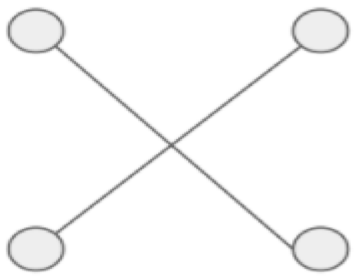


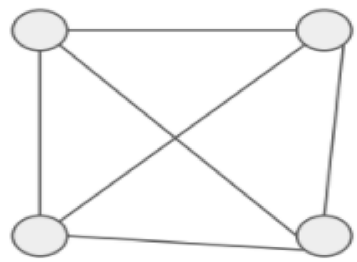
Independent Set: {u, w}, {v, x}, {y, v}, {y, w}

Clique: {x, u, v}

3.3 Chromatic Number

Smallest number of colors needed to color the vertices of a Graph G(V, H) such that no two adjacent vertices have the same color. Chromatic Number= 2



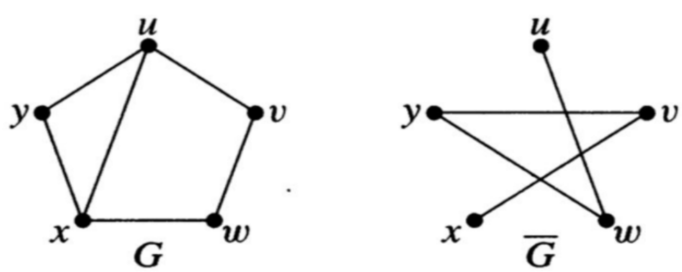


Chromatic Number = 2 Chromatic Number = 4

3.4 Complement of a graph

A complement of a graph **G** is a graph **G'** such that **V(G) = V(G')** and fills all missing edges and removes existing edges from a complete graph.

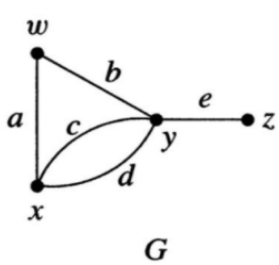
Or Draw edges between all pairs of vertices of an **independent set.**



3.5 Graphs Representation

3.5.1 Adjacency Matrix

An adjacency matrix of a graph G is square matrix of size **|V|** X **|V|** in which i-j th element denotes number of edges between ith and jth vertices (|V| = number of vertices). It is denoted as A(G).



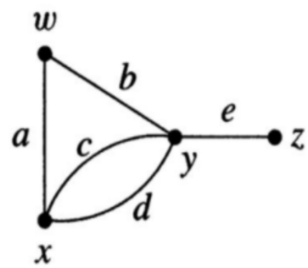
$$\begin{array}{c c} & \begin{array}{cccc} w & x & y & z \end{array} \\ \begin{array}{c} w \\ x \\ y \\ z \end{array} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

$A(G)$

Advantage:
→ Quickly check if two nodes have a direct edge or not.
→ Best representation if graph is almost complete
Disadvantage:
→ Large memory complexity.

3.5.2 Incident Matrix

An incident matrix of a graph G is a matrix of size **|V|** X **|E|** in which i-j th element = 1 if i th vertex is an endpoint of j th edge (|V| = number of vertices, |E| = number of edges). It is denoted as M(G).



