Function

A function f from a set X to a set Y, denoted $f{:}X \to Y$, is a relation satisfying the following properties:
(F1) $\forall x \in X \ \ \exists y \in Y \ \ (x,y) \in f$.
(F2) $\forall x \in X \ \ \forall y_1, y_2 \in Y \left(\left((x,y_1) \in f \wedge (x,y_2) \in f\right) \to y_1 = y_2\right)$

Definition: Another View of Function
Let $f : X \to Y$ be the type signature of function.
$$\forall x \in X \ \exists y \in Y,\ \{y\} = \{b \mid (x,b) \in f\}"\ "$$
Preimage: If $f(x)=y$, then x is a **preimage** of y.

Setwise image and preimage
Let $f{:}X \to Y$ be a function
Let f^{-1} be its inverse function $Y \to X$ (that may exists)
Let **setwise_image**(f) : $\mathcal{P}(X) \to \mathcal{P}(Y)$ be a function from power set X to power set Y
Let **setwise_preimage**(f) : $\mathcal{P}(Y) \to \mathcal{P}(X)$ be a function power set Y to power set X
If $A \subseteq X$, then setwise_image(f)(A)=$\{f(x) : x \in A\}$.
If $B \subseteq Y$, then setwise_preimage(f)(B) $= \{x \in X : f(x) \in B\}$

Common Relations Summary

R : A x B	f : A → B
R^{-1} : B x A	f^{-1} : B → A (may not exists)

setwise_image(R) : $\mathcal{P}$(A) → $\mathcal{P}$(B)
setwise_preimage(R) : $\mathcal{P}$(B) → $\mathcal{P}$(A)

Other Functions from relation R ?

A → $\mathcal{P}$(B)
B → $\mathcal{P}$(A)
A x B → Bool
( ) → $\mathcal{P}$(A x B)

Preimage and inverse function:

- $\boldsymbol{f^{-1}}$ (a) *or setwise_preimage*(f) (a) as a setwise preimage function always exists, whereas inverse function f^{-1} (x) may or may not exists.
- To denote the setwise preimage of a single element in the co-domain, we must write $\boldsymbol{f^{-1}}$ $(\{b\})$ instead of f^{-1} (b).

Domain, co-domain, range
Let $f{:}A \to B$ be a function from set A to set B.
- A is the **domain** of f and B the **co-domain** of f.
- The **range** of f is the (setwise) image of A under f:
$$\{b \in B : b = f(a) \text{ for some } a \in A\}.$$

Sequence (of infinite length): A **sequence** $a_0, a_1, a_2, \cdots$ can be represented by a function a whose domain is $\mathbb{Z}_{\geq 0}$ that satisfies $a(n) = a_n$ for every $n \in \mathbb{Z}_{\geq 0}$.

String (of finite length): Let A be a set. A **string** or a word over A is an expression of the form $a_0 a_1 a_2 \cdots a_{l-1}$ where $l \in \mathbb{Z}_{\geq 0}$ and $a_0, a_1, a_2, \cdots, a_{l-1} \in A$. Here l is called the **length** of the string. The **empty string** ε is the string of length 0.

Theorem 7.1.1 Function Equality
Two functions $f{:}A \to B$ and $g{:}C \to D$ are equal, iff
- $A = C$ and $B = D$
- $f(x) = g(x)\ \forall x \in A$.

Injection: $\forall x_1, x_2 \in X \ \ (f(x_1) = f(x_2) \Rightarrow x_1 = x_2)$.
A function $f{:}X \to Y$ is **not injective** iff
$$\exists x_1, x_2 \in X \ \ (f(x_1) = f(x_2) \wedge x_1 \neq x_2)$$
Surjection: $\forall y \in Y \ \ \exists x \in X \ \ (y = f(x))$. Every element in the co-domain has at least one preimage. So, **range = co-domain**.
A function $f{:}X \to Y$ is **not surjective** iff
$\exists y \ \in Y \ \ \forall x \in X \ \ (y \neq f(x))$.
Bijective: iff f is injective and surjective, i.e.
$$\forall y \in Y \ \ \exists! x \in X \ \ (y = f(x)).$$
Inverse function: Let $f{:}X \to Y$. Then $g{:}Y \to X$ is an **inverse** of f iff $\forall x \in X \ \ \forall y \in Y \ \ (y = f(x) \Leftrightarrow x = g(y))$.
Uniqueness of inverses: If g_1 and g_2 are inverses of $f{:}X \to Y$, then $g_1 = g_2$.
Theorem 7.2.3: If $f{:}X \to Y$ is a bijection, then $f^{-1}{:}Y \to X$ is also a bijection. In other words, $f{:}X \to Y$ is bijective iff f has an inverse.
Composition of Functions: Let $f{:}X \to Y$ and $g{:}Y \to Z$ be functions. Define a new function $g \circ f{:}X \to Z$ as follows:
$$(g \circ f)(x) = g(f(x))\ \forall x \in X.$$
Theorem 7.3.1 Composition with an Identity Function
If f is a function from a set X to a set Y, and id_X is the identity function on X, and id_Y is the identity function on Y, then
$$f \circ id_X = f$$
$$id_Y \circ f = f$$
$$\begin{tikzcd} {} \arrow[r, "id_X"] \arrow[dr, "f"'] & X \arrow[d, "f"] \arrow[dr, "f"] & \\ & Y \arrow[r, "id_Y"'] & Y \end{tikzcd}$$

