Graphs & Trees (Lecture 2)

Today's Topics

- Graph Representation
- Graph Isomorphism
- Graph Connectivity / Paths

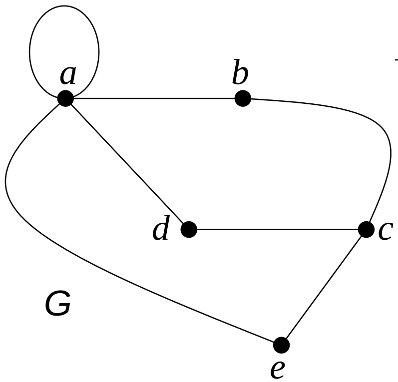
Representing Graphs

We can use:

- Adjacency Lists
- Adjacency Matrices
- Incidence Matrices

Adjacency Lists

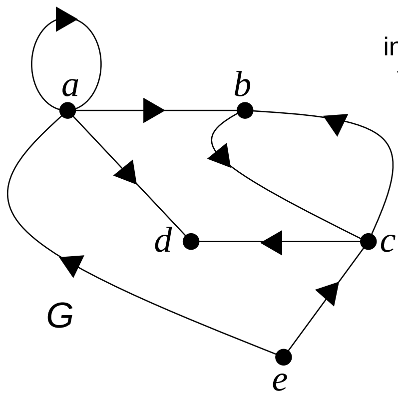
- represent graphs with no multiple edges
- specify vertices that are adjacent to each vertex



vertex	adjacent vertices
a	$a\ b\ d\ e$
b	$a\ c$
c	$b\ d\ e$
d	$a\ c$
e	$a\ c$

Adjacency Lists

- For directed graph, specify terminal vertices of each vertex



initial vertex	terminal vertices
a	$a\ b\ d$
b	c
c	$b\ d$
d	-
e	$a\ c$

Adjacency Matrices

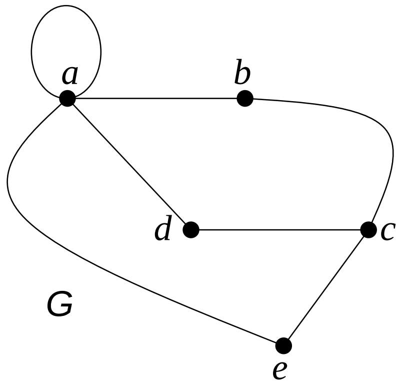
$G = (V,E)$, $V = \{v_1, v_2, \ldots, v_n\}$

$$A = [a_{ij}]$$

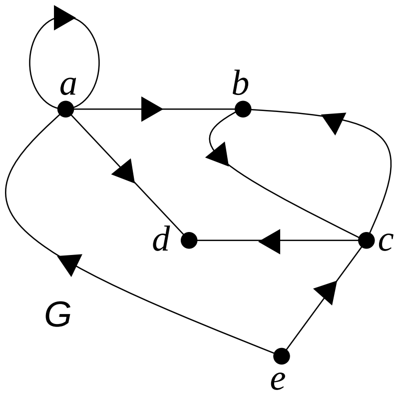
Directed Graph → $a_{ij} =$ Number of edges corresponding to $\{v_i, v_j\}$

Undirected Graph → $a_{ij} =$ Number of edges corresponding to (v_i, v_j)

Adjacency Matrices



Adjacency Matrices



Incidence Matrices

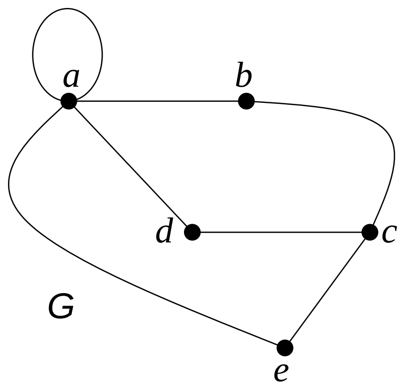
$G=(V,E)$, $V = \{v_1, v_2, \ldots, v_n\}$, $E = \{e_1, e_2, \ldots, e_m\}$

Undirected Graph

$$M = [m_{ij}]$$

$$m_{ij} = \begin{cases} 1 & \text{when } e_j \text{ is incident with } v_i \\ 0 & \text{otherwise.} \end{cases}$$

Incidence Matrices



Isomorphism of Graphs

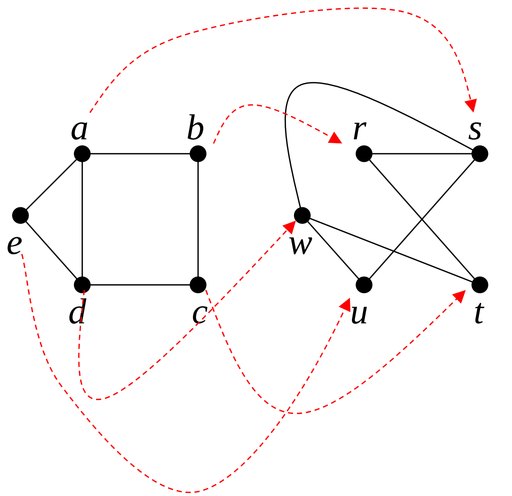
$G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ are two simple graphs. G_1 and G_2 are **isomorphic** when:

There is a 1-to-1 and onto function f from V_1 to V_2 with the property that

$\forall_{a,b \in V_1}$ a and b are adjacent in G1 $\leftrightarrow$ $f(a)$ and $f(b)$ are adjacent in G2.

f is called an **isomorphism**.