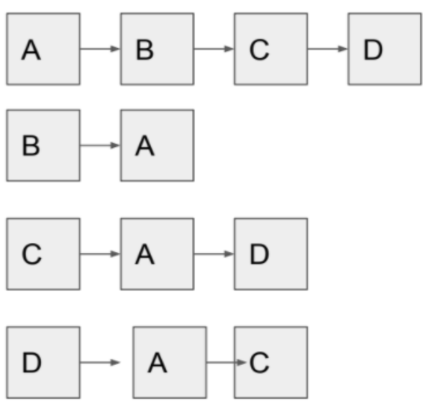
G

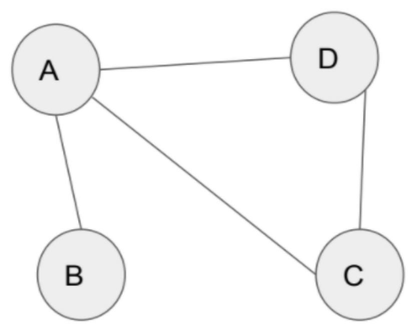
$$
\begin{array}{c}
 \\ w \\ x \\ y \\ z
\end{array}
\begin{array}{c}
\begin{array}{ccccc} a & b & c & d & e \end{array} \\
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\end{array}
$$

$M(G)$

3.5.2 List

It is represented as a linked list of neighbours.





3.6 Notations

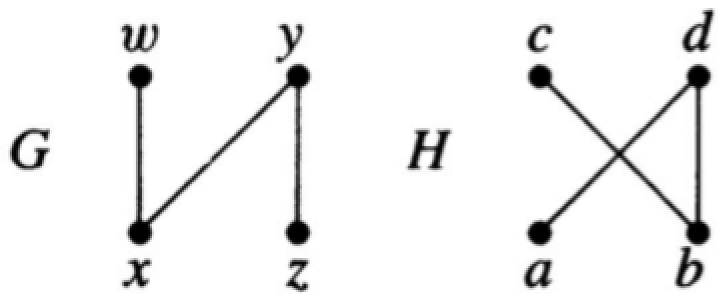
- P : Path with n vertices
- C : Cycle with n vertices
- K : Complete Graph with n vertices
- K , : Bipartite graph between two independent set
- N : Null graph
- W : Wheel Graph
- Q : Hypercube Graph

3.7 Isomorphism

Two graphs G and H is isomorphic if and only if it satisfy following condition

1. Number of vertices in both graph are same
2. Number of edges in both graph are same
3. Their edge connectivity is retained

If G is isomorphic to H then it is denoted as **G ≅ H**.



In this example
Condition 1 → SATISFIED
Condition 2 → SATISFIED
Condition 3 → SATISFIED

Now, How is condition 3 satisfying? To prove it we need some function **f()** such that,

f(w) → c
f(x) → b
f(y) → d
f(z) → a

Isomorphic graph follow following conditions:

1. **Reflexive**

   ∀ G, G1 ≅ G1

   We can use a bijection f such that f(V1) → V1

2. **Symmetric**

   If G1 ≅ G2 then G2 ≅ G1

   Proof:
   Since, G1 ≅ G2
   ⇒ ∃ a bijection f such that f(V1) → V2
   And ∃ a bijection f^{-1} such that f^{-1}(V2) → V1
   ⇒ G2 ≅ G1

3. **Transitivity**
   If $G1 \cong G2$ and $G2 \cong G3$ then $G1 \cong G3$

   Proof:

   Since, $G1 \cong G2$ and $G2 \cong G3$

   $\Rightarrow$ $\exists$ f, g bijection such that

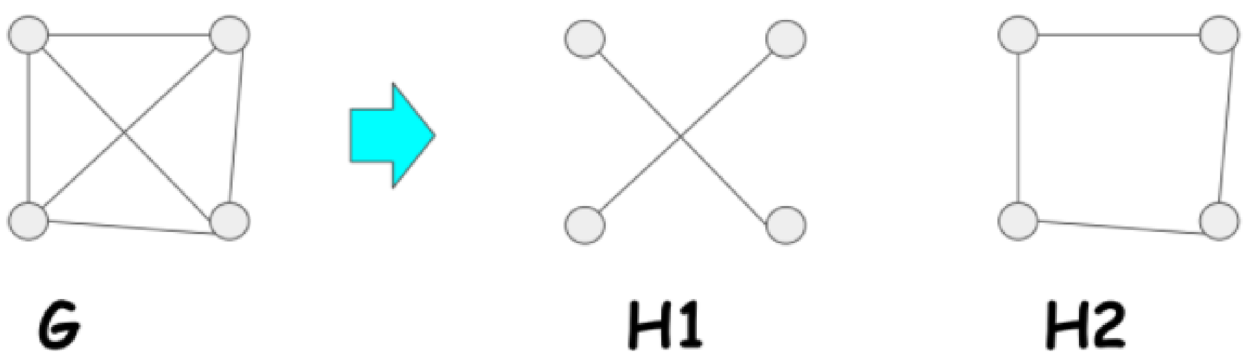
   $\Rightarrow$ $f(V1) \rightarrow V2$ & $g(V2) \rightarrow V3$

