CS 577 - Discrete Primer

Marc Renault

Department of Computer Sciences
University of Wisconsin – Madison

Spring 2023

TopHat Section 001 Join Code: 020205
TopHat Section 002 Join Code: 394523

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

Discrete Mathematics

Definition

Rigorous mathematical study of discrete structures.

Discrete Mathematics

Definition

Rigorous mathematical study of discrete structures.

Key Discrete Concepts for CS 577

Core

- Logic
- Sets
- Recurrences
- Relations and Function
- Graphs and Trees
- Counting

Discrete Mathematics

Definition

Rigorous mathematical study of discrete structures.

Key Discrete Concepts for CS 577

Core

- Logic
- Sets
- Recurrences
- Relations and Function
- Graphs and Trees
- Counting

Applied in CS 577

- Proofs esp. Induction
- Invariants
- Program Correctness

Logic

Propositions

Definition

A statement that is either true or false.

Propositions

Definition

A statement that is either true or false.

Example

True proposition:

- Empire is the best Star Wars movie

Propositions

Definition

A statement that is either true or false.

Example

True proposition:

- ~~Empire is the best Star Wars movie~~
- Ottawa is the capital of Canada.

Propositions

Definition

A statement that is either true or false.

Example

True proposition:

- ~~Empire is the best Star Wars movie~~
- Ottawa is the capital of Canada.

False proposition:

- Toronto is the capital of Canada.

Propositions

Definition

A statement that is either true or false.

Example

True proposition:

- ~~Empire is the best Star Wars movie~~
- Ottawa is the capital of Canada.

False proposition:

- Toronto is the capital of Canada.

Operations

- And: $\wedge$, &, &&
- Or: $\vee$, |, ||
- Negation: $\neg$, !
- Implies: $\implies$
- If and only if (iff): $\iff$
  $P \iff Q \equiv P \implies Q \wedge Q \implies P$

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F				
F	T				
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F			
F	T				
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F		
F	T				
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	
F	T				
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T				
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F			
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T		
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F				
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F			
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T		
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T				

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T			

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T	T		

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T	T	T	

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T	T	T	F

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T	T	T	F

Logical Equivalence

TopHat 1: Is $P \implies Q$ equivalent to $\neg P \implies \neg Q$?

Truth Tables

a	b	$a \wedge b$	$a \vee b$	$a \implies b$	$\neg a$
F	F	F	F	T	T
F	T	F	T	T	T
T	F	F	T	F	F
T	T	T	T	T	F

Logical Equivalence

TopHat 1: Is $P \implies Q$ equivalent to $\neg P \implies \neg Q$?
Exercise: Prove it!

Predicates

Definition

For an underlying domain D. A predicate is a mapping of D to propositions.

PREDICATES

Definition

For an underlying domain D. A predicate is a mapping of D to propositions.

Quantifiers

- For all: $\forall$. $\forall x \in \mathbb{Z}, Even(x) \iff Odd(x+1)$
- There exists: $\exists$. $\exists$person $\in$ This Room, $LovesStarWars(x)$
- Order matters when combining quantifiers!

Predicates

Definition

For an underlying domain D. A predicate is a mapping of D to propositions.

Quantifiers

- For all: $\forall$. $\forall x \in \mathbb{Z}, Even(x) \iff Odd(x+1)$
- There exists: $\exists$. $\exists$person $\in$ This Room, $LovesStarWars(x)$
- Order matters when combining quantifiers!

Logical Equivalence

TopHat 2: What is the logical equivalence of $\neg(\forall x S(x))$?

Sets

Sets

Definition

- A well-defined collection of elements from some domain. Each element in a set is unique.
- A *multiset* may contain duplicates.

Sets

Definition

- A well-defined collection of elements from some domain. Each element in a set is unique.
- A *multiset* may contain duplicates.

Ex: $A = \{11, 12, 13\}$
$B = \{x \in \mathbb{Z} \mid 10 < x \leq 13\}$

Sets

Definition

- A well-defined collection of elements from some domain. Each element in a set is unique.
- A *multiset* may contain duplicates.