Theorem 7.3.2 Composition of a Function with Its Inverse
If $f{:}X \to Y$ is a bijection with inverse function $f^{-1}{:}Y \to X$, then
$$f^{-1} \circ f = id_X$$
$$f \circ f^{-1} = id_Y$$
Theorem: Associativity of Function Composition
Let $f{:}A \to B,\ g{:}B \to C$ and $h{:}C \to D$. Then
$$(h \circ g) \circ f = h \circ (g \circ f).$$
$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"'] \arrow[dr, "h \circ g"] & \\ & C \arrow[r, "h"'] & D \end{tikzcd}$$

Theorem 7.3.3, 7.3.4:
- If $f{:}X \to Y$ and $g{:}Y \to Z$ are both injective, then $g \circ f$ is injective.
- If $f{:}X \to Y$ and $g{:}Y \to Z$ are both surjective, then $g \circ f$ is surjective.

$\mathbb{Z}_n$: The quotient $\mathbb{Z}/\!\sim_n$ where $\sim_n$ is the congruence-mod-n relation on $\mathbb{Z}$, is denoted $\mathbb{Z}_n$.
Addition and Multiplication on $\mathbb{Z}_n$:
Define addition $+$ and multiplication $\bullet$ on $\mathbb{Z}_n$ as follows: whenever $[x],[y] \in \mathbb{Z}_n$
$$[x]+[y] = [x+y] \ \text{ and } \ [x] \bullet [y] = [x \bullet y]$$
Multiplication on $\mathbb{Z}_n$ is well defined:
For all $n \in \mathbb{Z}^+$ and all $[x_1],[y_1],[x_2],[y_2] \in \mathbb{Z}_n$,
$$[x_1] = [x_2] \text{ and } [y_1] = [y_2] \Rightarrow \ [x_1] \bullet [y_1] = [x_2] \bullet [y_2].$$

Mathematical Induction

Let $P(n)$ denotes the property on all integers $n \geq a$

Weak (regular) induction (or 1PI)

If

- $P(a)$ holds
- For every $k \geq a$, $P(k) \Rightarrow P(k+1)$

Then $P(n)$ holds for all $n \geq a$.

Strong induction (or 2PI)

If

- $P(a)$ holds
- For every $k \geq a$, $(P(a) \wedge P(a+1) \wedge \cdots \wedge P(k)) \Rightarrow P(k+1)$

Then $P(n)$ holds for all $n \geq a$.

Strong induction (or 2PI) (other variations possible)

If

- $P(a), P(a+1), \cdots, P(b)$ hold
- For every $k \geq b$, $P(k) \Rightarrow P(k+b-a+1)$

Then $P(n)$ holds for all $n \geq a$.

Well-Ordering Principle for the Integers

Every nonempty subset of $\mathbb{Z}_{\geq 0}$ has a smallest element.

Proof (by contradiction):

1. Suppose not, i.e. let $S \subseteq \mathbb{Z}_{\geq 0}$ be non-empty with no smallest element.
2. For each $n \in \mathbb{Z}_{\geq 0}$, let $P(n)$ be the proposition "$n \notin S$".
3. Inductive step:
   - 3.1. Let $k \in \mathbb{Z}_{\geq 0}$ such that $P(0), P(1), \cdots, P(k-1)$ are true, i.e., $0, 1, \cdots, k-1 \notin S$.
   - 3.2. If $k \in S$, then k is the smallest element of S by the induction hypothesis as $S \subseteq \mathbb{Z}_{\geq 0}$, which contradicts our assumption that S has no smallest element
   - 3.3. So $k \notin S$ and thus $P(k)$ is true.
4. Hence $\forall n \in \mathbb{Z}_{\geq 0}$ $P(n)$ is true by 2PI.
5. This implies $S = \emptyset$, contradicting line 1 that S is non-empty.