   $\Rightarrow$ $\exists$ gof a bijection such that $gof(V1) \rightarrow V3$

   $\Rightarrow$ $\mathbf{G1 \cong G3}$

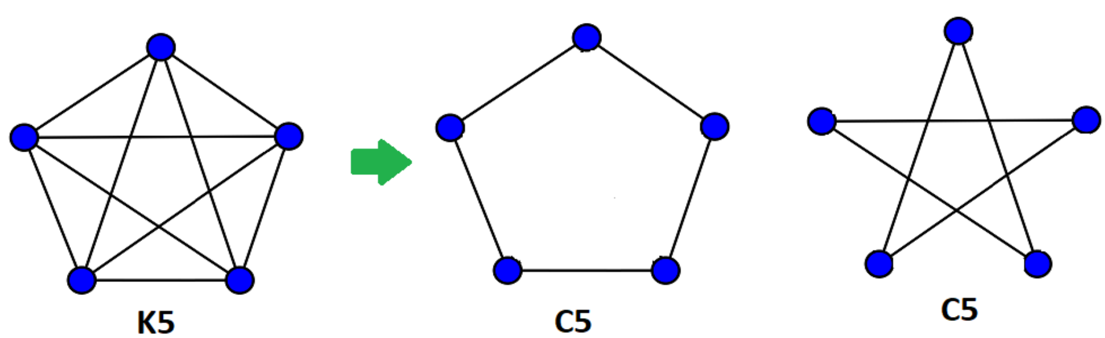
3.8 Decomposition

A decomposition of a graph is a list of subgraphs such that each edge appears in exactly one subgraph in the list.



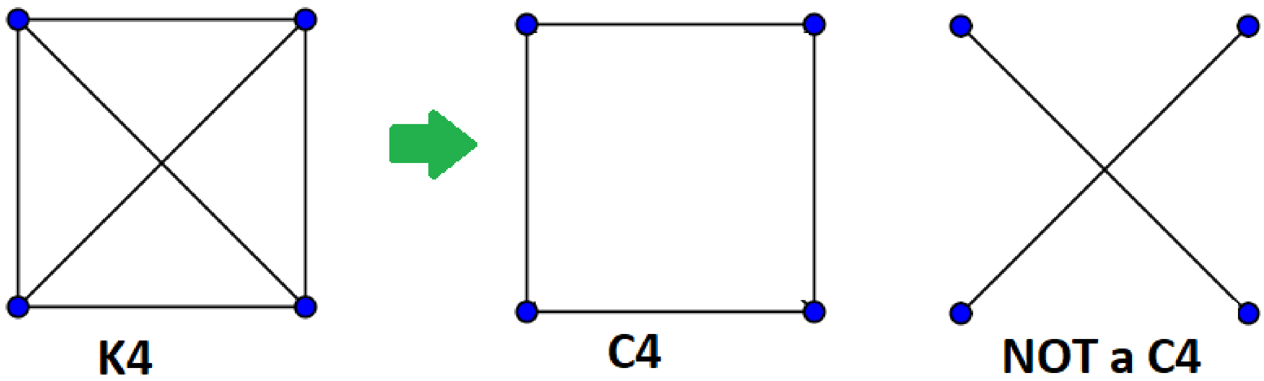
Q. Can K5 be decomposed into two C5?

