Discrete Mathematics for Computer Science

Lecture 19: Graph

Dr. Ming Tang

Department of Computer Science and Engineering
Southern University of Science and Technology (SUSTech)
Email: tangm3@sustech.edu.cn

SUSTech Southern University of Science and Technology

The Principle of Well-Ordered Induction

Recall: $(S, \preccurlyeq)$ is a well-ordered set if it is a poset such that $\preccurlyeq$ is a total ordering and every nonempty subset of S has a least element.

The Principle of Well-Ordered Induction: Suppose that $(S, \preccurlyeq)$ is a well-ordered set. Then $P(x)$ is true for all $x \in S$, if

Inductive Step: For every $y \in S$, if $P(x)$ is true for all $x \in S$ with $x \prec y$, then $P(y)$ is true.

SUSTech Southern University of Science and Technology

The Principle of Well-Ordered Induction: Example

Consider lexicographic ordering defined on set $\mathbb{N} \times \mathbb{N}$: $(x_1, y_1) \preccurlyeq (x_2, y_2)$ if either $x_1 < x_2$, or $x_1 = x_2$ and $y_1 < y_2$. Recall that it is a well-ordered set.

Example: Suppose that $a_{m,n}$ is defined recursively for $(m, n) \in \mathbb{N} \times \mathbb{N}$ by $a_{0,0} = 0$ and

$$a_{m,n} = \begin{cases} a_{m-1,n} + 1, & \text{if } n = 0 \text{ and } m > 0, \\ a_{m,n-1} + n, & \text{if } n > 0. \end{cases}$$

Show that $a_{m,n} = m + n(n+1)/2$ for all $(m, n) \in \mathbb{N} \times \mathbb{N}$.

SUSTech Southern University of Science and Technology

The Principle of Well-Ordered Induction: Example

Example: Suppose that $a_{m,n}$ is defined recursively for $(m, n) \in \mathbb{N} \times \mathbb{N}$ by $a_{0,0} = 0$ and

$$a_{m,n} = \begin{cases} a_{m-1,n} + 1, & \text{if } n = 0 \text{ and } m > 0, \\ a_{m,n-1} + n, & \text{if } n > 0. \end{cases}$$

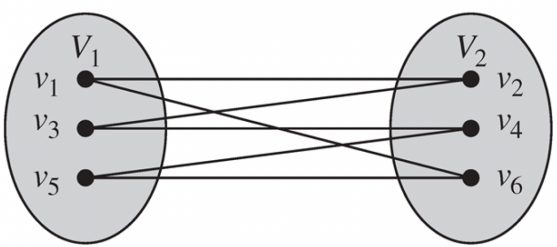
Show that $a_{m,n} = m + n(n+1)/2$ for all $(m, n) \in \mathbb{N} \times \mathbb{N}$.

- Basic Step: $a_{0,0} = 0 + 0 \cdot (0+1)/2 = 0$
- Inductive Step: Suppose that $a_{m',n'} = m' + n'(n'+1)/2$ whenever $(m', n') \prec (m, n)$. We aim to prove that $a_{m,n} = m + n(n+1)/2$.
  - $n = 0$, under which $a_{m,n} = a_{m-1,n} + 1$: Since $(m-1, n) \prec (m, n)$, we have $a_{m-1,n} = m - 1 + n(n+1)/2$. Thus, $a_{m,n} = m + n(n+1)/2$.
  - $n > 0$, under which $a_{m,n} = a_{m,n-1} + n$: Since $(m, n-1) \prec (m, n)$, we have $a_{m,n-1} = m + (n-1)(n-1+1)/2$. Thus, $a_{m,n} = m + n(n+1)/2$.

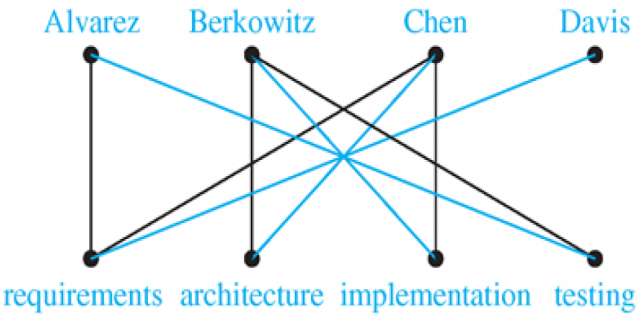
SUSTech Southern University of Science and Technology

Bipartite Graphs

Definition: A simple graph G is bipartite if V can be partitioned into two disjoint subsets V_1 and V_2 such that every edge connects a vertex in V_1 and a vertex in V_2.



Matching the elements of one set to elements in another. A matching is a subset of E s.t. no two edges are incident with the same vertex.



SUSTech Southern University of Science and Technology

Bipartite Graphs and Matchings

A matching M in a bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) is a complete matching from V_1 to V_2 if every vertex in V_1 is the endpoint of an edge in the matching, or equivalently, if $|M| = |V_1|$.

Theorem (Hall's Marriage Theorem): The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.

SUSTech Southern University of Science and Technology

Proof of Hall's Theorem

Theorem (Hall's Marriage Theorem): The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.

Proof: "only if"

Suppose that there is a complete matching M from V_1 to V_2. Consider an arbitrary subset $A \subseteq V_1$.

Then, for every vertex $v \in A$, there is an edge in M connecting v to a vertex in V_2.

Thus, there are at least as many vertices in V_2 that are neighbors of vertices in V_1 as there are vertices in V_1.

Hence, $|N(A)| \geq |A|$.

SUSTech Southern University of Science and Technology

Proof of Hall's Theorem

Theorem (Hall's Marriage Theorem): The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.

Proof: "if", use strong induction to prove it.

Basic Step: $|V_1| = 1$

Inductive hypothesis: Let k be a positive integer. If $G = (V, E)$ is a bipartite graph with bipartition (V_1, V_2), and $|V_1| = j \leq k$, then there is a complete mathching M from V_1 to V_2 whenever the condition that $|N(A)| \geq |A|$ for all $A \subseteq V_1$ is met.

Inductive step: Suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

SUSTech Southern University of Science and Technology

Proof of Hall's Theorem

Inductive step: Suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

(i) For all integers j with $1 \leq j \leq k$, the vertices in every set of j elements from W_1 are adjacent to at least $j + 1$ elements of W_2.

(ii) For some integer j with $1 \leq j \leq k$, there is a subset W_1' of j vertices such that there are exactly j neighbors of these vertices in W_2.

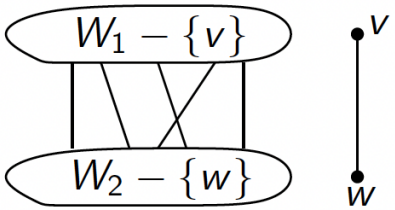
SUSTech Southern University of Science and Technology

Proof of Hall's Theorem

Inductive step: Suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

(i) For all integers j with $1 \leq j \leq k$, the vertices in every set of j elements from W_1 are adjacent to at least $j + 1$ elements of W_2.

We select a vertex $v \in W_1$ and an element $w \in N(\{v\})$. The inductive hypothesis tells us there is a complete matching from $W_1 - \{v\}$ to $W_2 - \{w\}$.



(ii) For some integer j with $1 \leq j \leq k$, there is a subset W_1' of j vertices such that there are exactly j neighbors of these vertices in W_2.

SUSTech Southern University of Science and Technology

Proof of Hall's Theorem

Inductive step: Suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

(i) For all integers j with $1 \leq j \leq k$, the vertices in every set of j elements from W_1 are adjacent to at least $j + 1$ elements of W_2.

(ii) For some integer j with $1 \leq j \leq k$, there is a subset W_1' of j vertices such that there are exactly j neighbors of these vertices in W_2.

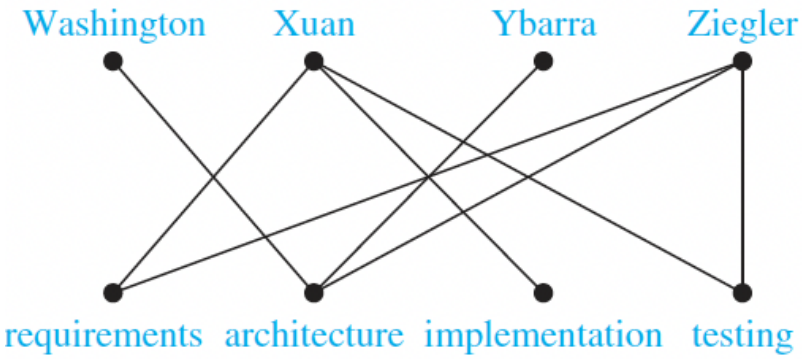
Let W_2' be the set of these neighbors. Then by i.h., there is a complete matching from W_1' to W_2'. Now consider the graph $K = (W_1 - W_1', W_2 - W_2')$.