Structural Induction

Recursive definition of of a set S.

base clause	Specify that certain elements, called **founders**, are in S: if c is a **founder**, then $c \in S$.
recursion clause	Specify certain functions, called constructors, under which the set S is closed: if f is a constructor and $x \in S$, then $f(x) \in S$.
minimality clause	Membership for S can always be demonstrated by (finitely many) successive applications of the clauses above

Structural induction over S.

To prove that $\forall x \in S\ P(x)$ is true, where each $P(x)$ is a proposition, it suffices to:

(basis step) show that $P(c)$ is true for every founder c; and

(induction step) show that $\forall x \in S(P(x) \Rightarrow P(f(x)))$ is true for every constructor f.

In words, if <u>all the founders satisfy a property</u> P, and P is preserved by all constructors, then all elements of S satisfy P.

Cardinality

Pigeonhole Principle: Let A and B be finite sets. If there is an **injection** $f: A \to B$, then $|A| \leq |B|$.

Dual Pigeonhole Principle: Let A and B be finite sets. If there is a **surjection** $f: A \to B$, then $|A| \geq |B|$.

Finite set and Infinite set

Let $\mathbb{Z}_n = \{1,2,3, \ldots, n\}$, the set of positive integers from 1 to n.

A set S is said to be **finite** iff S is empty, or there exists a bijection from S to $\mathbb{Z}_n$ for some $n \in \mathbb{Z}^+$.

A set S is said to be **infinite** if it is not finite.

Cardinality: The **cardinality** of a finite set S, denoted $|S|$, is

(i) 0 if $S = \emptyset$, or

(ii) n if $f: S \to \mathbb{Z}_n$ is a bijection.

Equality of Cardinality of Finite Sets

Let A and B be any finite sets.

$|A| = |B|$ iff there is a bijection $f: A \to B$.

Same Cardinality (Cantor): Given any two sets A and B. A is said to have the **same cardinality** as B, written as $|A| = |B|$, iff there is a bijection $f: A \to B$.

Theorem 7.4.1 Properties of Cardinality

The same-cardinality relation is an equivalence relation.

For all sets A, B and C:

a. **Reflexive**: $|A| = |A|$.

b. **Symmetric**: $|A| = |B| \to |B| = |A|$.

c. **Transitive**: $(|A| = |B|) \wedge (|B| = |C|) \to |A| = |C|$.

Countably infinite: The set A having the same cardinality as $\mathbb{Z}^+$ is called countably infinite.

Countable Sets: $\mathbb{Z}^+$, $\mathbb{Q}^+$, $\mathbb{Z}^+ \times \mathbb{Z}^+$, $\mathbb{N}$

1. $|(0,1)| = |\mathbb{R}|$ (But they are **uncountable**)
2. $\aleph_0 = |\mathbb{Z}^+| = |\mathbb{N}|$

Definition: Cardinal numbers

Define $\aleph_0 = |\mathbb{Z}^+|$. (Some authors use $\mathbb{N}$ instead of $\mathbb{Z}^+$.)

$\aleph$ is pronounced "aleph", the first letter of the Hebrew alphabet. This is the first cardinal number.

Definition: Countably infinite

A set S is said to be **countably infinite** (or, S has the cardinality of natural numbers) iff $|S| = \aleph_0$.

Definitions: Countable set and Uncountable set

A set is said to be **countable** iff it is finite or countably infinite.

A set is said to be **uncountable** if it is not countable

$\mathbb{Z}$ is countable

$$f(n) = \begin{cases} n/2, & \text{if } n \text{ is an even positive integer} \\ -(n-1)/2, & \text{if } n \text{ is an odd positive integer} \end{cases}$$

Proposition 9.1

An infinite set B is countable if and only if there is a sequence $b_0, b_1, b_2, \cdots \in B$ in which every element of B appears exactly once.

Lemma 9.2: Countability via Sequence

An infinite set B is countable if and only if there is a sequence $b_0, b_1, b_2, \cdots$ in which every element of B appears.

Theorem 7.4.2 (Cantor)

The set of real numbers between 0 and 1,

$$(0,1) = \{x \in \mathbb{R} \mid 0 < x < 1\}$$

is uncountable.

Theorem 7.4.3

Any subset of any countable set is countable.

Corollary 7.4.4 (Contrapositive of Theorem 7.4.3)

Any set with an uncountable subset is uncountable.