Ans: **YES**



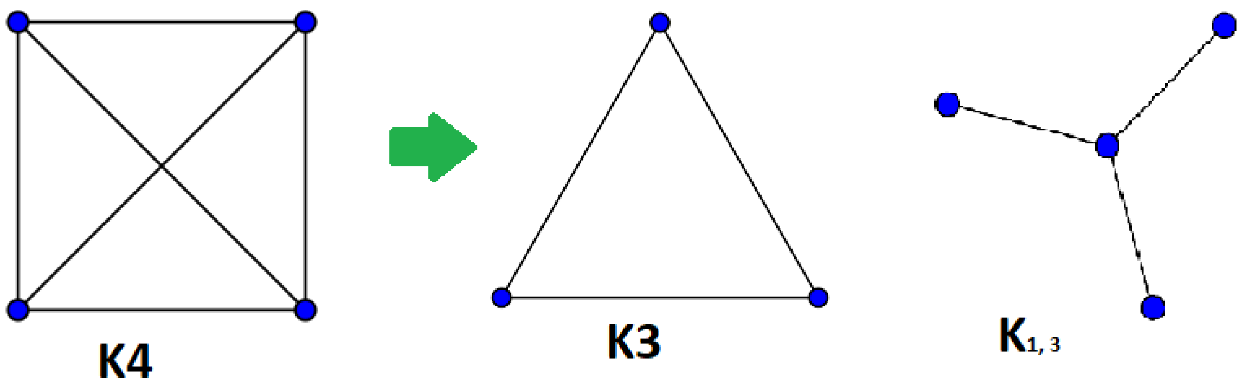
Q. Can K4 be decomposed into two C4s?

Ans: **NO**



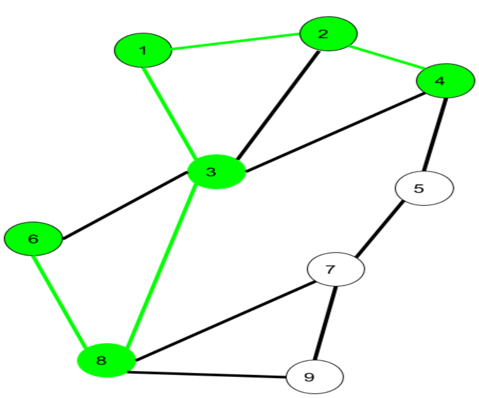
Q. $K_{1,n-1}$ and K_{n-1} ,decompose K_n

Ans: **YES**



3.9 Path

Continues traveling a graph such that all edges are distinct called path.



$\Rightarrow$ $6 \rightarrow 8 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 4$ is a path

$\Rightarrow$ $6 \rightarrow 8 \rightarrow 3 \rightarrow 6$ is a path

$\Rightarrow$ $6 \rightarrow 8 \rightarrow 3 \rightarrow 1 \rightarrow 6$ is **not** a path

$\Rightarrow$ A path in which If vertices and edges both are distinct, they are called a simple path.

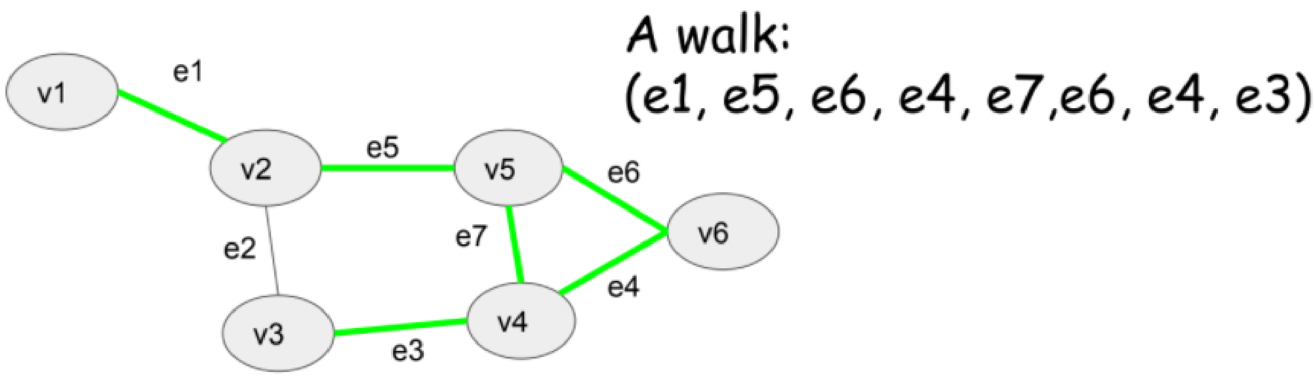
$\Rightarrow$ $6 \rightarrow 8 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 4$ is a simple path

