Connectivity

Section 10.4

Paths

Informal Definition: A *path* is a sequence of edges that begins at a vertex of a graph and travels from vertex to vertex along edges of the graph. As the path travels along its edges, it visits the vertices along this path, that is, the endpoints of these edges.

Applications: Numerous problems can be modeled with paths formed by traveling along edges of graphs such as:

- determining whether a message can be sent between two computers.
- efficiently planning routes for mail delivery.

Paths

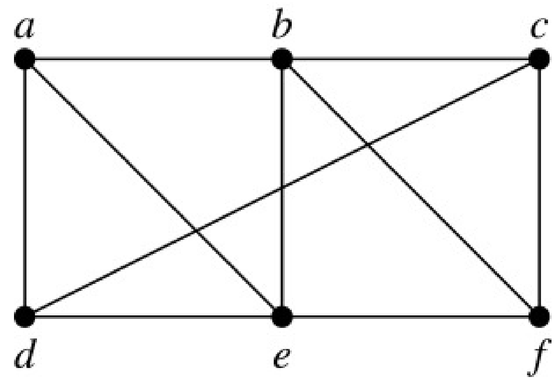
Definition: Let n be a nonnegative integer and G an undirected graph. A *path* of *length* n from u to v in G is a sequence of n edges $e_1, \ldots, e_n$ of G for which there exists a sequence $x_0 = u, x_1, \ldots, x_{n-1}, x_n = v$ of vertices such that e_i has, for $i = 1, \ldots, n$, the endpoints x_{i-1} and x_i.

- When the graph is simple, we denote this path by its vertex sequence $x_0, x_1, \ldots, x_n$ (since listing the vertices uniquely determines the path).
- The path is a *circuit* if it begins and ends at the same vertex ($u = v$) and has length greater than zero.
- The path or circuit is said to *pass through* the vertices $x_1, x_2, \ldots, x_{n-1}$ and *traverse* the edges $e_1, \ldots, e_n$.
- A path or circuit is *simple* if it does not contain the same edge more than once.

Paths (*continued*)

Example: In the simple graph here:

- *a*, *d*, *c*, *f*, *e* is a simple path of length 4.
- *d*, *e*, *c*, *a* is not a path because *e* is not connected to *c*.
- *b*, *c*, *f*, *e*, *b* is a circuit of length 4.
- *a*, *b*, *e*, *d*, *a*, *b* is a path of length 5, but it is not a simple path.

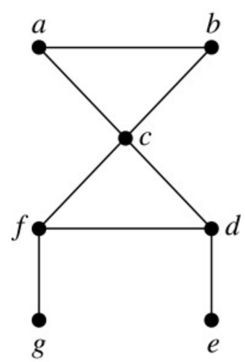


Connectedness in Undirected Graphs

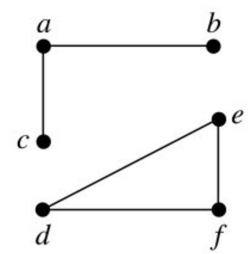
Definition:

- An undirected graph is called *connected* if there is a path between every pair of vertices.
- An undirected graph that is not *connected* is called *disconnected*.
- We say that we *disconnect* a graph when we remove vertices or edges, or both, to produce a disconnected subgraph.

Example: G_1 is connected. G_2 is not connected.



G_1

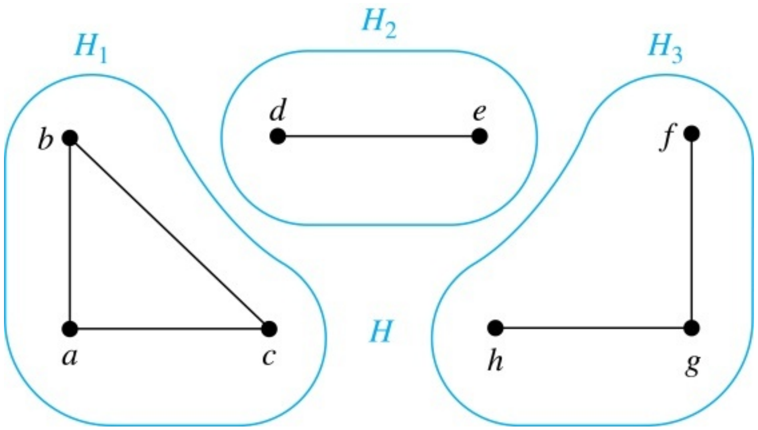


G_2

Connected Components

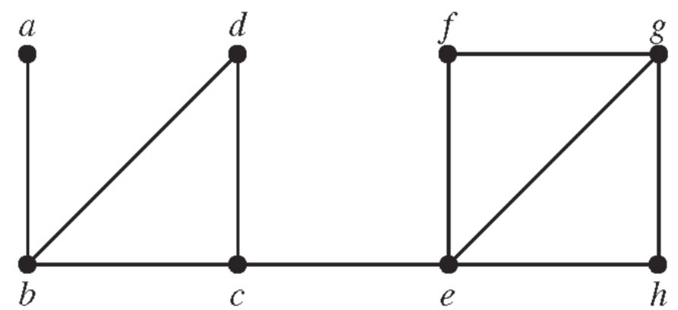
Definition: A *connected component* of a graph *G* is a connected subgraph of *G* that is not a proper subgraph of another connected subgraph of *G*. A graph *G* that is not connected has two or more connected components that are disjoint and have *G* as their union.