We will show that the condition $|N(A)| \geq |A|$ is met for all subsets A of $W_1 - W_1'$.

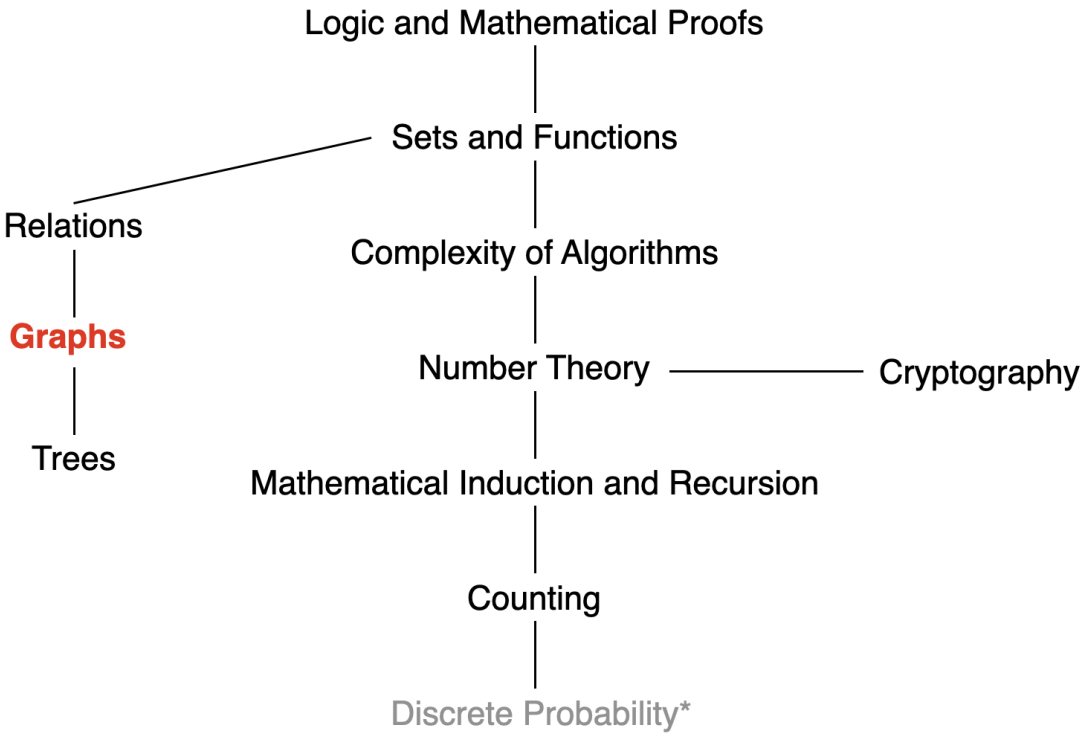
If not, there would be a subset of t vertices of $W_1 - W'$ such that have fewer than t vertices of $W_2 - W_2'$ as neighbors.

Then, the set of $j + t$ vertices of $W1$ (i.e., these t vertices and the j vertices removed) has fewer than $j + t$ neighbors in W_2, contradicting the hypothesis that $|N(A)| \geq |A|$ for all $A \subseteq W_1$.

Hall's Theorem: Example



This Lecture



Graph and terminologies, representing graphs and graph isomorphism, connectivity, Euler and Hamiliton path, ...

SUSTech Southern University of Science and Technology

Representation of Graphs

To represent a graph, we may use adjacency lists, adjacency matrices, and incidence matrices.

Definition: An adjacency list can be used to represent a graph with no multiple edges by specifying the vertices that are adjacent to each vertex of the graph.

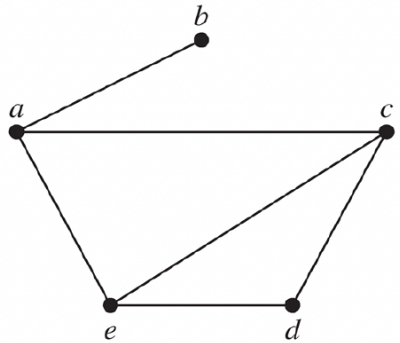


TABLE 1 An Adjacency List for a Simple Graph.

Vertex	*Adjacent Vertices*
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

SUSTech Southern University of Science and Technology

Representation of Graphs

Definition: An adjacency list can be used to represent a graph with no multiple edges by specifying the vertices that are adjacent to each vertex of the graph.

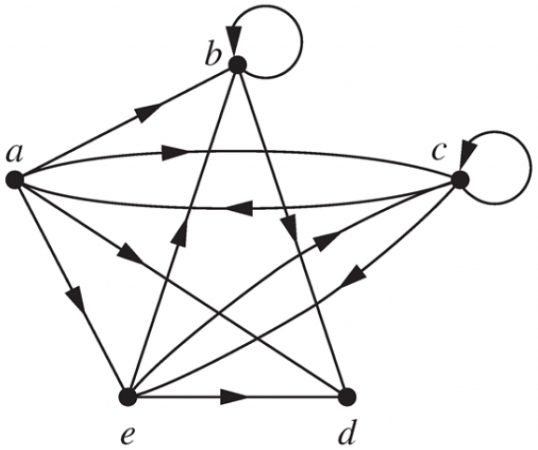


TABLE 2 An Adjacency List for a Directed Graph.

Initial Vertex	*Terminal Vertices*
a	b, c, d, e
b	b, d
c	a, c, e
d	
e	b, c, d

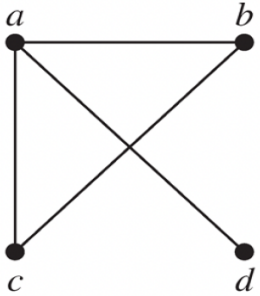
SUSTech Southern University of Science and Technology

Adjacency Matrices

Definition: Suppose that $G = (V, E)$ is a simple graph with $|V| = n$. Arbitrarily list the vertices of G as $v_1, v_2, \ldots, v_n$. The adjacency matrix A_G of G, is the $n \times n$ zero-one matrix with 1 as its (i, j)-th entry when v_i and v_j are adjacent, and 0 as its (i, j)-th entry when they are not adjacent.

$$\mathbf{A}_G = [a_{ij}]_{n \times n}, \text{ where}$$

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

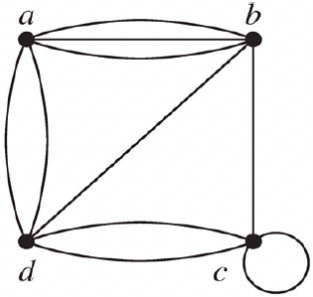


$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

SUSTech Southern University of Science and Technology

Adjacency Matrices

Adjacency matrices can also be used to represent graphs with loops and multiple edges. The matrix is no longer a zero-one matrix.



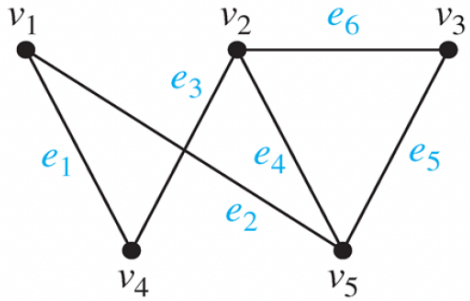
$$\begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

SUSTech Southern University of Science and Technology

Incidence Matrices

Definition: Let $G = (V, E)$ be an undirected graph with vertices v_1, v_2, . . . , v_n and edges e_1, e_2, . . . , e_m. The incidence matrix with respect to the ordering of V and E is the $n \times m$ matrix $\mathrm{M} = [m_{ij}]_{n \times m}$, where

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



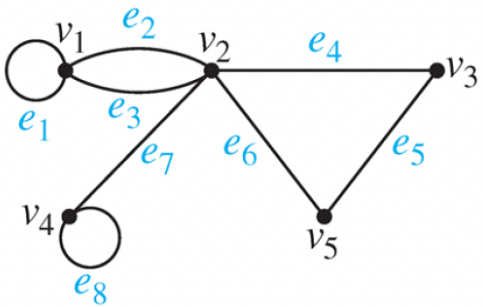
$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

SUSTech Southern University of Science and Technology

Incidence Matrices

Definition: Let $G = (V, E)$ be an undirected graph with vertices v_1, v_2, . . . , v_n and edges e_1, e_2, . . . , e_m. The incidence matrix with respect to the ordering of V and E is the $n \times m$ matrix $\mathrm{M} = [m_{ij}]_{n \times m}$, where

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



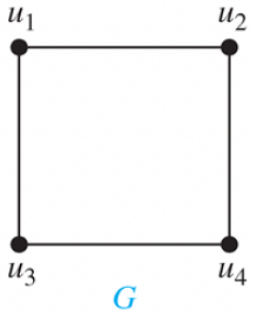
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

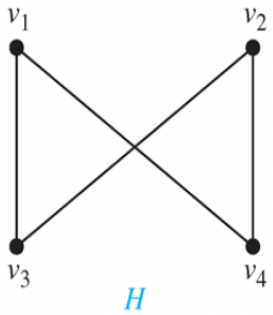
SUSTech Southern University of Science and Technology

Isomorphism of Graphs

Definition: The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there is a one-to-one and onto function from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if $f(a)$ and $f(b)$ are adjacent in G_2, for all a and b in V_1. Such a function is called an isomorphism.