Proposition 9.3

Every infinite set has a countably infinite subset.

Lemma 9.4: Union of Countably Infinite Sets.

Let A and B be countably infinite sets. Then $A \cup B$ is countable.

Counting

Theorem 9.5.2 Permutations with sets of indistinguishable objects

Suppose a collection consists of n objects of which

n_1 are of type 1 and are indistinguishable from each other
n_2 are of type 2 and are indistinguishable from each other
$\vdots$
n_k are of type k and are indistinguishable from each other

and suppose that $n_1 + n_2 + \ldots + n_k = n$. Then the number of distinguishable permutations of the n objects is

$$\binom{n}{n_1}\binom{n-n_1}{n_2}\binom{n-n_1-n_2}{n_3}\cdots\binom{n-n_1-n_2-\cdots-n_{k-1}}{n_k}$$

$$= \frac{n!}{n_1!n_2!n_3!\cdots n_k!}$$

Definition: Multiset

An ***r*-combination with repetition allowed**, or **multiset of size *r***, chosen from a set X of n elements is an unordered selection of elements taken from X with repetition allowed.

If $X = \{x_1, x_2, \cdots, x_n\}$, we write an r-combination with repetition allowed as $[x_{i_1}, x_{i_2}, \cdots, x_{i_r}]$ where each x_{i_j} is in X and some of the x_{i_j} may equal each other.

Theorem 9.6.1 Number of *r*-combinations with Repetition Allowed

The number of r-combination with repetition allowed (multisets of size r) that can be selected from a set of n elements is:

$$\binom{r+n-1}{r}$$

This equals the number of ways r objects can be selected from n categories of objects with repetitions allowed.

How many solutions are there to the give equations?

(a) $x_1 + x_2 + x_3 = 20$, each x_i is a nonnegative integer.

(b) $x_1 + x_2 + x_3 = 20$, each x_i is a positive integer.

(a) $n = 3, r = 20.$

$$\binom{r+(n-1)}{r} = \binom{20+2}{20} = \binom{22}{20} = \frac{22\cdot 21}{2} = 231$$

(b) Convert to: $y_1 + y_2 + y_3 = 17$, each y_i is a nonnegative integer

$n = 3, r = 17.$

$$\binom{r+(n-1)}{r} = \binom{17+2}{17} = \binom{19}{17} = \frac{19\cdot 18}{2} = 171$$

Theorem 9.7.1 Pascal's Formula

Let n and r be positive integers, $r \le n$. Then

$$\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r}$$

	Order Matters	Order Does Not Matter
Repetition Is Allowed	n^k	$\binom{k+n-1}{k}$
Repetition Is Not Allowed	$P(n,k)$	$\binom{n}{k}$

Theorem 6.3.1 Number of elements in a Power Set

If a set X has n ($n \ge 0$) elements, then $\mathcal{P}(X)$ has 2^n elements.

- $\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n$
- $\binom{n}{n-r} = \frac{n!}{(n-r)!(n-(n-r))!} = \frac{n!}{(n-r)!r!} = \binom{n}{r}$

Probability

Definition: Independent Events

If A and B are events in a sample space S, then A and B are **independent**, if and only if,

$$P(A \cap B) = P(A) \cdot P(B)$$

Definition: Pairwise Independent and Mutually Independent

Let A, B and C be events in a sample space S. A , B and C are **pairwise independent**, if and only if, they satisfy conditions 1 – 3 below. They are **mutually independent** if, and only if, they satisfy all four conditions below.

1. $P(A \cap B) = P(A) \cdot P(B)$
2. $P(A \cap C) = P(A) \cdot P(C)$
3. $P(B \cap C) = P(B) \cdot P(C)$
4. $P(A \cap B \cap C) = P(A) \cdot P(B) \cdot P(C)$

Definition: Conditional Probability

Let A and B be events in a sample space S. If $P(A) \ne 0$, then the **conditional probability of *B* given *A***, denoted $P(B|A)$, is

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \quad 9.9.1$$

$$P(A \cap B) = P(B|A) \cdot P(A) \quad 9.9.2 \qquad 9.9.3$$

Theorem 9.9.1 Bayes' Theorem

Suppose that a sample space S is a union of mutually disjoint events $B_1, B_2, B_3, \ldots, B_n$. Suppose A is an event in S, and suppose A and all the B_i have non-zero probabilities. If k is an integer with $1 \le k \le n$, then

$$P(B_k|A) = \frac{P(A|B_k)\cdot P(B_k)}{P(A|B_1)\cdot P(B_1) + P(A|B_2)\cdot P(B_2) + \cdots + P(A|B_n)\cdot P(B_n)}$$

