Some mathematical foundations

Kevin De Keyser

November 25, 2017

Abstract

Blub. This was mostly an attempt of trying to write down a few foundations of mathematics. I kind of failed due to the scope of it.
Not missing: Propositional logic, predicate logic, a little bit of proof theory, group theory.
Missing : Type theory, category theory, theory of computation, any kind of higher mathematics (linear algebra, analysis, function theory,...), other types of logic (fuzzy, paraconsistent logics, ...)
I don't plan to continue this.

Contents

1 Predicate logic

1.1 Definitions

Axiom Accepted statement that doesn't require proof. These are rules of inference without antecedent.

Conjecture An idea, which is believed to be true but hasn't yet been proven.

Corollary A proposition that follows from a theorem.

Definition A defintion simplifies wording, is a shorthand for a formulae and doesn't require any proof. For example: A sphere is the collection of all points with distance r from a fixed point.

Lemma A lemma is a mini-theorem, usually a stepping stone for proving a conjecture.

Postulate The same thing as an axiom.

Proof A proof is a demonstration that shows the necessity of a statement being true assuming a few axioms.

Theorem A result that has been proven true (using axioms).

Triviality Something trivial is obvious and needs little proof.

1.2 Propositional logic

Propositional logic, also known as propositional calculus, sentential logic, sentential calculus, zeroth-order logic is the study of logical connectives and their propositions (Definitions explained below). Propositional logic also has a formal notation for these statements by introducing propositional variables and logical connectives and is mainly used to check whether an argument is valid or not.
Another more functional definition, would be that a propositional formula is a function: $\mathbb{B}^n \mapsto \mathbb{B} \,:\, n \in \mathbb{N}^+$

Logical connective An operator, which acts on propositional variables, see 1.2.2.

Propositional variables The variables used in propositional logic are usually denoted as $p\ q\ r$ or as $\varphi\ \psi$. They are elements of the *Boolean domain* usually denoted as: $\{T, F\}$ or $\{\top(\backslash top), \bot(\backslash bot)\}$ or $\{1, 0\}$ or $\mathbb{B}$.
item[Proposition] A proposition is a statement that expresses a concept that can be true or false.
item[Literal] A literal is either a propositional variable or a negated propositional variable.

Judgement A judgement (or assertion) is a metalogical statement which usually stands for the assignment in a logic. 'P is true' is a judgement in propositional logic, 'P is true at time t' is a judgement in temporal logic, 'P is possibly true' is a judgement in modal logic. The other common use for judgement is syntatic, such as the statement: 'P is a well-formed formula'.

Atomic clause An atomic clause is a statement with no logical connectives (Propositions / Predicates).

Evaluation A proposition can be evaluated, whenever the values of the propositional variables are known. The process of deriving the truth value of the proposition is called evaluation and can be done in polynomial running time.

Satisfaction A proposition is satisfiable, if it is possible to evaluate it as true by the means of assigning values to the propositional variables within the proposition. $A \leftrightarrow \neg A$ is not satisfiable for example. Checking satisfiability in general is NP-complete

1.2.1 Truth table notation

Truth tables are a way to write down every possible combination of truth values of each propositional variable and to figure out the truth value of the entire analysed proposition for that combination.

Truth tables conventionally note the propositional variables in the columns on the left and note the propositions to their right. Also in order to evaluate more complex propositions the sub-propositions are written in the order of operation in which they are evaluated. In logic the second row almost always is the row with all propositional variables being $\top$ (true), while in computer science 0 is more commonly used in the second row.

Example Checking whether modus ponendo tollens is a valid argument:

$$\begin{array}{l} \neg(P \land Q) \\ \underline{P \qquad\qquad} \\ \therefore \neg Q \end{array} \quad = \quad (\neg(P \land Q) \land P) \to \neg Q$$

P	Q	$P \land Q$	$\neg(P \land Q)$	$\neg(P \land Q) \land P$	$\neg Q$	$(\neg(P \land Q) \land P) \to \neg Q$
$\top$	$\top$	$\top$	$\bot$	$\bot$	$\bot$	$\top$
$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$
$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$

An alternative way of writing truth tables is by writing down the proposition in the top row and then create a column for each proposition and each logical connective. In these columns you write down the respective truth values of the propositions. Under the connectives you therefore write down the evaluated truth value from the left- and right-handside of that connective, according to its rules. It is recommended to highlight the last evaluated connective, which represents the truth value of the entire proposition.