Are the two graphs isomorphic?





Define a one-to-one correspondence: $f(u_1) = v_1$, $f(u_2) = v_4$, $f(u_3) = v_3$, and $f(u_4) = v_2$.

SUSTech Southern University of Science and Technology

Isomorphism of Graphs

It is usually difficult to determine whether two simple graphs are isomorphic using brute force since there are $n!$ possible one-to-one correspondences.

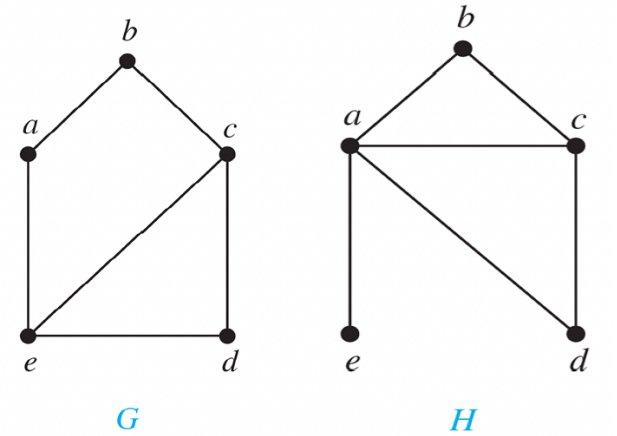
Sometimes it is not difficult to show that two graphs are not isomorphic. We can achieve this by checking some graph invariants.

Useful graph invariants include the number of vertices, number of edges, degree sequence, etc.

SUSTech Southern University of Science and Technology

Isomorphism of Graphs: Example

Determine whether these two graphs are isomorphic.

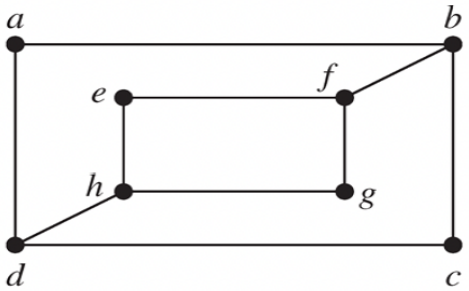


H has a vertex of degree one, namely, e, whereas G has no vertices of degree one. It follows that G and H are not isomorphic.

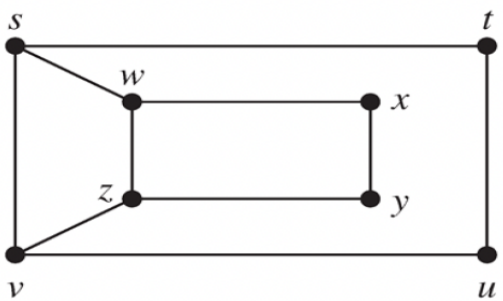
SUSTech Southern University of Science and Technology

Isomorphism of Graphs: Example

Determine whether these two graphs are isomorphic.



G



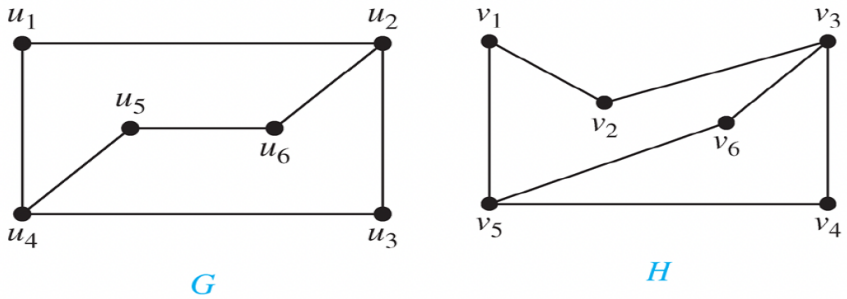
H

G and H are not isomorphic. This is because $deg(a) = 2$ in G, and a must correspond to either t, u, x, or y in H. However, each of these four vertices in H is adjacent to another vertex of degree two in H, which is not true for a in G.

SUSTech Southern University of Science and Technology

Isomorphism of Graphs: Example

Determine whether these two graphs are isomorphic.

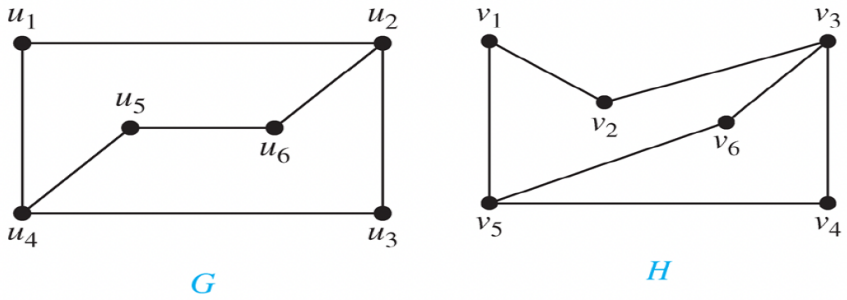


Because many isomorphic invariants (e.g., number of vertices/edges, degree) agree, G and H may be isomorphic. We now will define a function f:

- $f(u_1)$ can be either v_4 or v_6, because u_1 is not adjacent to any other vertex of degree two. We arbitrarily set $f(u_1) = v_6$.
- u_2 is adjacent to u_1, so $f(u_2)$ can be either v_3 or v_5. We arbitrarily set $f(u_2) = v_3$.
- ...
- $f(u_3) = v_4$, $f(u_4) = v_5$, $f(u_5) = v_1$, and $f(u_6) = v_2$.

Isomorphism of Graphs: Example

Determine whether these two graphs are isomorphic.



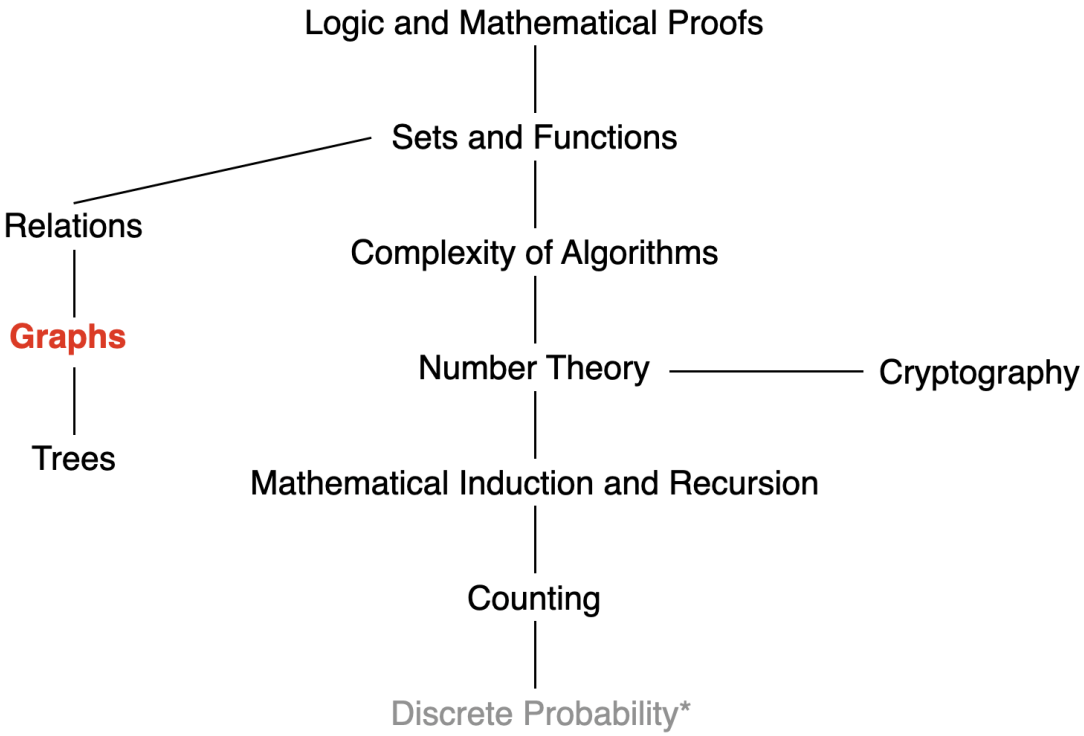
$f(u_1) = v_6,\ f(u_2) = v_3,\ f(u_3) = v_4,\ f(u_4) = v_5,\ f(u_5) = v_1,\ f(u_6) = v_2.$

$$\mathbf{A}_H = \begin{array}{c} \\ v_6 \\ v_3 \\ v_4 \\ v_5 \\ v_1 \\ v_2 \end{array} \begin{array}{c} \begin{array}{cccccc} v_6 & v_3 & v_4 & v_5 & v_1 & v_2 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}. \qquad \mathbf{A}_G = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{array} \begin{array}{c} \begin{array}{cccccc} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array},$$

SUSTech University of Science and Technology

We conclude that f is an isomorphism, so G and H are isomorphic.