Linearity of Expectation

The expected value of the sum of random variables is equal to the sum of their individual expected values, regardless of whether they are independent. For random variables X and Y,

$$E[X+Y] = E[X] + E[Y]$$

For random variables $X_1, X_2, \cdots, X_n$ and constants $c_1, c_2, \cdots, c_n$,

$$E\left[\sum_{i=1}^{n} c_i \cdot X_i\right] = \sum_{i=1}^{n}(c_i \cdot E[X_i])$$

Graph and Tree

Definition: Undirected Graph

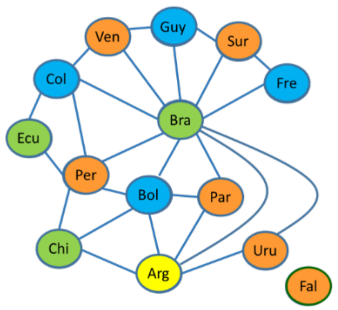
An undirected **graph** G consists of 2 finite sets: a nonempty set V of **vertices** and a set E of **edges**, where each (undirected) edge is associated with a set consisting of either one or two vertices called its **endpoints**.

An edge is said to **connect** its endpoints; two vertices that are connected by an edge are called **adjacent vertices**; and a vertex that is an endpoint of a loop is said to be **adjacent to itself**.

An edge is said to be **incident on** each of its endpoints, and two edges incident on the same endpoint are called **adjacent edges**.

We write $e = \{v, w\}$ for an undirected edge e incident on vertices v and w.

Map Colouring Problem



Solve it as a graph problem.

Draw a graph in which the vertices represent the states, with every edge joining two vertices represents the states sharing a common border.

Such two vertices cannot be coloured with the same colour.

A **vertex colouring** of a graph is an assignment of colours to vertices so that no two adjacent vertices have the same colour.

Definition: Directed Graph

A **directed graph**, or **digraph**, G, consists of 2 finite sets: a nonempty set V of **vertices** and a set E of **directed edges**, where each (directed) edge is associated with an ordered pair of vertices called its **endpoints**.

We write $e = (v, w)$ for a directed edge e from vertex v to vertex w.

Definition: Simple Graph

A **simple graph** is an undirected graph that does not have any loops or parallel edges. (That is, there is at most one edge between each pair of distinct vertices.)

Definition: Complete Graph

A **complete graph** on ***n* vertices**, $n > 0$, denoted K_n, is a simple graph with n vertices and exactly one edge connecting each pair of distinct vertices.

Definition: Bipartite Graph

A **bipartite graph** (or bigraph) is a simple graph whose vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V.

Definition: Complete Bipartite Graph

A **complete bipartite graph** is a bipartite graph on two disjoint sets U and V such that every vertex in U connects to every vertex in V.

If $|U| = m$ and $|V| = n$, the complete bipartite graph is denoted as $K_{m,n}$.

Definition: Subgraph of a Graph

A graph H is said to be a **subgraph** of graph G iff every vertex in H is also a vertex in H, every edge in H is also an edge in G, and every edge in H has the same endpoints as it has in G.

Definition: Degree of a Vertex and Total Degree of a Graph

Let G be a graph and v a vertex of G. The **degree** of v, denoted **deg(v)**, equals the number of edges that are incident on v, with an edge that is a loop counted twice.

The **total degree of *G*** is the sum of the degrees of all the vertices of G.

Theorem 10.1.1 The Handshake Theorem

If the vertices of G are $v_1, v_2, \ldots, v_n$, where $n \geq 0$, then the total degree of G

$= \deg(v_1) + \deg(v_2) + \ldots + \deg(v_n) = 2 \times$ (the number of edges of G).

Corollary 10.1.2

The total degree of a graph is even.

Proposition 10.1.3

In any graph there are an even number of vertices of odd degree.

Definition: Indegree and outdegree of a Vertex of a Directed Graph

Let $G=(V,E)$ be a directed graph and v a vertex of G. The **indegree** of v, denoted **deg⁻(v)**, is the number of directed edges that end at v. The **outdegree** of v, denoted **deg⁺(v)**, is the number of directed edges that originate from v.

Note that $$\sum_{v\in V} deg^-(v) = \sum_{v\in V} deg^+(v) = |E|$$

Definitions

Let G be a graph, and let v and w be vertices of G.