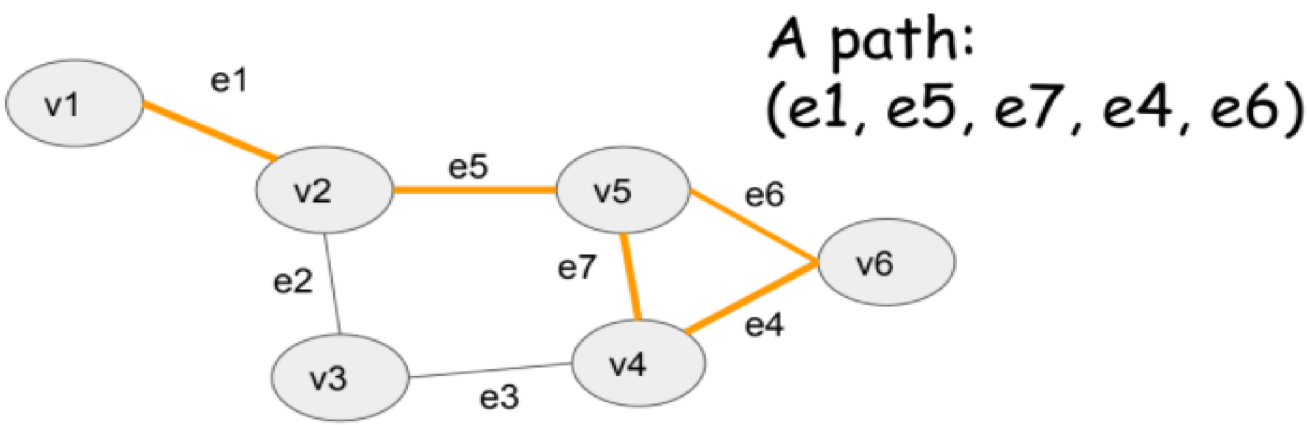
3.10 Walk

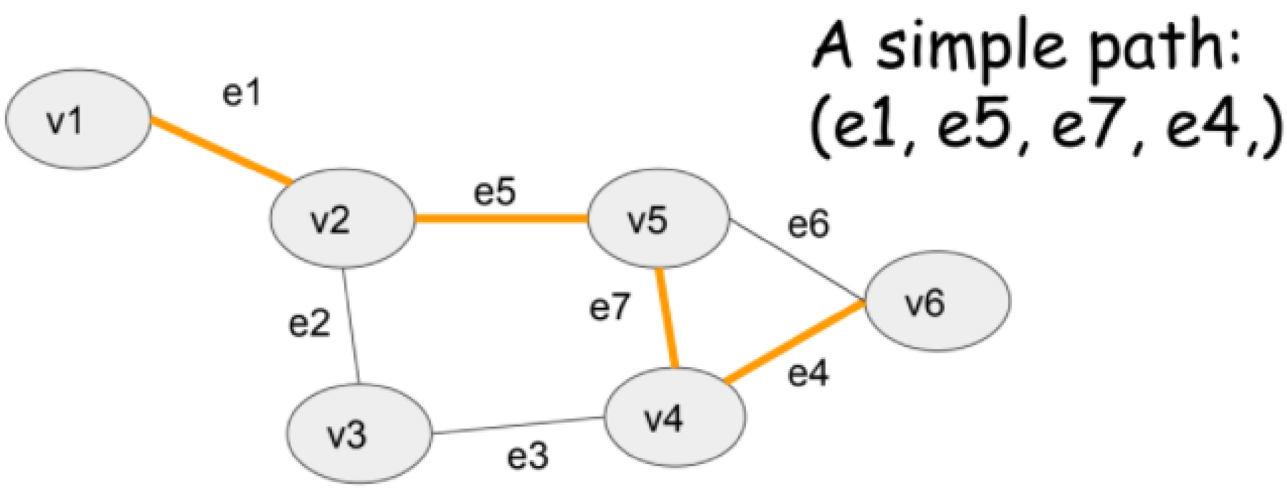
A walk in a graph is a sequence of edges such that each edge (except the first one) starts with a vertex where the previous edges ended.