$(\neg$	$(P$	$\land$	$Q)$	$\land$	$P)$	$\boldsymbol{\to}$	$(\neg$	$Q)$
$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\boldsymbol{\top}$	$\bot$	$\top$
$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\boldsymbol{\top}$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\boldsymbol{\top}$	$\bot$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\boldsymbol{\top}$	$\top$	$\bot$

1.2.2 Logical connectives

This section contains a small table without truth tables and a larger one with proof tables.

Connectivity	Syntax	Other notations	Gate symbols	English representation
Tautology *Truth*	$\top$	T, 1, true, V		True
Contradiction *Falsum*	$\bot$	F, 0, false, O		False
Proposition	p	φ	Buffer	P
Negation	$\neg p$	$\sim p$, $-p$, $!p$, $\overline{p}$, p', $N\varphi$	NOT	not P
Conjunction	$p \wedge q$	$p\&q$, $p\&\&q$, $K\varphi\psi$	AND	P and Q
Inclusive or *Inclusive disjunction*	$p \vee q$	$p\vert q$, $p\vert\vert q$, $p+q$, $A\varphi\psi$	OR	P or Q or both
Conditional	$p \rightarrow q$	$p \supset q$, $C\varphi\psi$	IMPLY	If P then Q P implies Q P only if Q
Converse implication	$p \leftarrow q$	$p \ \subset q$, $B\varphi\psi$		Not Q without P
Biconditional	$p \leftrightarrow q$	$p = q$, $E\varphi\psi$	XNOR	P if and only if Q P iff Q P implies Q and Q implies P Both or neither
Exclusive or *Exclusive disjunction*	$p \oplus q$	$p \veebar q$, $p \not\leftrightarrow q$, $J\varphi\psi$	XOR	Either P or Q P or Q, but not both
Alternative denial *Sheffer's stroke*	$p \uparrow q$	$p\vert q$, $D\ \varphi\psi$	NAND	Not both P and Q
Joint denial *Peirce's arrow* *Quine's dagger*	$p \downarrow q$	$p \dagger q$, $X\varphi\psi$	NOR	Neither P nor Q
Abjunction *Nonimplication*	$p \nrightarrow q$	$p\not\supset q$, $L\varphi\psi$	NIMPLY	Only P and not Q
Converse Nonimplication	$p \nleftarrow q$	$p\not\subset q$, $M\varphi\psi$		Only Q and not P