A **walk from *v* to *w*** is a finite alternating sequence of adjacent vertices and edges of G. Thus a walk has the form $v_0 e_1 v_1 e_2 \ldots v_{n-1} e_n v_n$, where the v's represent vertices, the e's represent edges, $v_0=v$, $v_n=w$, and for all $i \in \{1, 2, \ldots, n\}$, v_{i-1} and v_i are the endpoints of e_i. The number of edges, n, is the **length** of the walk.

The **trivial walk** from v to v consists of the single vertex v.

A **trail from *v* to *w*** is a walk from v to w that does not contain a repeated edge.

A **path from *v* to *w*** is a trail that does not contain a repeated vertex.

A **closed walk** is a walk that starts and ends at the same vertex.

A **circuit** (or **cycle**) is a closed walk of length at least 3 that does not contain a repeated edge.

A **simple circuit** (or **simple cycle**) is a circuit that does not have any other repeated vertex except the first and last.

An undirected graph is **cyclic** if it contains a loop or a cycle; otherwise, it is **acyclic**.

Definition: Connectedness

Two vertices v and w of a graph G are **connected** iff there is a walk from v to w.

The graph *G* is connected iff given *any* two vertices v and w in G, there is a walk from v to w. Symbolically, G is connected iff $\forall$ vertices $v, w \in V(G)$, $\exists$ a walk from v to w.

Lemma 10.2.1

Let G be a graph.

a. If G is connected, then any two distinct vertices of G can be connected by a path.
b. If vertices v and w are part of a circuit in G and one edge is removed from the circuit, then there still exists a trail from v to w in G.
c. If G is connected and G contains a circuit, then an edge of the circuit can be removed without disconnecting G.

Definition: Connected Component

A graph H is a **connected component** of a graph G iff

1. The graph H is a subgraph of G;
2. The graph H is connected; and
3. No connected subgraph of G has H as a subgraph and contains vertices or edges that are not in H.

Definitions: Euler Circuit and Eulerian Graph

Let G be a graph. An **Euler circuit** for G is a circuit that contains every vertex and every edge of G.

An **Eulerian graph** is a graph that contains an Euler circuit.

Theorem 10.2.2

If a graph has an Euler circuit, then every vertex of the graph has positive even degree.

Contrapositive Version of Theorem 10.2.2

If some vertex of a graph has odd degree, then the graph doesn't have an Euler circuit.

Theorem 10.2.3

If a graph G is connected and the degree of every vertex of G is a positive even integer, then G has an Euler circuit.

Theorem 10.2.4

A graph G has an Euler circuit iff G is connected and every vertex of G has positive even degree.

Definition: Euler Trail

Let G be a graph, and let v and w be two distinct vertices of G. An **Euler trail/path from *v* to *w*** is a sequence of adjacent edges and vertices that starts at v, ends at w, passes through every vertex of G at least once, and traverses every edge of G exactly once.

Corollary 10.2.5

Let G be a graph, and let v and w be two distinct vertices of G. There is an Euler trail from v to w iff G is connected, v and w have odd degree, and all other vertices of G have positive even degree.

Definitions: Hamiltonian Circuit and Hamiltonian Graph

Given a graph G, a **Hamiltonian circuit** for G is a simple circuit that includes every vertex of G. (That is, every vertex appears exactly once, except for the first and the last, which are the same.)

A **Hamiltonian graph** (also called **Hamilton graph**) is a graph that contains a Hamiltonian circuit.

Proposition 10.2.6

If a graph G has a Hamiltonian circuit, then G has a subgraph H with the following properties:

1. H contains every vertex of G.
2. H is connected.
3. H has the same number of edges as vertices.
4. Every vertex of H has degree 2.

Definition: n^{th} Power of a Matrix

For any $n \times n$ matrix **A**, the **powers of A** are defined as follows:

$\mathbf{A}^0 = \mathbf{I}$ where **I** is the $n \times n$ identity matrix

$\mathbf{A}^n = \mathbf{A}\,\mathbf{A}^{n-1}$ for all integers $n \geq 1$

Theorem 10.3.2

If G is a graph with vertices $v_1, v_2, \ldots, v_m$ and **A** is the adjacency matrix of G, then for each positive integer n and for all integers $i, j = 1, 2, \ldots, m$,

the ij-th entry of $\mathbf{A}^n$ = the number of walks of length n from v_i to v_j.

Definition: Isomorphic Graph

Let $G = (V_G, E_G)$ and $G' = (V_{G'}, E_{G'})$ be two graphs.