Example:



Cut Vertices and Cut Edges

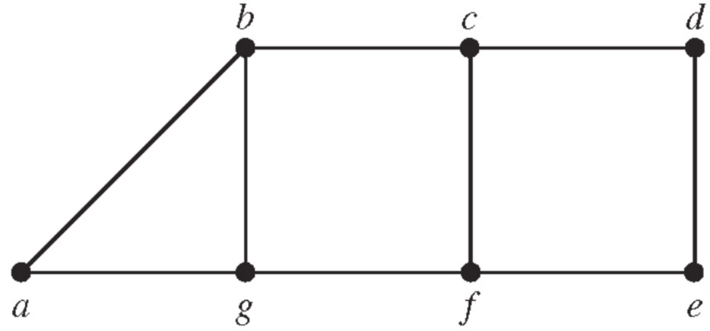
- Removing a *cut vertex* from a connected graph produces a subgraph that is disconnected.
- Removing a *cut edge* has the same effect.
- EXAMPLE: Which are the cut vertices and cut edges in the following graph:
  - Vertices: *b, c, e*
  - Edges: {*a*, *b*}, {*c*, *e*}



G_1

Vertex Cut

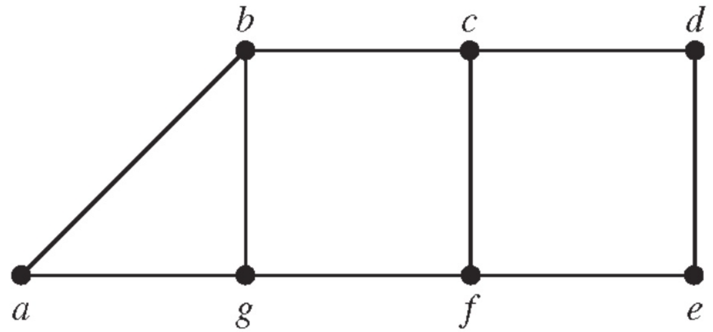
- A *vertex cut*, or *separating set* is a subset *V*' of *V* if *G* – *V*' is disconnected.
- The *vertex connectivity* of G $\kappa(G)$ is the minimum number of vertices in a vertex cut.
- EXAMPLE: What is $\kappa(G_3)$?



G_3

Edge Cut

- A set of edges *E*' is an *edge cut* of *G* if *G* – *E*' is disconnected.
- The *edge connectivity* of *G* $\lambda(G)$ is the minimum number of edges in an edge cut.
- EXAMPLE: What is $\lambda(G_3)$?



G_3

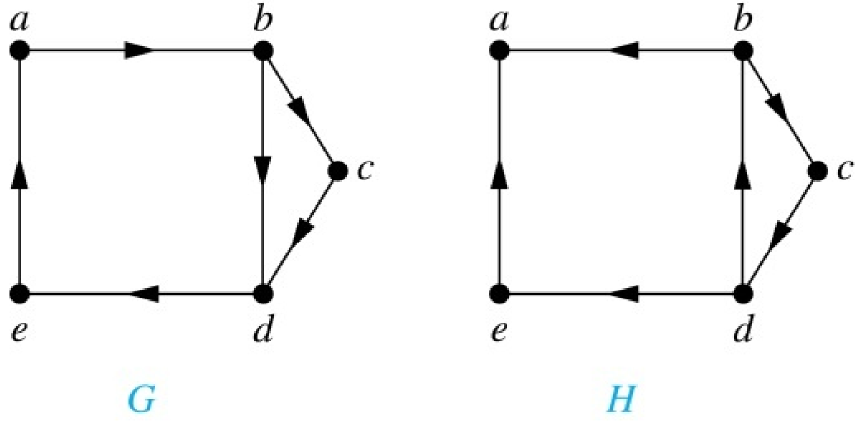
Connectedness in Directed Graphs

Definition: A directed graph is *strongly connected* if there is a path from *a* to *b* and a path from *b* to *a* whenever *a* and *b* are vertices in the graph.

Definition: A directed graph is *weakly connected* if there is a path between every two vertices in the underlying undirected graph, which is the undirected graph obtained by ignoring the directions of the edges of the directed graph.