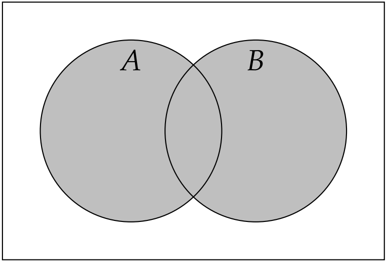
Ex: $A = \{11, 12, 13\}$
$B = \{x \in \mathbb{Z} \mid 10 < x \leq 13\}$

Basic Notations

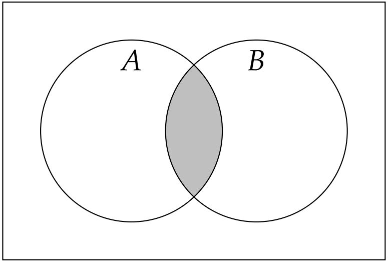
$A \subset B$	A is a proper subset of B, meaning that A contains some (or none) of the elements of B but not all.
$A \subseteq B$	A is subset of B and A may contain all of the elements of B.
$\lvert A \rvert$	The cardinality of A is the number of elements in the set.

Set Operations

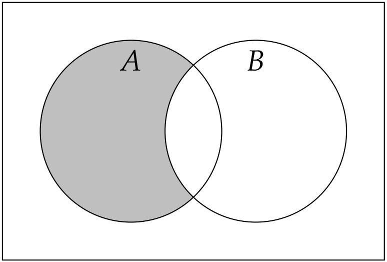
Union: $A \cup B$



Intersection: $A \cap B$

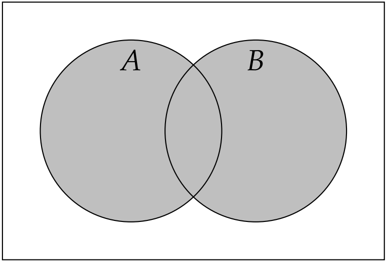


Set difference: $A \setminus B$ or $A - B$

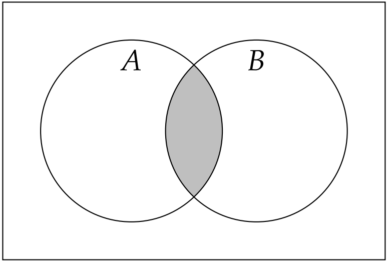


Set Operations

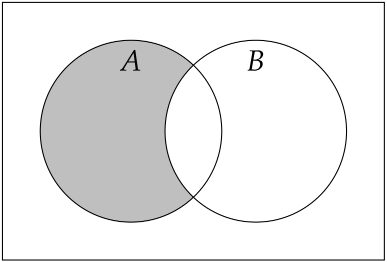
Union: $A \cup B$



Intersection: $A \cap B$



Set difference: $A \setminus B$ or $A - B$



Other Notions

$\emptyset$ **or** $\{\}$	The null or empty set.
$\mathcal{P}(A)$	Power set of A. A set of all possible subsets of A (including $\emptyset$).

TopHats

TopHat 3

What is $\{a, b, c\} \setminus \{c, d, e\}$?

TopHats

TopHat 3

What is $\{a, b, c\} \setminus \{c, d, e\}$?

TopHat 4

What is the size of $\mathcal{P}(A)$ for some set A?

Relations and Functions

Relations

Cartesian Product

For two set A and B, $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$.

Relations

Cartesian Product

For two set A and B, $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$.

Definition

A relation between sets A and B is a defined subset of $A \times B$.
Ex: $R = \{(x, y) \mid x \neq y\}$

Relations

Definition

A relation between sets A and B is a defined subset of $A \times B$.
Ex: $R = \{(x, y) \mid x \neq y\}$

Properties of Relations

Reflexive If $\forall a \in A, R(a, a)$. (*antireflexive*: $\forall a \in A, \neg R(a, a)$)

Symmetric If $\forall a, b \in A, R(a, b) \iff R(b, a)$. (*antisymmetric*: $\forall a, b \in A, R(a, b) \cap R(b, a) \implies a = b$)

Transitive If $\forall a, b, c \in A, R(a, b) \cap R(b, c) \implies R(a, c)$.