Isomorphism of Graphs



Isomorphism, f

$f(a) = s$, $f(b) = r$

$f(c) = t$, $f(d) = w$

$f(e) = u$

$\{a,b\}$	$\{r,s\}$
$\{b,c\}$	$\{r,t\}$
$\{c,d\}$	$\{t,w\}$
$\{d,a\}$	$\{u,w\}$
$\{d,e\}$	$\{s,u\}$
$\{a,e\}$	$\{s,w\}$

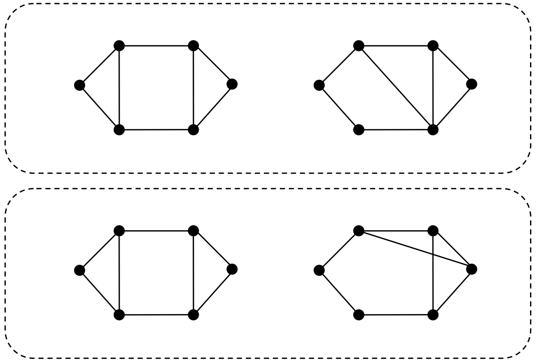
Isomorphism of Graphs

- To show that two graphs are isomorphic
  - Find an isomorphism.
- To show that two graphs are NOT isomorphic
  - Argue that no isomorphism exists.

Graph Invariants

- A property preserved by isomorphism of graphs is called a **graph invariant**.

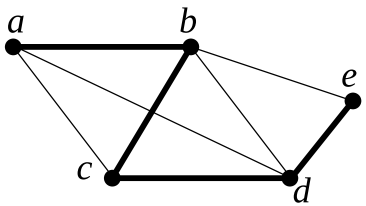
E.g.:
- Number of vertices
- Number of edges
- Degrees of vertices
- Adjacency between vertices with specified degrees



Paths

- A **path** is a sequence of edges that begins with a vertex of a graph and travels from vertex to vertex along edges of the graph.
- The **length** of a path is the number of edges in that path.
- A path is a **circuit** if it begins and ends at the same vertex and the length is not 0.
- A path is **simple** if it does not contain the same edge more than once.

Paths



$(\{a,b\},\{b,c\},\{c,d\},\{d,e\})$ is a simple path of length 4.

When there are no multiple paths, we can represent a path by the sequence of vertices it pass through.

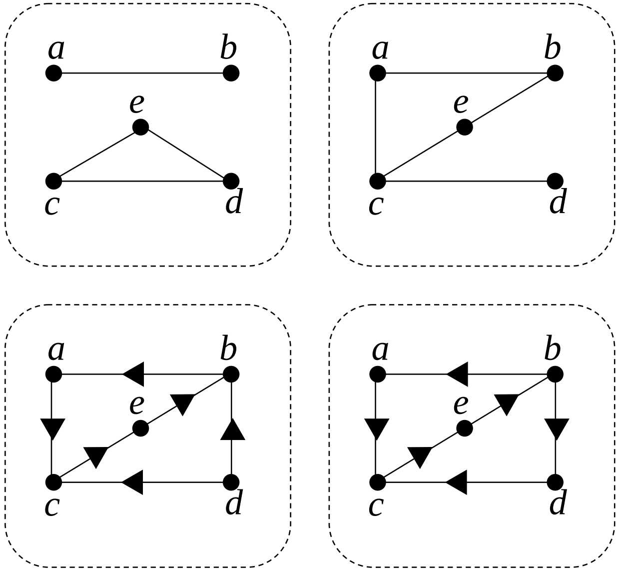
$(\{a,b\},\{b,c\},\{c,d\},\{d,e\})$ ➡ (a,b,c,d,e)

Connectedness

An undirected graph is called **connected** ↔ There is a path between every pair of distinct vertices of the graph.

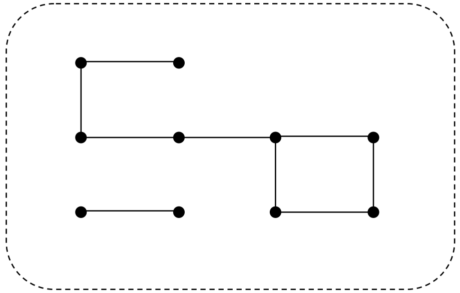
An directed graph is **strongly connected** ↔ There is a path from a to b and from b to a whenever a and b are distinct vertices in the graph.

An directed graph is **weakly connected** ↔ There is a path between every pair of distinct vertices of the graph. (Disregard the edge directions.)



Connected Components

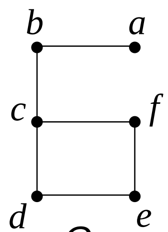
- A **connected component** of a graph is maximal connected subgraph of that graph.



Cut Vertex / Cut Edge

- A vertex is called a **cut vertex** (or **articulation point**) if the removal of that vertex along with all edges incident with it produces a subgraph with more connected components.
- An edge whose removal produces a subgraph with more connected components is called a **cut edge** (or **bridge**).

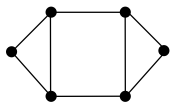
Find all cut vertices and cut edges in *G*.

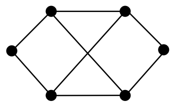


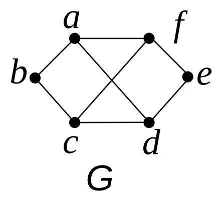
G

Paths and Isomorphism

- The existence of a simple circuit of a particular length is a graph invariant.
- Paths are sometimes useful in finding an isomorphism.



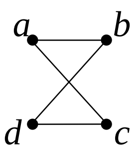




Show that a simple circuit of length 3 does not exist in *G*.

Counting Paths Between Vertices

How many paths of length 4 are there from *a* to *d* in the following graph?



Counting Paths Between Vertices

The number of paths of a certain length between two vertices in a graph can also be determined from the graph's adjacency matrix.