This Lecture



Graph and terminologies, representing graphs and graph isomorphism, connectivity, Euler and Hamiliton path, ...

SUSTech Southern University of Science and Technology

Path: Undirected Graph

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 $e_1, e_2, \ldots, e_n$ 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 the endpoints x_{i-1} and x_i for $i = 1, ..., n$.

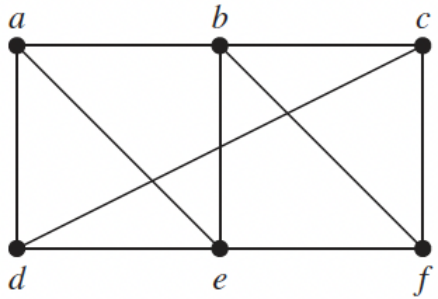
The path is a circuit if it begins and ends at the same vertex, i.e., if $u = v$, and has length greater than zero.

A path or circuit is simple if it does not contain the same edge more than once.

Length of a path = the number of edges on path

SUSTech Southern University of Science and Technology

Path: Undirected Graph



- a, d, c, f, e is a simple path of length 4.
- d, e, c, a is not a path, because $\{e, c\}$ is not an edge.
- b, c, f, e, b is a circuit of length 4.
- The path a, b, e, d, a, b, which is of length 5, is not simple because it contains the edge $\{a, b\}$ twice.

SUSTech Southern University of Science and Technology

Path: Directed Graph

Definition: Let n be a nonnegative integer and G an directed graph. A path of length n from u to v in G is a sequence of n edges $e_1, e_2, \ldots, e_n$ 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 is associated with initial vertex x_{i-1} and terminal vertex x_i for $i = 1, ..., n$.

A path of length greater than zero that begins and ends at the same vertex is called a circuit or cycle.

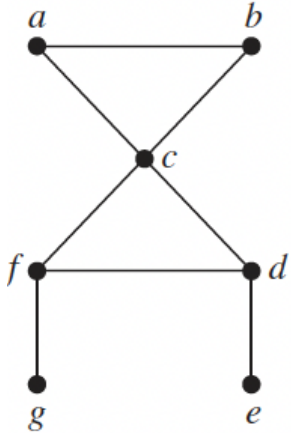
A path or circuit is called simple if it does not contain the same edge more than once.

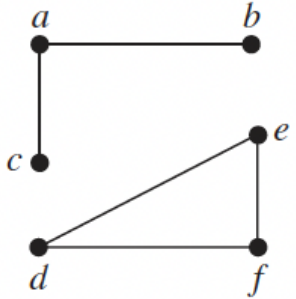
SUSTech Southern University of Science and Technology

Connectivity

An undirected graph is called connected if there is a path between every pair of distinct vertices of the graph.

An undirected graph that is not connected is called disconnected.

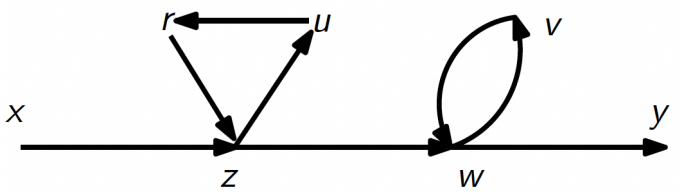




SUSTech Southern University of Science and Technology

Connectivity

Lemma: If there is a path between two distinct vertices x and y of a graph G, then there is a simple path between x and y in G.
Proof: Just delete cycles (loops).



Path from x to y: $x, z, u, r, z, w, v, w, y$.
Path from x to y: x, z, w, y.

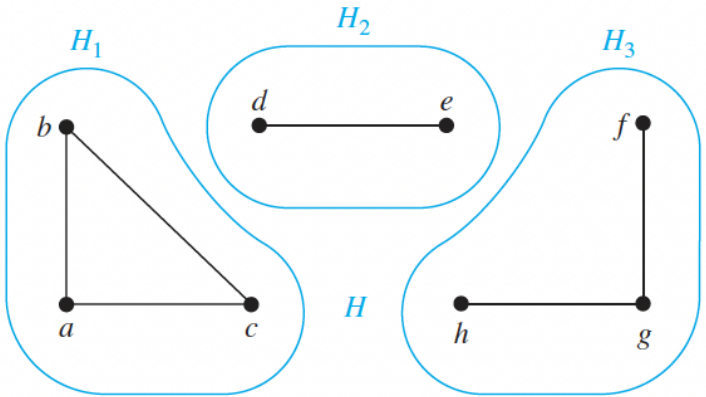
Theorem: There is a simple path between every pair of distinct vertices of a connected undirected graph.

SUSTech Southern University of Science and Technology

Connectivity

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.

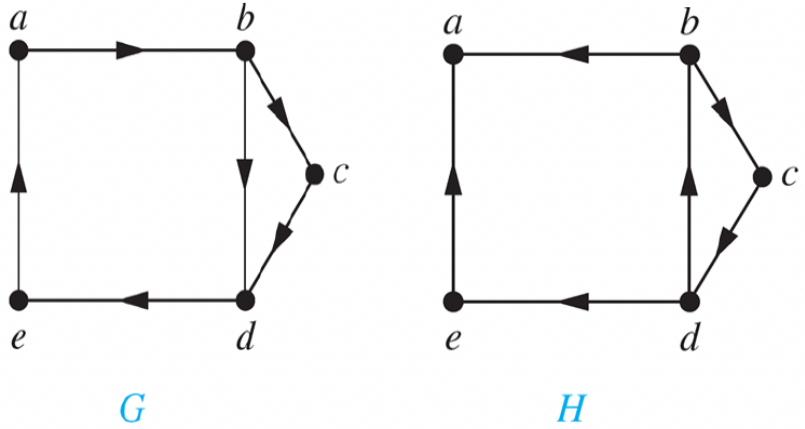


SUSTech Southern University of Science and Technology

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.



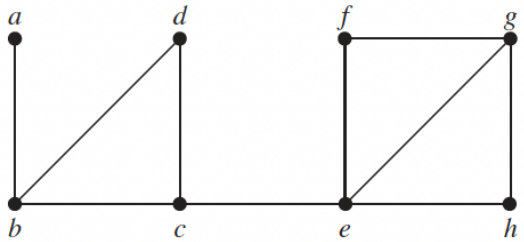
G is strongly connected; H is weakly connected.

SUSTech Southern University of Science and Technology

Cut Vertices and Cut Edges

Sometimes the removal from a graph of a vertex and all incident edges disconnect the graph.

Such vertices are called cut vertices. Similarly we may define cut edges.

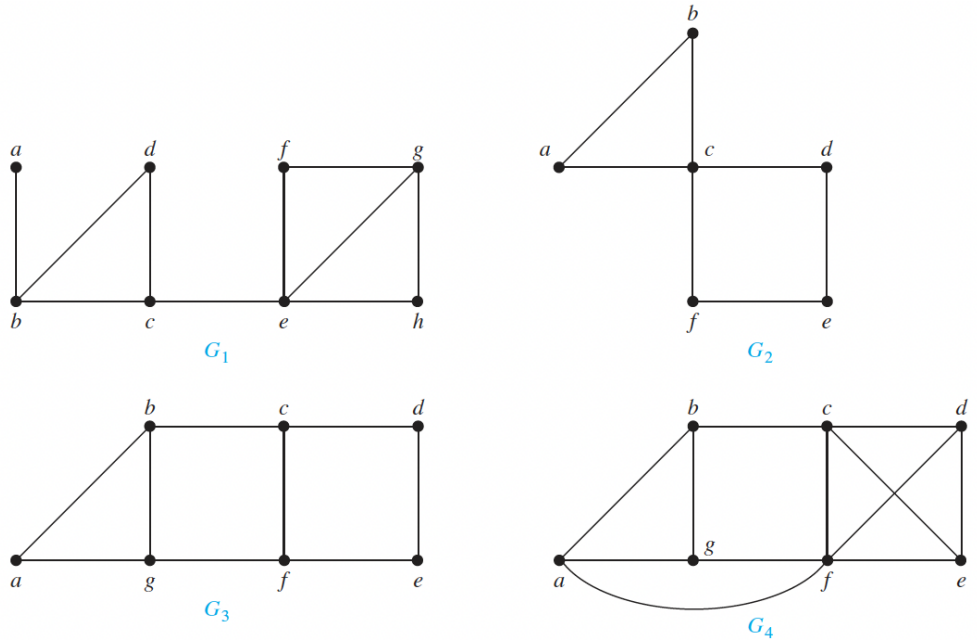


The cut vertices are b, c, and e.