Connectedness in Directed Graphs (*continued*)

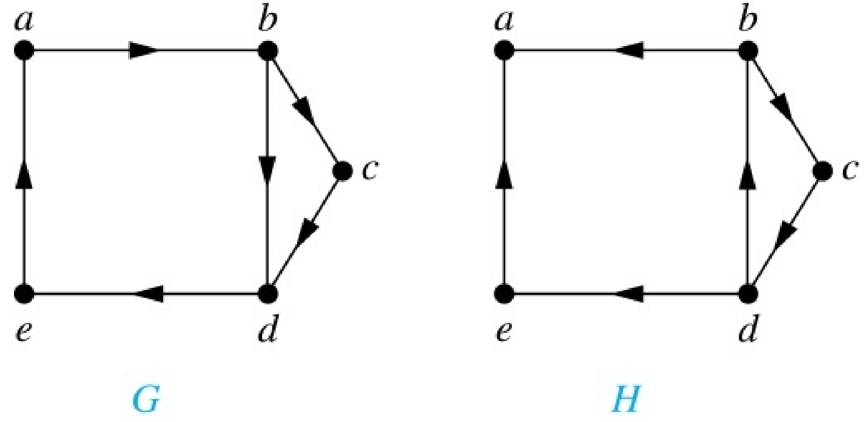
Example:



Connectedness in Directed Graphs (*continued*)

Example:

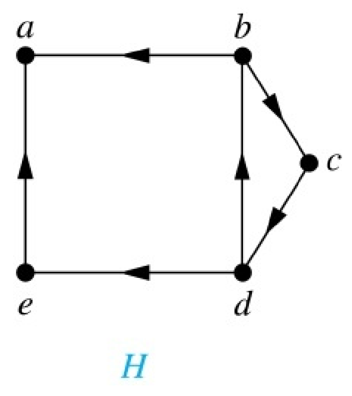
- *G* is strongly connected because there is a path between any two vertices in the directed graph. Hence, *G* is also weakly connected.
- The graph *H* is not strongly connected, since there is no directed path from *a* to *b*, but it is weakly connected.



Connectedness in Directed Graphs (*continued*)

Definition: The subgraphs of a directed graph *G* that are strongly connected but not contained in larger strongly connected subgraphs, that is, the maximal strongly connected subgraphs, are called the *strongly connected components* or *strong components* of *G*.

Example: Strongly connected components of H?

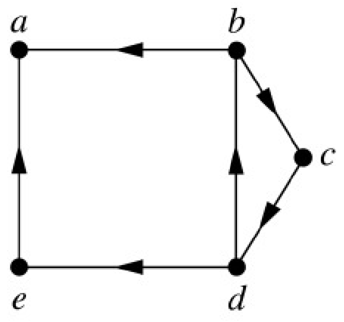


Connectedness in Directed Graphs (*continued*)

Definition: The subgraphs of a directed graph *G* that are strongly connected but not contained in larger strongly connected subgraphs, that is, the maximal strongly connected subgraphs, are called the *strongly connected components* or *strong components* of *G*.

Example: The graph *H* has three strongly connected components:

- the vertex *a*
- the vertex *e*
- the subgraph consisting of the vertices *b*, *c*, *d* and edges (*b*,*c*), (*c*,*d*), and (*d*,*b*).



H

Counting Paths between Vertices

- We can use the adjacency matrix of a graph to find the number of paths between two vertices in the graph.

Theorem: Let G be a graph with adjacency matrix **A** with respect to the ordering $v_1, \ldots, v_n$ of vertices (with directed or undirected edges, multiple edges and loops allowed). The number of different paths of length r from v_i to v_j, where $r > 0$ is a positive integer, equals the (i,j)th entry of $\mathbf{A}^r$.

Proof by mathematical induction:

Basis Step: By definition of the adjacency matrix, the number of paths from v_i to v_j of length 1 is the (i,j)th entry of **A**.

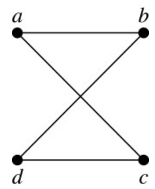
Inductive Step: For the inductive hypothesis, we assume that that the (i,j)th entry of $\mathbf{A}^r$ is the number of different paths of length r from v_i to v_j.

- Because $\mathbf{A}^{r+1} = \mathbf{A}^r\,\mathbf{A}$, the (i,j)th entry of $\mathbf{A}^{r+1}$ equals $b_{i1}a_{1j} + b_{i2}a_{2j} + \cdots + b_{in}a_{nj}$, where b_{ik} is the $(i,k)th$ entry of $\mathbf{A}^r$. By the inductive hypothesis, b_{ik} is the number of paths of length r from v_i to v_k.
- A path of length $r + 1$ from v_i to v_j is made up of a path of length r from v_i to some v_k, and an edge from v_k to v_j. By the product rule for counting, the number of such paths is the product of the number of paths of length r from v_i to v_k (i.e., b_{ik}) and the number of edges from from v_k to v_j (i.e, a_{kj}). The sum over all possible intermediate vertices v_k *is* $b_{i1}a_{1j} + b_{i2}a_{2j} + \cdots + b_{in}a_{nj}$.

Counting Paths between Vertices (*continued*)

Example: How many paths of length four are there from *a* to *d* in the graph G.

G



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

adjacency matrix of G

Solution: The adjacency matrix of *G* (ordering the vertices as *a*, *b*, *c*, *d*) is given above. Hence the number of paths of length four from *a* to *d* is the (1, 4)th entry of $\mathbf{A}^4$. The eight paths are:

a, b, a, b, d	*a, b, a, c, d*
a, b, d, b, d	*a, b, d, c, d*
a, c, a, b, d	*a, c, a, c, d*
a, c, d, b, d	*a, c, d, c, d*

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