Relations

Definition

A relation between sets A and B is a defined subset of $A \times B$.
Ex: $R = \{(x, y) \mid x \neq y\}$

Properties of Relations

Reflexive If $\forall a \in A, R(a, a)$. (*antireflexive*: $\forall a \in A, \neg R(a, a)$)

Symmetric If $\forall a, b \in A, R(a, b) \iff R(b, a)$. (*antisymmetric*: $\forall a, b \in A, R(a, b) \cap R(b, a) \implies a = b$)

Transitive If $\forall a, b, c \in A, R(a, b) \cap R(b, c) \implies R(a, c)$.

Types of Relations

- Equivalence Relations: reflexive, symmetric, and transitive.
- Order Relations: antisymmetric and transitive.
- Functions

Functions

Definition

$f : A \to B$ is a function from A to B. That is for every $a \in A$ there is at most one $b \in B$.
Ex. $f(x) = y + 1$ for $x, y \in \mathbb{R}$.

Functions

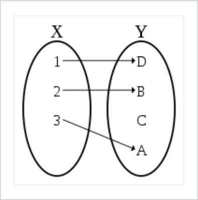
Definition

$f : A \to B$ is a function from A to B. That is for every $a \in A$ there is at most one $b \in B$.
Ex. $f(x) = y + 1$ for $x, y \in \mathbb{R}$.

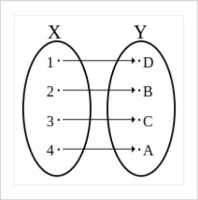
Terminology

- **Domain**: The values of A.
- **Range / Codomain**: The values of B

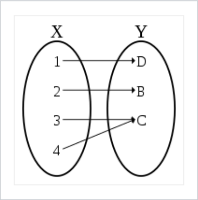
Functions



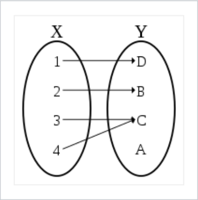
An injective non-surjective function (injection, not a **bijection**)



An injective surjective function (**bijection**)



A non-injective surjective function (surjection, not a **bijection**)



A non-injective non-surjective function (also not a **bijection**)

Types of Functions

- one-to-one / injective
- onto / surjective
- bijection (both onto and one-to-one)

Induction

Proof by Induction

What is induction?

- The most important proof technique in discrete math and CS.
- It proves that $P(n)$ holds for every natural number n, i.e., $n = 0, 1, 2, 3, \ldots$.

Proof by Induction

What is induction?

- The most important proof technique in discrete math and CS.
- It proves that $P(n)$ holds for every natural number n, i.e., $n = 0, 1, 2, 3, \ldots$.

Induction Formula

Step 1 State the induction hypothesis.

Step 2 Show that the induction hypothesis holds for the base case(s).

Step 3 Assume hypothesis is true for k, show that it holds for $k + 1$.

Proof by Induction

Induction Formula

Step 1 State the induction hypothesis.

Step 2 Show that the induction hypothesis holds for the base case(s).

Step 3 Assume hypothesis is true for k, show that it holds for $k+1$.

Special Types of Induction

- Strong induction: we assume true for 1 to k instead of just k.
- Structural induction: we are reasoning about a structure that we map to the natural numbers.

Proof by Induction

Induction Formula

Step 1 State the induction hypothesis.

Step 2 Show that the induction hypothesis holds for the base case(s).

Step 3 Assume hypothesis is true for k, show that it holds for $k+1$.

Induction Exercises

- Show $\sum_1^n 2^n = 2^{n+1} - 2$.

Proof by Induction

Induction Formula

Step 1 State the induction hypothesis.

Step 2 Show that the induction hypothesis holds for the base case(s).

Step 3 Assume hypothesis is true for k, show that it holds for $k+1$.

Induction Exercises

- Show $\sum_1^n 2^n = 2^{n+1} - 2$.
- Show, for $n \geq 5$, $4n < 2^n$.

Proofs

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Proof by Picture Actually not valid!

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Direct Proof Series of implications.

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Direct Proof Series of implications.

Indirect Proof (Contrapositive) To show $P \implies Q$, we show that $\neg Q \implies \neg P$.

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Direct Proof Series of implications.

Indirect Proof (Contrapositive) To show $P \implies Q$, we show that $\neg Q \implies \neg P$.

If and only if $P \iff Q$ means proving $P \implies Q$ and $Q \implies P$.

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Direct Proof Series of implications.