The cut edges are $\{a, b\}$ and $\{c, e\}$.

Cut Vertices and Cut Edges

A set of edges E' is called an edge cut of G if the subgraph $G - E'$ is disconnected. The edge connectivity $\lambda(G)$ is the minimum number of edges in an edge cut of G.

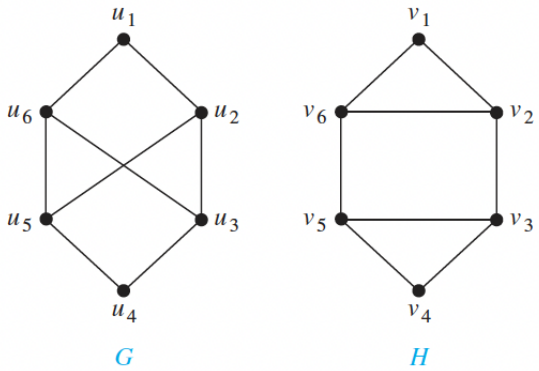


$\lambda(G_1) = 1;\ \lambda(G_2) = 2;\ \lambda(G_3) = 2;\ \lambda(G_4) = 3$

SUSTech Southern University of Science and Technology

Paths and Isomorphism

The existence of a simple circuit of length k is isomorphic invariant. This can be used to construct mappings that may be isomorphisms.

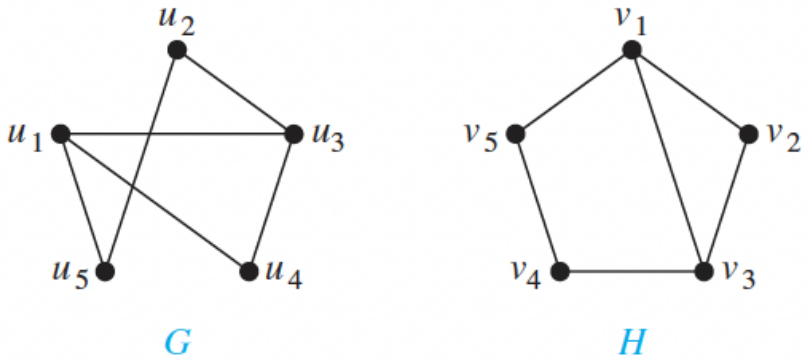


Not isomorphic. H has a simple circuit of length three, namely, v_1, v_2, v_6, v_1, whereas G has no simple circuit of length three.

SUSTech Southern University of Science and Technology

Paths and Isomorphism

The existence of a simple circuit of length k is isomorphic invariant. This can be used to construct mappings that may be isomorphisms.



Because many isomorphic invariants (e.g., number of vertices/edges, degree, circuit) agree, G and H may be isomorphic. Let $f(u_1) = v_3$, $f(u_4) = v_2$, $f(u_3) = v_1$, $f(u_2) = v_5$, and $f(u_5) = v_4$. We can show that f is an isomorphism, so G and H are isomorphic.

SUSTech Southern University of Science and Technology

Counting Paths between Vertices

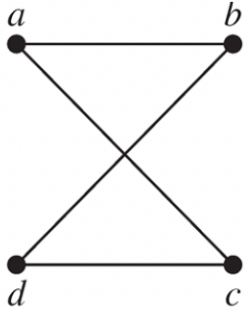
Theorem: Let G be a graph with adjacency matrix A with respect to the ordering $v_1, v_2, \ldots, v_n$ of vertices. The number of different paths of length r from v_i to v_j, where r is a positive integer, equals the (i, j)-th entry of A^r.

Note: with directed or undirected edges, multiple edges and loops allowed

SUSTech Southern University of Science and Technology

Counting Paths between Vertices:

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



$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \qquad A^4 = \begin{bmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{bmatrix}$$

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.

Counting Paths between Vertices

Theorem: Let G be a graph with adjacency matrix A with respect to the ordering $v_1, v_2, \ldots, v_n$ of vertices. The number of different paths of length r from v_i to v_j, where r is a positive integer, equals the (i,j)-th entry of A^r.

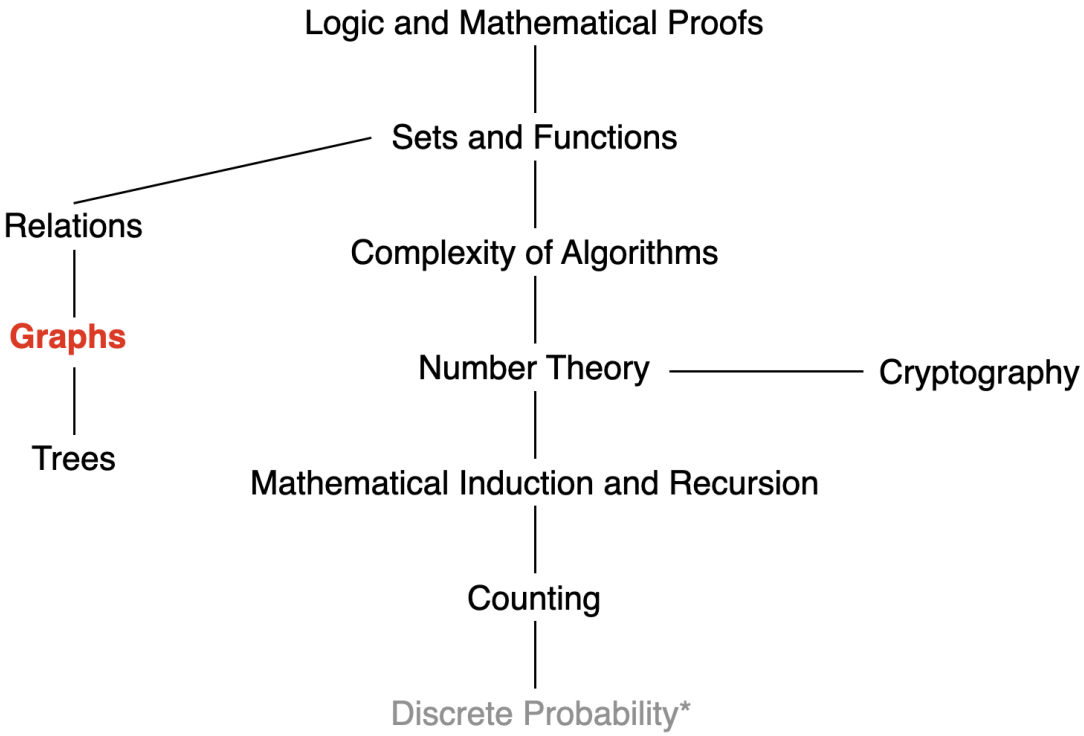
Proof (by induction):

- Basic Step: The number of paths from v_i to v_j of length 1 is the (i,j)-th entry of A.
- Inductive hypothesis: Assume that the (i,j)-th entry of A^r is the number of different paths of length r from v_i to v_j.
- Inductive Step: $A^{r+1} = A^r A$. The (i,j)-th entry of 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 A^r. By the inductive hypothesis, b_{ik} is the number of paths of length r from v_i to v_k.
- Inductive Conclusion: (i,j)-th entry of A^{r+1} counts all paths with length $r+1$ for all possible intermediate vertices v_k.

This Lecture

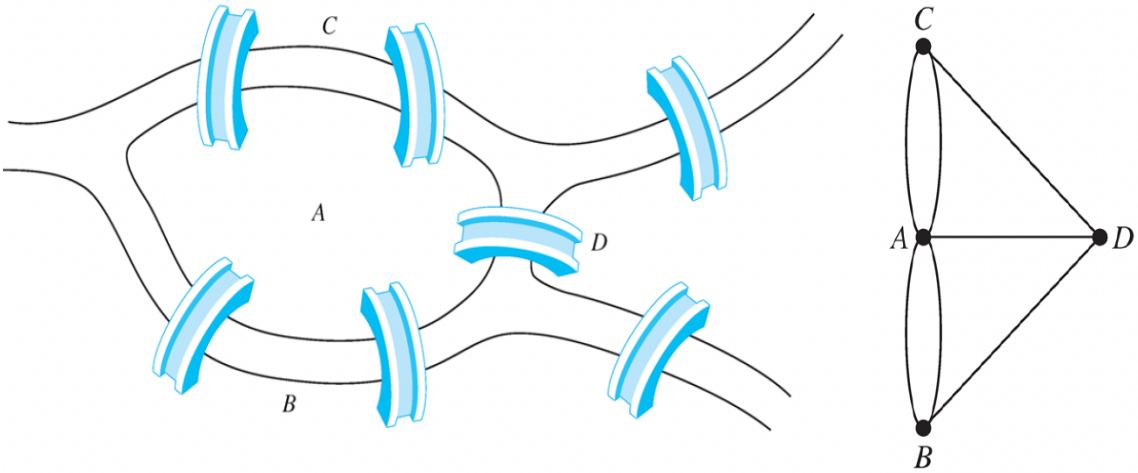


Graph and terminologies, representing graphs and graph isomorphism, connectivity, Euler and Hamiliton path, ...