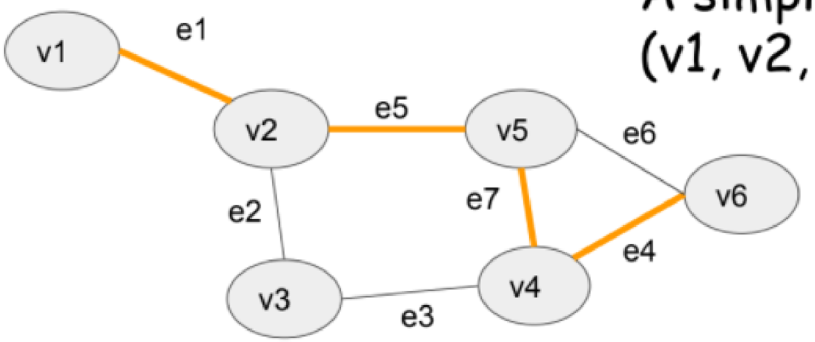
- The length of a walk is the number of edges in it.
- A path is a walk where all edges are distinct.

Examples:

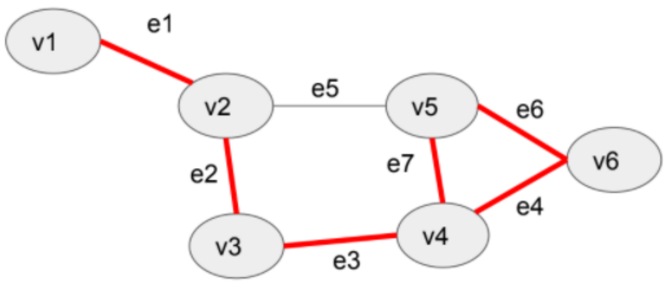








A simple path:
(v1, v2, v5, v4, v6)

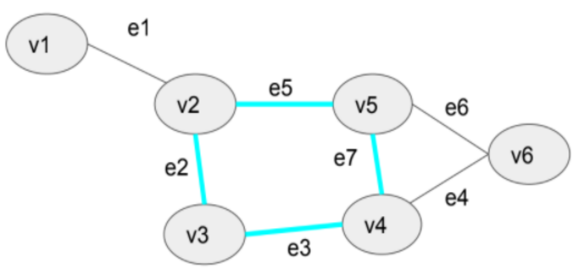


A walk:
(e1, e2, e3, e4, e6,e7)

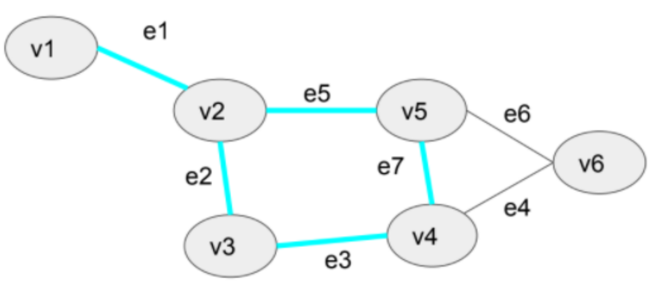
3.11 Cycle

A cycle is a path in which the first and the last vertices are the same.