<table>
<tr><th>Connectivity</th><th>Syntax</th><th>Other notations</th><th>Gate symbols</th><th>Truth Table</th><th>Properties</th><th>English representation</th></tr>
<tr><td><i>Tautology</i>
<i>Truth</i></td><td>$\top$</td><td>T, 1, true, V</td><td></td><td></td><td></td><td>True</td></tr>
<tr><td><i>Contradiction</i>
<i>Falsum</i></td><td>$\bot$</td><td>F, 0, false, O</td><td></td><td></td><td></td><td>False</td></tr>
<tr><td><i>Proposition</i></td><td>p</td><td>φ</td><td>Buffer</td><td></td><td></td><td>P</td></tr>
<tr><td><i>Negation</i></td><td>$\neg p$</td><td>$\sim p$, $-p$, $!p$, $\overline{p}$, p', $N\varphi\psi$</td><td>NOT</td><td>$\begin{array}{c||c} p & \neg p \\ \hline \mathrm{F} & \mathrm{T} \\ \hline \mathrm{T} & \mathrm{F} \end{array}$</td><td></td><td>not P</td></tr>
<tr><td><i>Conjunction</i></td><td>$p \wedge q$</td><td>$p\&q$, $p\&\&q$, $K\varphi\psi$</td><td>AND</td><td>$\begin{array}{c|c||c} p & q & p \wedge q \\ \hline \mathrm{F} & \mathrm{F} & \mathrm{F} \\ \hline \mathrm{F} & \mathrm{T} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{F} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{T} & \mathrm{T} \end{array}$</td><td>commutative
associative
idempotent
identity element $\top$</td><td>P and Q</td></tr>
<tr><td><i>Inclusive or</i>
<i>Inclusive disjunction</i></td><td>$p \vee q$</td><td>$p|q$, $p||q$, $p+q$, $A\varphi\psi$</td><td>OR</td><td>$\begin{array}{c|c||c} p & q & p \vee q \\ \hline \mathrm{F} & \mathrm{F} & \mathrm{F} \\ \hline \mathrm{F} & \mathrm{T} & \mathrm{T} \\ \hline \mathrm{T} & \mathrm{F} & \mathrm{T} \\ \hline \mathrm{T} & \mathrm{T} & \mathrm{T} \end{array}$</td><td>commutative
associative
idempotent
identity element $\bot$</td><td>P or Q or both</td></tr>
<tr><td><i>Conditional</i></td><td>$p \rightarrow q$</td><td>$p \supset q$, $C\varphi\psi$</td><td>IMPLY</td><td>$\begin{array}{c|c||c} p & q & p \rightarrow q \\ \hline \mathrm{F} & \mathrm{F} & \mathrm{T} \\ \hline \mathrm{F} & \mathrm{T} & \mathrm{T} \\ \hline \mathrm{T} & \mathrm{F} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{T} & \mathrm{T} \end{array}$</td><td>not commutative
not associative
not idempotent
no identity element</td><td>If P then Q
P implies Q
P only if Q</td></tr>
<tr><td><i>Converse implication</i></td><td>$p \leftarrow q$</td><td>$p\ \subset q$, $B\varphi\psi$</td><td></td><td>$\begin{array}{c|c||c} p & q & p \leftarrow q \\ \hline \mathrm{F} & \mathrm{F} & \mathrm{T} \\ \hline \mathrm{F} & \mathrm{T} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{F} & \mathrm{T} \\ \hline \mathrm{T} & \mathrm{T} & \mathrm{T} \end{array}$</td><td>not commutative
not associative
not idempotent
no identity element</td><td>Not Q without P</td></tr>
<tr><td><i>Biconditional</i></td><td>$p \leftrightarrow q$</td><td>$p = q$, $E\varphi\psi$</td><td>XNOR</td><td>$\begin{array}{c|c||c} p & q & p \leftrightarrow q \\ \hline \mathrm{F} & \mathrm{F} & \mathrm{T} \\ \hline \mathrm{F} & \mathrm{T} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{F} & \mathrm{F} \\ \hline \mathrm{T} & \mathrm{T} & \mathrm{T} \end{array}$</td><td>commutative
associative
not idempotent
identity element $\top$</td><td>P if and only if Q
P iff Q
P implies Q and Q implies P
Both or neither</td></tr>
</table>