SUSTech Southern University of Science and Technology

Euler Paths

Königsberg seven-bridge problem: People wondered whether it was possible to start at some location in the town, travel across all the bridges once without crossing any bridge twice, and return to the starting point.

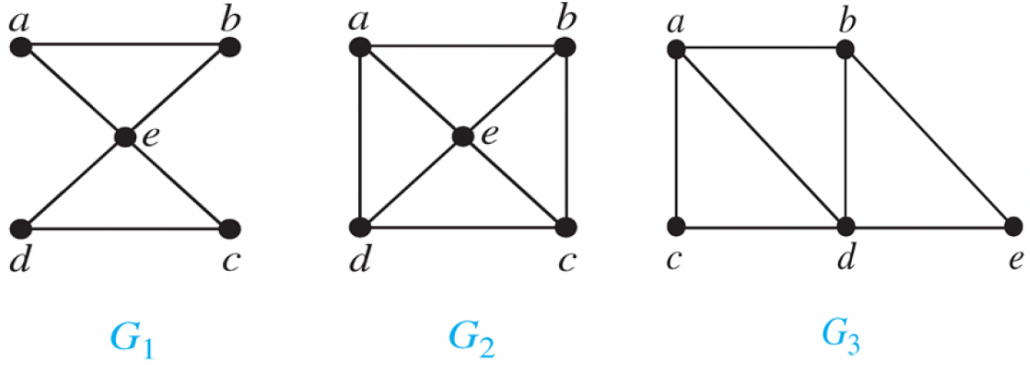


Euler Paths and Circuits

Definition: An Euler circuit in a graph G is a simple circuit containing every edge of G. An Euler path in G is a simple path containing every edge of G.

Recall that a path or circuit is simple if it does not contain the same edge more than once.

Example: Which of the undirected graphs have an Euler circuit? Of those that do not, which have an Euler path?

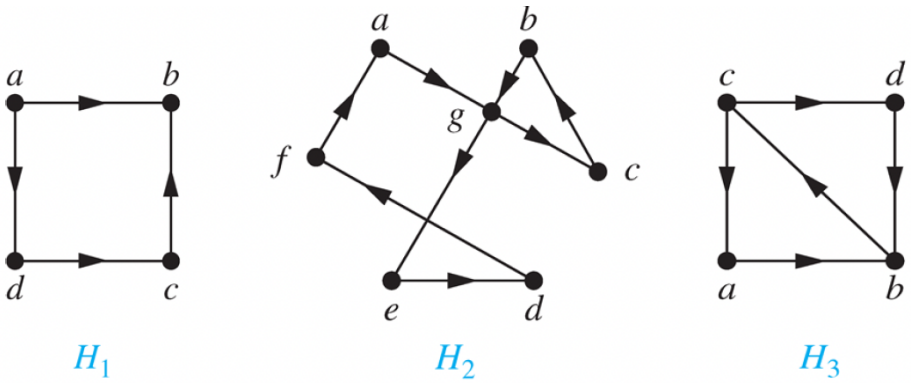


G_1: an Euler circuit, e.g., a, e, c, d, e, b, a;
G_2: neither; G_3: an Euler path, e.g., a, c, d, e, b, d, a, b

Euler Paths and Circuits

Definition: An Euler circuit in a graph G is a simple circuit containing every edge of G. An Euler path in G is a simple path containing every edge of G.

Example: Which of the directed graphs have an Euler circuit? Of those that do not, which have an Euler path?



H_1: neither; H_2: an Euler circuit, e.g., a, g, c, b, g, e, d, f, a; H_3: an Euler path, e.g., c, a, b, c, d, b

Necessary Conditions for Euler Circuits and Paths

Euler Circuit $\Rightarrow$ The degree of every vertex must be even

- Each time the circuit passes through a vertex, it contributes two to the vertex's degree.
- The circuit starts with a vertex a and ends at a, then contributes two to $deg(a)$.

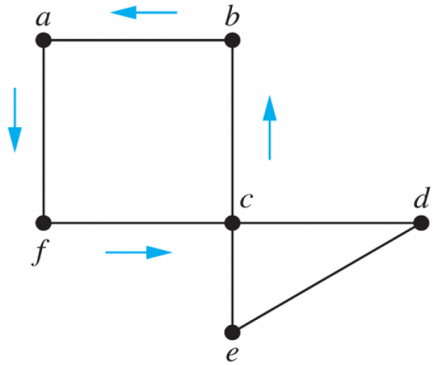
Euler Path $\Rightarrow$ The graph has exactly two vertices of odd degree

- The initial vertex and the final vertex of an Euler path have odd degree.

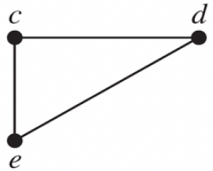
SUSTech Southern University of Science and Technology

Sufficient Conditions for Euler Circuits and Paths

G is a connected multigraph with ≥ 2 vertices, all of even degree.



G



H

We will form a simple circuit that begins at an arbitrary vertex a of G, building it edge by edge.

The path begins at a, and it must terminate at a. This is because every time we enter a vertex other than a, we can leave it.

An Euler circuit has been constructed if all the edges have been used. Otherwise, consider the subgraph H obtained from G by deleting the edges already used. Every vertex in H has even degree ...

Algorithm for Constructing an Euler Circuit

ALGORITHM 1 Constructing Euler Circuits.

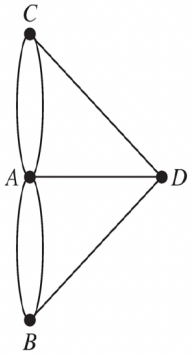
procedure *Euler*(G: connected multigraph with all vertices of even degree)
circuit := a circuit in G beginning at an arbitrarily chosen vertex with edges successively added to form a path that returns to this vertex
$H := G$ with the edges of this circuit removed
while H has edges
    subcircuit := a circuit in H beginning at a vertex in H that also is an endpoint of an edge of *circuit*
    $H := H$ with edges of *subcircuit* and all isolated vertices removed
    circuit := *circuit* with *subcircuit* inserted at the appropriate vertex
return *circuit* {*circuit* is an Euler circuit}

SUSTech Southern University of Science and Technology

Euler Circuits and Paths

Theorem: A connected multigraph with at least two vertices has an Euler circuit if and only if each of its vertices has even degree.

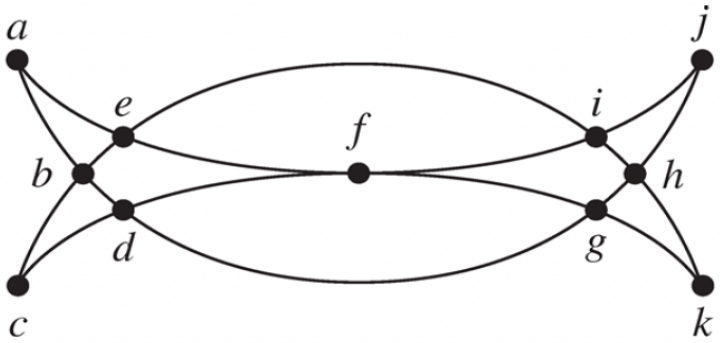
Theorem: A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree.



No Euler circuit, no Euler path

SUSTech Southern University of Science and Technology

Euler Circuits and Paths: Example

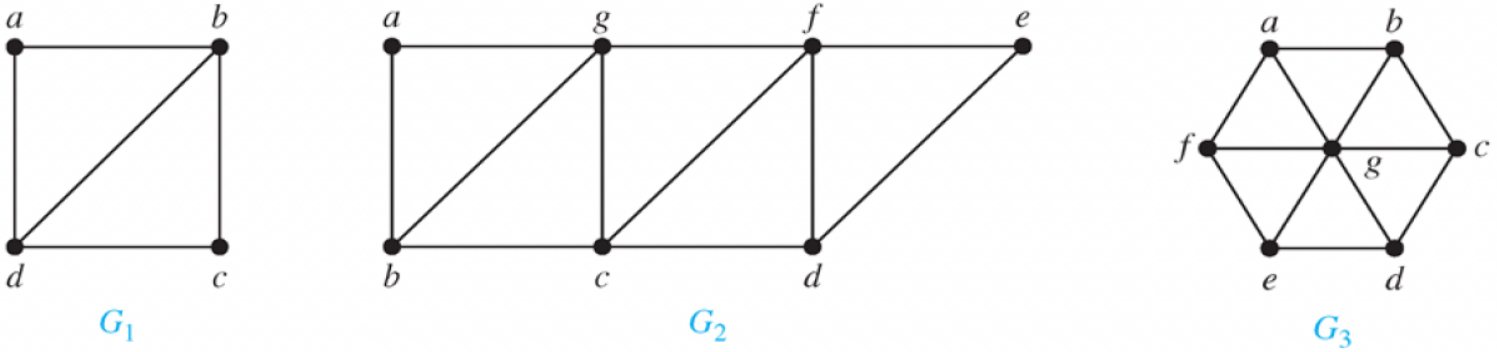


It has such a circuit because all its vertices have even degree.
We will use the algorithm to construct an Euler circuit:

- Form the circuit a, b, d, c, b, e, i, f, e, a;
- Obtain the subgraph H by deleting the edges in this circuit and all vertices that become isolated;
- Form the circuit d, g, h, j, i, h, k, g, f, d in H;
- Splice this new circuit into the first circuit at the appropriate place produces the Euler circuit
  a, b, d, g, h, j, i, h, k, g, f, d, c, b, e, i, f, e, a.