Indirect Proof (Contrapositive) To show $P \implies Q$, we show that $\neg Q \implies \neg P$.

If and only if $P \iff Q$ means proving $P \implies Q$ and $Q \implies P$.

Proof by Contradiction Assume claim is not true, and derive a contradiction.

Proofs

Definition

A proof of a proposition P is a chain of logical deductions ending in P and starting from some set of axioms.

Types (other than induction)

Direct Proof Series of implications.

Indirect Proof (Contrapositive) To show $P \implies Q$, we show that $\neg Q \implies \neg P$.

If and only if $P \iff Q$ means proving $P \implies Q$ and $Q \implies P$.

Proof by Contradiction Assume claim is not true, and derive a contradiction.

Proof by Cases (Brute Force / Exhaustion) Split into cases and prove separately for each case.

Counting

Counting

Basic Techniques

- k-to-1 Rule: Is there a k to 1 ratio between 2 sets?
- Sum Rule: Combine disjoint sets; add cardinality.
- Product Rule: Cartesian product of sets; multiply cardinality.

Counting

Basic Techniques

- k-to-1 Rule: Is there a k to 1 ratio between 2 sets?
- Sum Rule: Combine disjoint sets; add cardinality.
- Product Rule: Cartesian product of sets; multiply cardinality.

Perms and Comb

- **k-Permutation**: $k!$
- **r-Permutation of n items**: ${}_nP_r = P(n,r) = \frac{n!}{(n-r)!}$
- **r-Combination of n items**: ${}_nC_r = C(n,r) = \frac{n!}{r!(n-r)!}$

Counting

Basic Techniques

- k-to-1 Rule: Is there a k to 1 ratio between 2 sets?
- Sum Rule: Combine disjoint sets; add cardinality.
- Product Rule: Cartesian product of sets; multiply cardinality.

Perms and Comb

- ***k*-Permutation**: $k!$
- ***r*-Permutation of *n* items**: ${}_nP_r = P(n, r) = \frac{n!}{(n-r)!}$
- ***r*-Combination of *n* items**: ${}_nC_r = C(n, r) = \frac{n!}{r!(n-r)!}$

Pigeonhole Principal

If n pigeons are placed into m holes, and $n > m$, then at least one hole has more than one pigeon.

Invariants

Invariants

Definition

- A property of a system that holds after every step.

Invariants

Definition

- A property of a system that holds after every step.
- Often critical to showing program correctness.

Invariants

Definition

- A property of a system that holds after every step.
- Often critical to showing program correctness.

Robot Exercise

Suppose we have a robot which walks on a 2-dimensional grid. The rows and columns of the grid are labelled by integers. Our robot starts at position $(0, 0)$, and can only move diagonally, one square at a time. Can we get to $(8, 9)$? Why or why not?

Program Correctness

Program Correctness

Definition

A program/algorithm is correct if it is:

- **Partial correctness/correct/sound** (any returned value is true), and
- **Termination/complete** (returns a value for all valid inputs).

Program Correctness

Definition

A program/algorithm is correct if it is:

- **Partial correctness/correct/sound** (any returned value is true), and
- **Termination/complete** (returns a value for all valid inputs).

Proving correctness

- Requires 2 proofs (one for soundness and one for completeness).
- Often requires identifying invariants and induction.

Recurrences

Recurrences

Definition

An inductive (or recursive) definition of a sequence.

Recurrences

Definition

An inductive (or recursive) definition of a sequence.

Methods for Solving Recurrences

- Guess Method / Recurrence Tree
- Unwind
- Master Theorem

Recurrences

Definition

An inductive (or recursive) definition of a sequence.

Methods for Solving Recurrences

- Guess Method / Recurrence Tree
- Unwind
- Master Theorem

Exercises

Assume $T(1) = 1$ for all.