Connectivity	Syntax	Other notations	Gate symbols	Truth Table			Properties	English representation
Exclusive or	$p \oplus q$	$p \veebar q$, $p \nleftrightarrow q$, $J\varphi\psi$	XOR	p	q	$p \oplus q$	commutative	Either P or Q
Exclusive disjunction				F	F	F	associative	P or Q, but not both
				F	T	T	not idempotent	
				T	F	T	identity element $\perp$	
				T	T	F		
Alternative denial	$p \uparrow q$	$p \vert q$, $D\ \varphi\psi$	NAND	p	q	$p \uparrow q$	commutative	Not both P and Q
Sheffer's stroke				F	F	T	not associative	
				F	T	T	not idempotent	
				T	F	T	no identity element	
				T	T	F		
Joint denial	$p \downarrow q$	$p \dagger q$, $X\varphi\psi$	NOR	p	q	$p \downarrow q$	commutative	Neither P nor Q
Peirce's arrow				F	F	T	not associative	
Quine's dagger				F	T	F	not idempotent	
				T	F	F	no identity element	
				T	T	F		
Abjunction	$p \nrightarrow q$	$p \not\supset q$, $L\varphi\psi$	NIMPLY	p	q	$p \nrightarrow q$	not commutative	Only P and not Q
Nonimplication				F	F	F	not associative	
				F	T	F	not idempotent	
				T	F	T	identity element $\perp$	
				T	T	F		
Converse Nonimplication	$p \nleftarrow q$	$p \not\subset q$, $M\varphi\psi$		p	q	$p \nleftarrow q$	not commutative	Only Q and not P
				F	F	F	not associative	
				F	T	T	not idempotent	
				T	F	F	no identity element	
				T	T	F		

1.2.3 Other symbols in propositional logic

Name	Symbol	Meaning
(Logical) Implication	$A \Rightarrow B$ (Semantical mathematically) $A \vDash B$ (Semantical logically) $A \vdash B$ (Syntactical logically)	A conditional, which is always true, is called an implication. However the symbols $\vdash$ and $\vDash$ may only be between logical s
(Logical) Equivalence	$A \Leftrightarrow B$ (Semantical mathematically) $A \equiv B$ (Semantical logically)	A bijection, which is always true, is called an equivalence. An equivalence is a semant
Therefore	$\therefore$	Literally means *therefore* and can be used in a proof, see 1.2.4
Because	$\because$	Literally means *because* and can be used in a proof, see 1.2.4
End of proof	■ □ $Q.E.D$	The gravestone symbol stands for end of proof. $Q.E.D.$ is latin for 'quod erat demonstrandum', meaning 'which is what had to be proven'.
Definition	$x := y$ $x :\Leftrightarrow y$ $x :\equiv y$	This sign means x is defined as y. It is usually used to reduce long statements: p xnor $q := (p \wedge q) \vee (\neg p \wedge \neg q)$

1.2.4 Rules of Inference

Inference is the act of deriving logical conclusions from premises.

Antecedent The antecedent is the collection of premises.

Conclusion Under the assumptions of premises a conclusion must always be logically true.

Consequent The consequent is the collection of conclusions.

Counterexample A counterexample is a way of assigning propositional variables true or false for all premises and making the conclusion false, which implies invalidity.

Deduction Deducing and inferring can be interchanged. They mean exactly the same in logic.

Inference Inference is the act of deriving logical conclusions from premises. Deduction means exactly the same thing.

Paradox The word paradox has multiple meanings.
A veridical paradox is an unintuitive argument, being nonetheless valid. Examples from mathematics include: Monty Hall problem, Hilberts Hotel, Banach-Tarski paradox, Skolem's paradox. Examples from physics include: Twin paradox, Ladder paradox, Schrödinger's cat. If the consequent seems wrong, but the antecedent true, it is an *absurd* argument.
A falsidical paradox (also called *fallacy*) is an argument that follows from not following the rules of the formal system (rules of inference in propositional logic). Examples from mathematics include: Affirming the consequent, Horse paradox (using induction), Existential fallacy, Illicit Minor/Major.
A contradiction is a logical incompability between propositions (sometimes referred to as an *antinomy*). Whenever the propositions (axioms) inside an axiomatic system you work in form a contradiction, the axiomatic system is inconsistent. Famous examples include Russel's paradox (in naive set theory), Zeno's paradox, Galilei's paradox and the Liar's paradox (1.2.7).

Premise Premises are propositions or formulas of propositions assumed to be true.

Syllogism A logical argument with two premises and one conclusion.

Validity An argument is only valid, if it is impossible for the premises to be true and the conclusion to be false. If the premises are true (which is not objective) the argument is **sound**.