G* is isomorphic to *G', denoted $G \cong G'$, if and only if there exist bijections $g: V_G \to V_{G'}$ and $h: E_G \to E_{G'}$ that preserve the edge-endpoint functions of G and G' in the sense that for all $v \in V_G$ and $e \in E_G$,

v is an endpoint of $e \Leftrightarrow g(v)$ is an endpoint of $h(e)$.

Alternative definition (for simple graphs)

Let $G = (V_G, E_G)$ and $G' = (V_{G'}, E_{G'})$ be two simple graphs.

G* is isomorphic to *G' if and only if there exists a permutation $\pi: V_G \to V_{G'}$ such that $\{u, v\} \in E_G \Leftrightarrow \{\pi(u), \pi(v)\} \in E_{G'}$.

Theorem 10.4.1 Graph Isomorphism is an Equivalence Relation

Let S be a set of graphs and let $\cong$ be the relation of graph isomorphism on S.
Then $\cong$ is an equivalence relation on S.

Definition: Planar Graph

A **planar graph** is a graph that can be drawn on a (two-dimensional) plane without edges crossing.

Euler's Formula

For a connected planar simple graph $G = (V, E)$ with $e = |E|$ and $v = |V|$, if we let f be the number of faces, then

$$f = e - v + 2$$

Kuratowski's Theorem:

A finite graph is planar if and only if it does not contain a subgraph that is a subdivision of the complete graph K_5 or the complete bipartite graph $K_{3,3}$.

Useful Conclusions

Consider (labelled) undirected graphs with 3 nodes.
Each node may or may not have a loop (2^3 possibilities),
and each node pair may or may not have an edge ($2^{\binom{3}{2}}$ possibilities).
Total $= 2^3(2^3) = 2^6 = 64$.

(i) #graphs with no loops $= 1^3 2^{\binom{3}{2}} = 8$
$\Rightarrow$ #graphs with loops $= 64 - 8 = 56$.

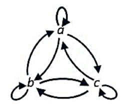
(ii) #graphs with a cycle (choice only in loops) $= 2^3 = 8$.

(iii) #($\sim$loop$\wedge \sim$cycle)$= 2^{\binom{3}{2}} - 1 = 7$.
#cyclic $=$ #(loop$\vee$cycle) $= 64 - 7 = 57$.

(iv) #connected$\Leftrightarrow \wedge, <, >$ or $\triangle$ and any choice of loops: $4(2^3) = 32$ possibilites

(v) tree $\Leftrightarrow \wedge, <,$ or $>$ and no loops: 3 possibilities

(vi) G has exactly 2 components $\Leftrightarrow$ 1 edge only and any choice of loops:
$\binom{3}{1}2^3 = 24$ possibilities.



$\mathcal{D}_3 = \{(\{a,b,c\}, D) \mid D \subseteq \{a,b,c\}^2\} = \{\{a,b,c\}\} \times \mathcal{P}(\{a,b,c\}^2)$.
$|\mathcal{D}_3| = |\mathcal{P}(\{a,b,c\}^2)| = 2^{|\{a,b,c\}^2|} = 2^{(3^2)} = 2^9 = 512$.

(i) G has a loop:
There are $2^6 = 64$ possible Ds *without* loops, so the number with loops is $512 - 64 = 448$.

(ii) G is acyclic:
The acyclic Ds have no loops.
There are $\binom{3}{2} = 3$ pairs of nodes;
for each pair, there are 3 choices: $x \to y$, $x \leftarrow y$, or no edge.

Altogether, there are 3^3 possibilities, 2 of which are cyclic:  
Number of acyclic Gs is $3^3 - 2 = 25$.

(iii) D is reflexive:
all loops are included, and each possible edge may be included/excluded,
so there are $1^3 2^6 = 64$ possible D's.

(iv) D is symmetric:
all (x,y) and (y,x) chosen together: $2^3 \times 2^3 = 64$ possibilities.

(v) D is antisymmetric:
D is antisymmetric iff there is no $x \to y$ and $x \leftarrow y$ pair.
There are $\binom{3}{2}$ pairs of nodes.
For each pair, there are 3 choices: $\circ \to \circ$, $\circ \leftarrow \circ$, or no edge.
For each node, there is a choice of whether to have a loop.
Altogether: $2^3 3^{\binom{3}{2}} = 2^3 3^3 = 216$ possible Ds.

(vi) D is a total order:
all permuations of $a \to b \to c$ (all loops are included)
i.e. $3! = 6$ possibilities.

Tree

Definition: Tree

(The graph is assumed to be undirected here.)

A **graph** is said to be **circuit-free** if and only if it has no circuits.

A simple graph is called a **tree** if and only if it is circuit-free and connected.

