Discrete Mathematics

Curated computer science study track (not a degree, no licensure claim). Original explanations and practice only, following the reviewed course handout.

From counting on fingers to counting anything

- 2.1-2.3 the proof engine (logic, sets, induction); 2.4-2.6 the counting engine.
- Every proof names its strategy; every identity is checked on a small example first.

Logic and proofs: translate, then choose the shape

1. **Equivalences** $p \to q \equiv \neg p \lor q$; $p \to q \equiv \neg q \to \neg p$; $\neg(p \land q) \equiv \neg p \lor \neg q$; negation swaps $\forall$ with $\exists$.
2. **Shapes** Direct (assume p, derive q) · **contraposition** (prove $\neg q \to \neg p$) · **contradiction** (assume $p \land \neg q$, absurdity).
3. **Named** n^2 even $\to$ n even, by contraposition: $n = 2k+1$ gives $n^2 = 2(2k^2+2k)+1$, odd.

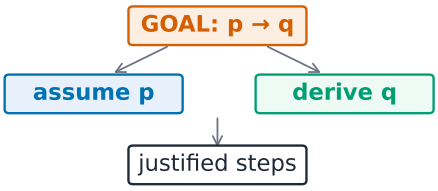
A truth table and its proof tree

the table checks

p	q	p→q	¬q→¬p
T	T	**T**	**T**
T	F	**F**	**F**
F	T	**T**	**T**
F	F	**T**	**T**

matching columns prove:
p → q ≡ ¬q → ¬p (contraposition)

the tree structures



DIRECT
assume p, derive q

CONTRAPOSITION
prove ¬q → ¬p

CONTRADICTION
assume p ∧ ¬q, absurdity

The table checks; the tree structures — three proof shapes choose your data.

Sets, functions and relations: arrows and matrices

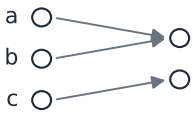
1. **Sets** Union, intersection, difference, complement; the **power set** of A pairs in/out bits - one bit each.
2. **Functions** f(n) = 2n **injective** not **surjective**; $\lfloor n/2 \rfloor$ surjective not injective; $-n$ bijective (own inverse).
3. **Relations** Reflexive = diagonal 1s; symmetric = matrix = transpose; transitive = check length-2 paths.

Set operations, arrow diagrams and relation properties

INJECTIVE

no shared outputs
e.g. 2n

SURJECTIVE



every output hit
e.g. n mod 2

BIJECTIVE

both: a perfect pairing
e.g. $-n$

set algebra

$A \cup B$, $A \cap B$, $A - B$, complement
power set: a bit string per subset
$g \circ f$ reads right to left

relations as matrices

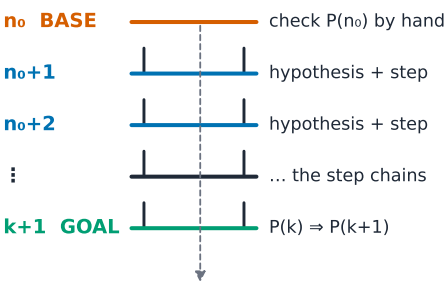
reflexive: diagonal all 1
symmetric: matrix = its transpose
transitive: check paths of length 2

Arrows classify the function; the matrix reads the relation at a glance.

Induction and recursion: the ladder

1. **Ladder** Base $P(n_0)$ by hand; step: assume P(k), derive P(k+1) - induction is chained modus ponens.
2. **Sum** $\Sigma_1{}^n$ i = n(n+1)/2: base $1 = 1 \cdot 2/2$; step $k(k+1)/2 + (k+1) = (k+1)(k+2)/2$.
3. **Faults** No base (ladder on air); hypothesis unused; step silently needs $k \geq 2$ - the horses proof fails at $k+1 = 2$.

The induction ladder: base, hypothesis, step



Σi = n(n+1)/2, proved

base: $1 = 1 \cdot 2/2$ ✓

step: assume $\Sigma_1^k i = k(k+1)/2$

then $\Sigma_1^{k+1} i = k(k+1)/2 + (k+1)$

$= (k+1)(k+2)/2$ ✓

the hypothesis is USED,
the algebra is SHOWN

the three faults

no base: a ladder on air
hypothesis never used
step needs $k \geq 2$ (all horses)

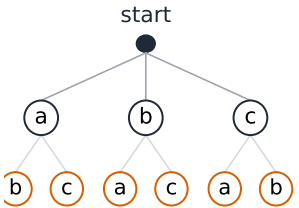
Base plants the ladder; the step climbs one rung.