1. All edges are distinct.
2. Start and End vertex are same.



⇒ V2 → V3 → V4 → V5 → V2 is a cycle

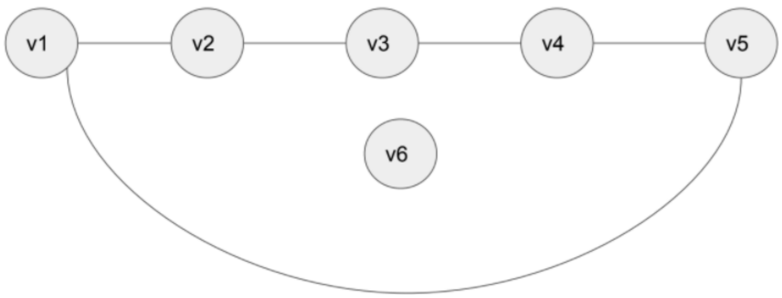


⇒ V1 → V2 → V3 → V4 → V5 → V2 is not any cycle but it contains a cycle

3.12 Connected Components

If there exists at least one path between 2 vertices called connected graphs otherwise called disconnected graphs.

- Maximal connected subgraphs of G is Connected Components
- Every graph with n vertices and k edges has at least n-k components.

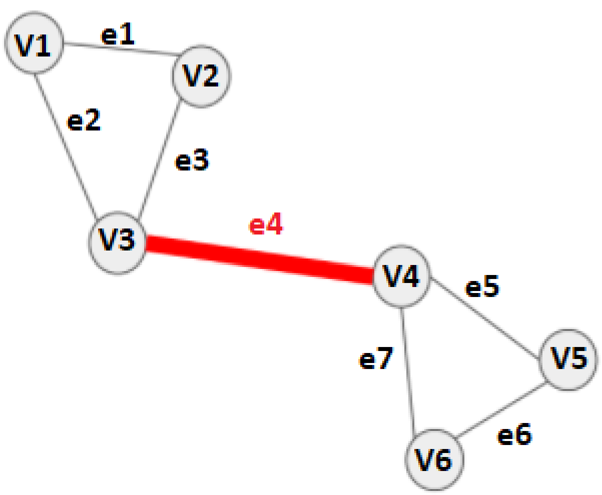


⇒V1, V2, V3, V4, V5 are one connected component and V6 is another connected component.

3.13 Cut-edge or Cut-vertex

A cut-edge or cut-vertex of a graph is an edge or vertex whose deletion increases the number of components.

- An edge is a cut-edge if and only if it belongs to no cycle.



⇒ **e4** is cut edge

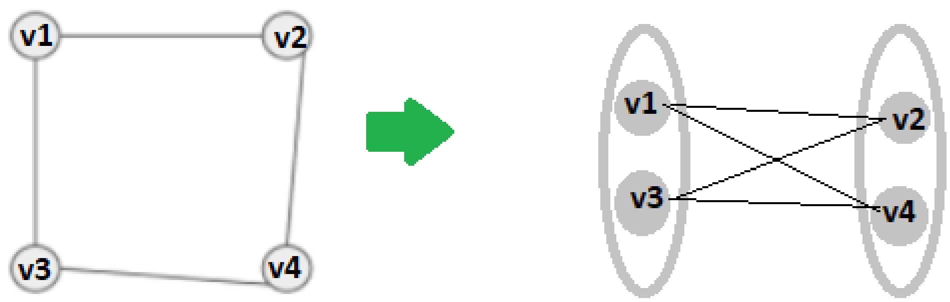
⇒ **V3** & **V4** are cut vertices

3.13 Bipartition of a graph

Bipartite is a graph whose vertices can be divided into two independent sets, U and V such that every edge (u, v) either connects a vertex from U to V or a vertex from V to U means no edge that connects vertices of the same set.

- A graph is bipartite iff it has no odd cycle.

Example:



[1]

1