A **trivial tree** is a tree that consists of a single vertex.

A simple graph is called a **forest** if and only if it is circuit-free and not connected.

Definitions: Terminal vertex (leaf) and internal vertex

Let T be a tree. If T has only one or two vertices, then each is called a **terminal vertex** (or **leaf**). If T has at least three vertices, then a vertex of degree 1 in T is called a **terminal vertex** (or **leaf**), and a vertex of degree greater than 1 in T is called an **internal vertex**.

Lemma 10.5.1

Any non-trivial tree has at least one vertex of degree 1.

Theorem 10.5.2

Any tree with n vertices ($n > 0$) has $n - 1$ edges.

Lemma 10.5.3

If G is any connected graph, C is any circuit in G, and one of the edges of C is removed from G, then the graph that remains is still connected.

Theorem 10.5.4

If G is a connected graph with n vertices and $n - 1$ edges, then G is a tree.

Definitions: Rooted Tree, Level, Height

A **rooted tree** is a tree in which there is one vertex that is distinguished from the others and is called the **root**.

The **level** of a vertex is the number of edges along the unique path between it and the root.

The **height** of a rooted tree is the maximum level of any vertex of the tree.

Definitions: Child, Parent, Sibling, Ancestor, Descendant

Given the root or any internal vertex v of a rooted tree, the **children** of v are all those vertices that are adjacent to v and are one level farther away from the root than v.

If w is a child of v, then v is called the **parent** of w, and two distinct vertices that are both children of the same parent are called **siblings**.

Given two distinct vertices v and w, if v lies on the unique path between w and the root, then v is an **ancestor** of w, and w is a **descendant** of v.

Definitions: Left Subtree, Right Subtree

Given any parent v in a binary tree T, if v has a left child, then the **left subtree** of v is the binary tree whose root is the left child of v, whose vertices consist of the left child of v and all its descendants, and whose edges consist of all those edges of T that connect the vertices of the left subtree.

The **right subtree** of v is defined analogously.

Definitions: Binary Tree, Full Binary Tree

A **binary tree** is a rooted tree in which every parent has at most two children. Each child is designated either a **left child** or a **right child** (but not both), and every parent has at most one left child and one right child.

A **full binary tree** is a binary tree in which each parent has exactly two children.

Theorem 10.6.1: Full Binary Tree Theorem

If T is a full binary tree with k internal vertices, then T has a total of $2k + 1$ vertices and has $k + 1$ terminal vertices (leaves).

Theorem 10.6.2

For non-negative integers h, if T is any binary tree with height h and t terminal vertices (leaves), then

$$t \le 2^h$$

Equivalently,

$$\log_2 t \le h$$

Definition: Spanning Tree

A **spanning tree** for a graph G is a subgraph of G that contains every vertex of G and is a tree.

Proposition 10.7.1

1. Every connected graph has a spanning tree.
2. Any two spanning trees for a graph have the same number of edges.

Definitions: Weighted Graph, Minimum Spanning Tree

A **weighted graph** is a graph for which each edge has an associated positive real number **weight** . The sum of the weights of all the edges is the **total weight** of the graph.

A **minimum spanning tree** for a connected weighted graph is a spanning tree that has the least possible total weight compared to all other spanning trees for the graph.

If G is a weighted graph and e is an edge of G, then $\boldsymbol{w(e)}$ denotes the weight of e and $\boldsymbol{w(G)}$ denotes the total weight of G.

Algorithm 10.7.1 Kruskal

Input: G [a connected weighted graph with n vertices]

Algorithm:

1. Initialize T to have all the vertices of G and no edges.
2. Let E be the set of all edges of G, and let $m = 0$.
3. While ($m < n - 1$)
   - 3a. Find an edge e in E of least weight.
   - 3b. Delete e from E.
   - 3c. If addition of e to the edge set of T does not produce a circuit, then add e to the edge set of T and set $m = m + 1$

   End while

Output: T [T is a minimum spanning tree for G]

Algorithm 10.7.2 Prim

Input: G [a connected weighted graph with n vertices]

Algorithm:

1. Pick a vertex v of G and let T be the graph with this vertex only.
2. Let V be the set of all vertices of G except v.
3. For $i = 1$ to $n - 1$
   - 3a. Find an edge e of G such that (1) e connects T to one of the vertices in V, and (2) e has the least weight of all edges connecting T to a vertex in V. Let w be the endpoint of e that is in V.
   - 3b. Add e and w to the edge and vertex sets of T, and delete w from V.

Output: T [T is a minimum spanning tree for G]