SUSTech Southern University of Science and Technology

Euler Circuits and Paths: Example



- G_1 contains exactly two vertices of odd degree, namely, b and d. Hence, it has an Euler path that must have b and d as its endpoints.
- G_2 has exactly two vertices of odd degree, namely, b and d. So it has an Euler path that must have b and d as endpoints.
- G_3 has no Euler path because it has six vertices of odd degree.

SUSTech Southern University of Science and Technology

Applications of Euler Paths and Circuits

Finding a path or circuit that traverses each

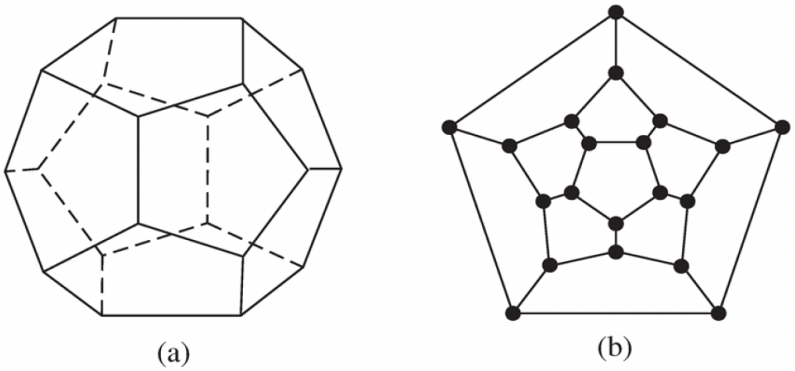
- street in a neightborhood
- road in a transportation network
- link in a communication network
- ...

SUSTech Southern University of Science and Technology

Hamilton Paths and Circuits

Euler paths and circuits contained every edge only once.

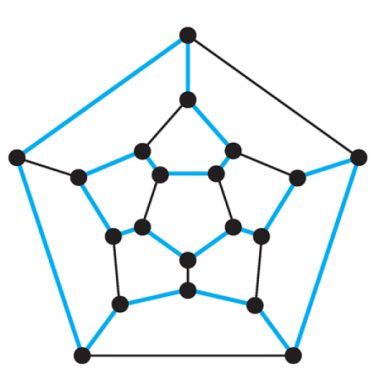
What about containing every vertex exactly once?



Hamilton Paths and Circuits

Euler paths and circuits contained every edge only once.

What about containing every vertex exactly once?

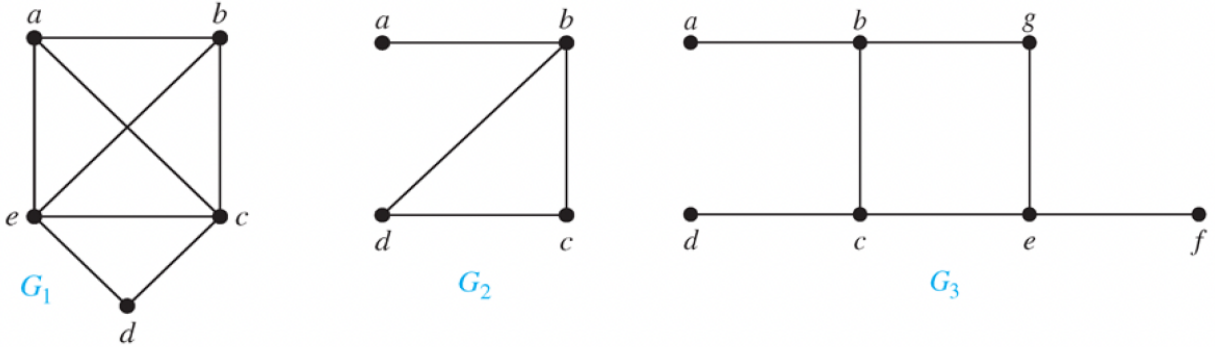


SUSTech Southern University of Science and Technology

Hamilton Paths and Circuits

Definition: A simple path in a graph G that passes through every vertex exactly once is called a Hamilton path, and a simple circuit in a graph G that passes through every vertex exactly once is called a Hamilton circuit.

Example: Which of these simple graphs has a Hamilton circuit or, if not, a Hamilton path?



- G_1 has a Hamilton circuit: a, b, c, d, e, a;
- G_2 has no Hamilton circuit (because containing every vertex must contain the edge a, b twice), but it has a Hamilton path;
- G_3 has neither, because any path containing all vertices must contain one of the edges $\{a, b\}$, $\{e, f\}$, and $\{c, d\}$ more than once.

Sufficient Conditions for Hamilton Circuits

No known simple necessary and sufficient conditions are known for the existence of a Hamilton circuit.

But, there are some useful sufficient conditions.

Dirac's Theorem: If G is a simple graph with $n \geq 3$ vertices such that the degree of every vertex in G is $\geq n/2$, then G has a Hamilton circuit.

Ore's Theorem: If G is a simple graph with $n \geq 3$ vertices such that $deg(u) + deg(v) \geq n$ for every pair of nonadjacent vertices, then G has a Hamilton circuit.

Example: Show that K_n has a Hamilton circuit whenever $n \geq 3$.

Hamilton path problem $\in$ NP-Complete

SUSTech Southern University of Science and Technology

Applications of Hamilton Paths and Circuits

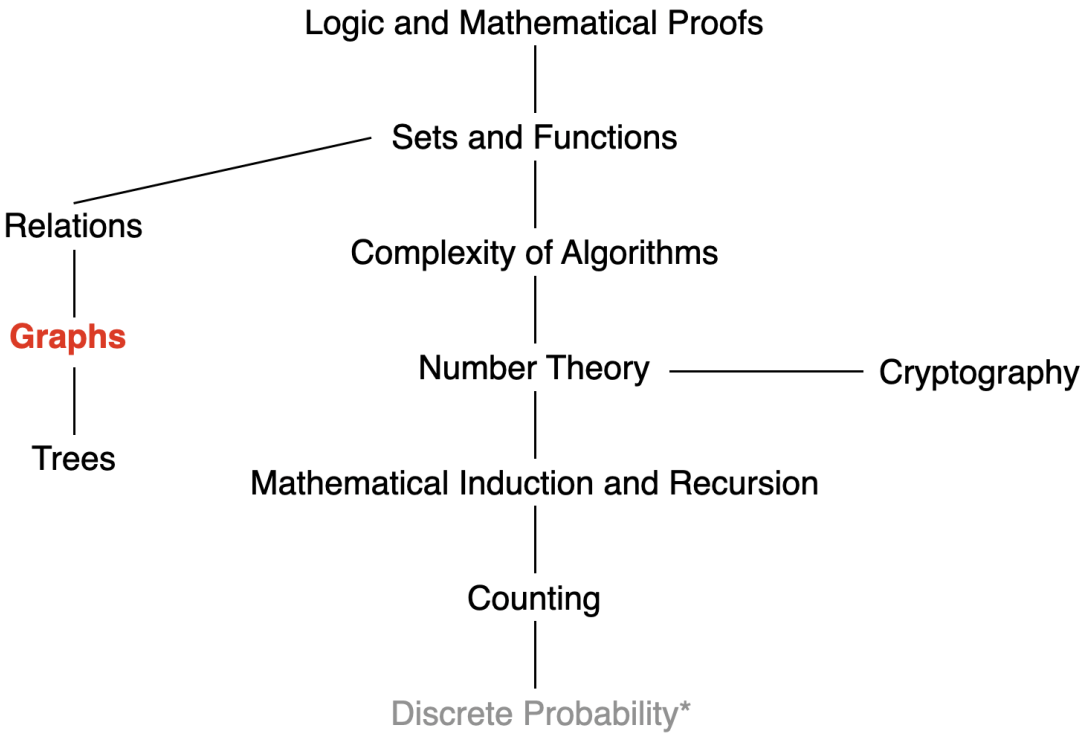
A path or a circuit that visits each city, or each node in a communication network exactly once, can be solved by finding a Hamilton path.