Counting and combinatorics: two rules, two formulas

1. **Rules** Disjoint alternatives ADD; sequential choices MULTIPLY - 4 routes $\times$ 3 vehicles $= 12$.
2. **Formulas** $P(n,r) = n!/(n-r)!$ ordered; $C(n,r) = n!/(r!(n-r)!)$ unordered - the $r!$ is the whole difference.
3. **IE** $|A \cup B \cup C|$ = singles $-$ pairs $+$ triple: $45+35+30-12-10-8+5 = 85$; team of 3 from 8 with a leader: 168 both ways.

Counting trees: ordered vs unordered

the tree: ordered vs unordered



ordered pairs (ab ≠ ba): $3 \times 2 = 6$

unordered {a,b} = {b,a}: $6 \div 2! = 3$

sequential choices MULTIPLY; disjoint alternatives ADD

the two formulas

$P(n, r) = n!/(n-r)!$ ordered

$C(n, r) = n!/(r!(n-r)!)$ unordered

same choices; combinations divide by the r! arrangements of each set

inclusion–exclusion

$|A \cup B \cup C|$ = singles − pairs + triple

45+35+30 − 12−10−8 + 5 = 85

in none of the three: 15

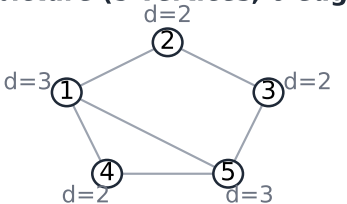
Branches multiply, alternatives add, overcounts subtract.

Graphs and trees: one graph, three representations

1. **Represent Adjacency matrix** n^2 space with $O(1)$ edge test; **adjacency list** $O(|V|+|E|)$ - density decides.
2. **Handshake** Every edge adds 2: $\Sigma \deg = 2|E|$ - drawn graph $3+2+2+2+3 = 12 = 2 \cdot 6$; odd-degree count always even.
3. **Trees** Connected + acyclic $\rightarrow |E| = |V| - 1$; 12-vertex tree, 5 leaves $\rightarrow$ internal degree sum $22 - 5 = 17$.

Small explicit graphs with adjacency structures

the picture (5 vertices, 6 edges)



Σdeg = 3+2+2+2+3 = 12 = 2·6 ✓

adjacency matrix

0	1	0	1	1
1	0	1	0	0
0	1	0	0	1
1	0	0	0	1
1	0	1	1	0

n^2 space, O(1) edge test

adjacency list

1: 2 4 5
2: 1 3
3: 2 5
4: 1 5
5: 1 3 4

O(|V|+|E|): sparse graphs

tree: connected + acyclic, so |E| = |V| − 1
12 vertices, 5 leaves: internal degree sum 22 − 5 = 17

BFS: queue, level by level, shortest hops · DFS: stack, components

Three representations of one graph - and the degree sum checked by handshake.

Discrete probability: counting with weights

1. **Sample** Two dice: 36 pairs; P(sum 7) $= 6/36 = 1/6$ - counting from 2.4 does the work.
2. **Condition** $P(A|B) = P(A \cap B)/P(B)$ - the B-branch renormalised; independence multiplies $P(A)P(B)$.
3. **Bayes** 1 % disease, Se $=$ Sp $= 0.9$: $P(\text{dis}|+) = 0.009/(0.009+0.099) = 1/12 \approx$ 8.3 % - rare stays rare.

An outcome tree with conditional branches

the outcome tree (test for a 1 % disease)

Bayes: P(dis | +) = 0.009 / (0.009 + 0.099)
= 0.009 / 0.108 = 1/12 ≈ 8.3 %

the + branch keeps 0.108 of the world;
the diseased slice of it is 0.009

the rules

P(E) = |E| / |Ω|
(equal outcomes: counting)

P(A | B) = P(A ∩ B) / P(B)
(the B branch, renormalised)

Bayes:
P(A|B) = P(B|A)·P(A) / P(B)

independent:
P(A ∩ B) = P(A)·P(B)

two dice: P(sum 7) = 6/36 = 1/6

The tree shows what conditioning removes: the branch, renormalised.

Check yourself

- **Power set** of a 3-element set...? (8 members.)
- P(7,2) and C(7,2)...? (42 and 21.)
- Edges in a 12-vertex tree...? (11.)
- P(sum 7) with two dice...? (1/6.)