- $T(n) = T(n/2) + 1$
- $T(n) = T(n/2) + n$
- $T(n) = 3T(n/3) + n$

Graphs and Trees

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)
- Trees

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)
- Trees
- Digraph

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)
- Trees
- Digraph
- Directed Acyclic Graph (DAG)

Graphs

Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)
- Trees
- Digraph
- Directed Acyclic Graph (DAG)
- Bipartite

Graphs

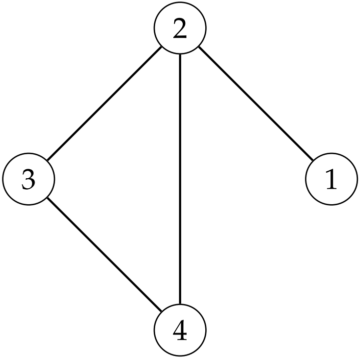
Definition

A graph G is a pair $G = (V, E)$, where V is a set of vertices/nodes and E is a set of edges/arcs connecting a pair of vertices. That is, $E \in V \times V$.

Some Special Graphs

- Complete graph (K_4)
- Cycle (C_4)
- Path (P_4)
- Trees
- Digraph
- Directed Acyclic Graph (DAG)
- Bipartite
- Forests

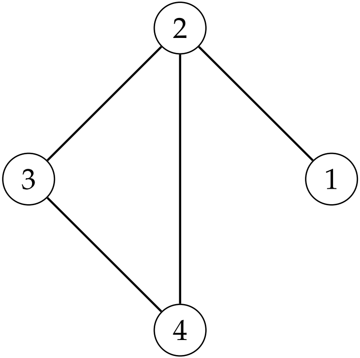
Graph Encodings



Representations

- **Adjacency matrix**: $|V|$ by $|V|$ matrix with a 1 if nodes are adjacent.

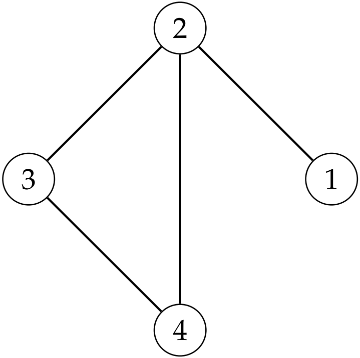
Graph Encodings



Representations

- **Adjacency matrix**: $|V|$ by $|V|$ matrix with a 1 if nodes are adjacent.
- **Adjacency list**: For each node, list adjacent nodes.

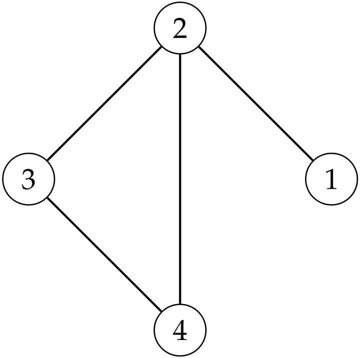
Graph Encodings



Representations

- **Adjacency matrix**: $|V|$ by $|V|$ matrix with a 1 if nodes are adjacent.
- **Adjacency list**: For each node, list adjacent nodes.
- **Edge list**: List of all node pairs representing the edges.

Graph Encodings



Representations

- **Adjacency matrix**: $|V|$ by $|V|$ matrix with a 1 if nodes are adjacent.
- **Adjacency list**: For each node, list adjacent nodes.
- **Edge list**: List of all node pairs representing the edges.
- **Incidence matrix**: $|V|$ by $|E|$ matrix with a 1 if node is incident to the edge.

Trees

Definition

- A connected graph without cycles.
- A single node may be designated as the *root* of the tree.
- Any node with degree 1 that is not the root is a *leaf*.

Trees

Definition

- A connected graph without cycles.
- A single node may be designated as the *root* of the tree.
- Any node with degree 1 that is not the root is a *leaf*.

Properties of a tree T

1. If $|V| \geq 2$, (unrooted) T has at least 2 leaves.
2. For all nodes u and v, there exists one path between them in T.
3. $|V| = |E| + 1$ for $|V| \geq 1$.

Trees

Definition

- A connected graph without cycles.
- A single node may be designated as the *root* of the tree.
- Any node with degree 1 that is not the root is a *leaf*.

Properties of a tree T

1. If $|V| \geq 2$, (unrooted) T has at least 2 leaves.
2. For all nodes u and v, there exists one path between them in T.
3. $|V| = |E| + 1$ for $|V| \geq 1$.

TopHat 6

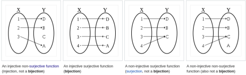
Is P_{10} a tree?

Appendix

References

Image Sources I

WISCONSIN UNIVERSITY OF WISCONSIN–MADISON https://brand.wisc.edu/web/logos/



https://en.wikipedia.org/wiki/Bijection