Traveling Salesperson Problem (TSP) asks for the shortest route a traveling salesperson should take to visit a set of cities.

the decision version of the TSP $\in$ NP-Complete

SUSTech Southern University of Science and Technology

This Lecture



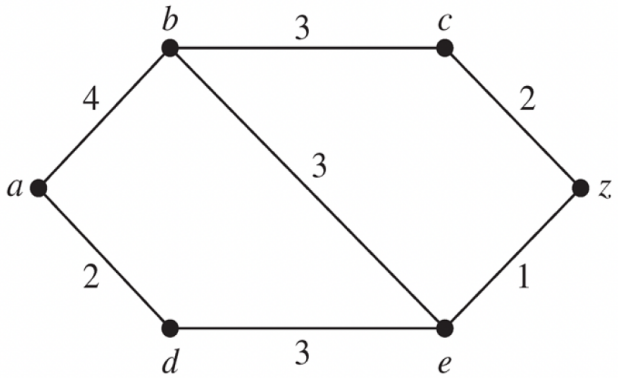
Graph and terminologies, representing graphs and graph isomorphism, connectivity, Euler and Hamiliton path, shortest-path problem ...

SUSTech Southern University of Science and Technology

Shortest Path Problems

Using graphs with weights assigned to their edges

Such graphs are called weighted graphs and can model lots of questions involving distance, time consuming, fares, etc.



What is the length of a shortest path between a and z?

SUSTech Southern University of Science and Technology

Dijkstra's Algorithm

S: a distinguished set of vertices
$L(v)$: the length of a shortest path from a to v that contains vertices only in S

(i) Set $L(a) = 0$ and $L(v) = \infty$ for all v, $S = \emptyset$

(ii) While $z \notin S$

$u :=$ a vertex not in S with $L(u)$ minimal

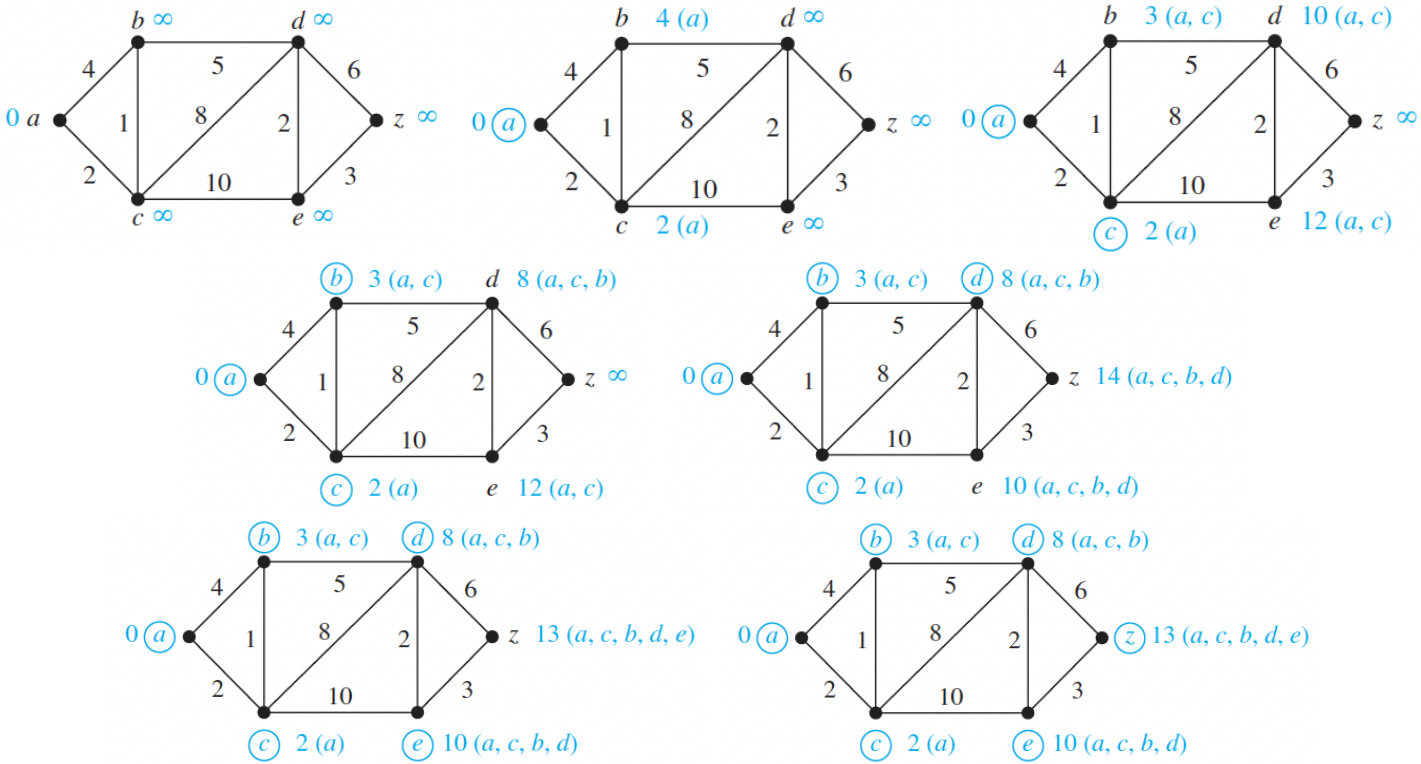
$S := S \cup \{u\}$

For all vertices v not in S

$L(v) := \min\{L(u) + w(u, v), L(v)\}$

SUSTech Southern University of Science and Technology

Dijkstra's Algorithm



$S = \{a, c, b, d, e, z\}$

$L(a) = 0,\ L(b) = 3,\ L(c) = 2,\ L(d) = 8,\ L(e) = 10,\ L(z) = 13$

Dijkstra's Algorithm

Theorem: Dijkstra's algorithm finds the length of a shortest path between two vertices in a connnected simple undirected weighted graph.

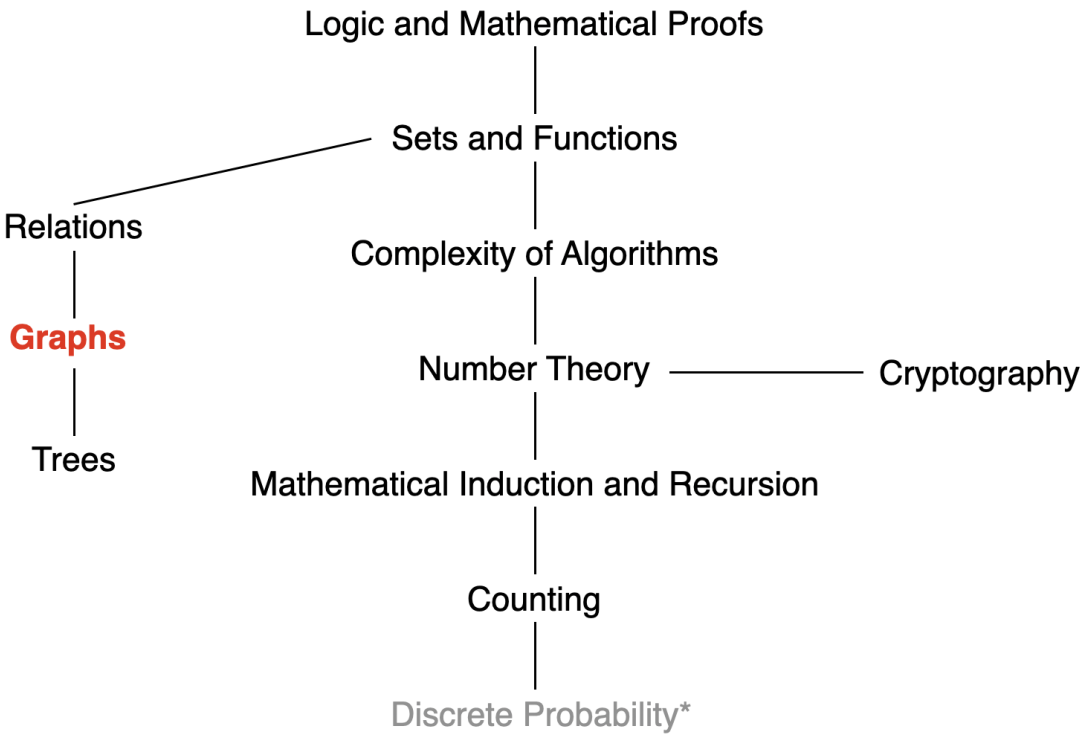
Proof by induction ... (P713 on textbook)

Theorem: Dijkstra's algorithm uses $O(n^2)$ operations (additions and comparisons) in a connected simple undirected weighted graph with n vertices.

- $n-1$ iterations
- no more than $2(n-1)$ operations are used at each iteration

SUSTech Southern University of Science and Technology

Next Lecture



..., Euler and Hamiliton path, shortest-path problem, Planar Graphs