Proofs in propositional logic are basically an implication where the premises / conclusions are linked with a conjunction:
(Premise 1 $\wedge$ Premise 2 $\wedge$ ... $\wedge$ Premise N) $\Rightarrow$ (Conclusion 1 $\wedge$ Conclusion 2 $\wedge$ ... $\wedge$ Conclusion N)

Proofs in propositional logic can also be written in the following ways:
Premise 1, ... and Premise N are true, *therefore* Conclusion 1, ... and Conclusion N have to be true.
Conclusion 1, ... and Conclusion N have to be true, *because* Premise 1, ... and Premise N are true.

$$\begin{array}{l} \text{Premise 1} \\ \vdots \\ \text{Premise N} \\ \hline \therefore \text{Conclusion 1} \\ \vdots \\ \therefore \text{Conclusion N} \end{array} \quad \text{or more compact} \quad \begin{array}{l} \text{Premise 1, Premise 2, ..., Premise N} \\ \hline \therefore \text{Conclusion 1, Conclusion 2, ..., Conclusion N} \end{array}$$

Name	Fitch	Inference	Example
Modus ponens	$\to$ *Elim*	$\dfrac{P \to Q, P}{\therefore Q}$	If it rains the streets are wet. It is raining. Therefore the streets are wet.
Modus tollens		$\dfrac{P \to Q, \neg Q}{\therefore \neg P}$	If it rains the streets are wet. The streets aren't wet. Therefore it is not raining.
Biconditional introduction	$\leftrightarrow$ *Intro*	$\dfrac{P \to Q, Q \to P}{\therefore P \leftrightarrow Q}$	If I am breathing then I am alive. If I am alive then I am breathing. Therefore I am alive if and only if I am breathing.
Biconditional elimination	$\leftrightarrow$ *Elim*	$\dfrac{P \leftrightarrow Q}{\therefore P \to Q, Q \to P}$	I am alive if and only if I am alive. Therefore I am alive if I am breathing. Therefore I am breathing if I am alive.
Conjunction introduction	$\wedge$*Intro*	$\dfrac{P, Q}{\therefore P \wedge Q}$	I exist. I think. Therefore I think and I exist.
Conjunction elimination	$\wedge$*Elim*	$\dfrac{P \wedge Q}{\therefore P, Q}$	I exist and I think. Therefore I exist. Therefore I think.
Disjunction introduction	$\vee$*Intro*	$\dfrac{P}{\therefore P \vee Q}$	Humans are real. Therefore humans are real or unicorns are real.
Case analysis		$\dfrac{P \to Q, R \to Q, P \vee R}{\therefore Q}$	If it is raining, the streets are wet. If I spill water on the streets, the streets are wet. It is raining or I am spilling water. Therefore the streets are wet.
Disjunctive syllogism		$\dfrac{P \vee Q, \neg P}{\therefore Q}$	I will eat salad or soup. I will not eat salad. Therefore I will eat soup.
Hypothetical syllogism (Transitivity)		$\dfrac{P \vee Q, Q \vee R}{\therefore P \vee R}$	If rains the streets are wet. If the streets are wet cars have to drive carefully. Therefore if it rains, cars have to drive carefully.
Modus ponendo tollens		$\dfrac{\neg(P \wedge Q), P}{\therefore \neg Q}$	It can not both snow and rain. It rains. Therefore it can't snow.

1.2.5 Rules of replacement

These are common logically equivalent statements, which can be exchanged.

Name	Fitch	Laws	
Commutative laws		$A \land B \Leftrightarrow B \land A$	$A \lor B \Leftrightarrow B \lor A$
Associative laws		$(A \land B) \land C \Leftrightarrow A \land (B \land C)$	$(A \lor B) \lor C \Leftrightarrow A \lor (B \lor C)$
Distributive laws		$(A \lor B) \land C \Leftrightarrow (A \land C) \lor (B \land C)$	$(A \land B) \lor C \Leftrightarrow (A \lor C) \land (B \lor C)$
De Morgan's laws		$\neg(A \land B) \Leftrightarrow \neg A \lor \neg B$	$\neg(A \lor B) \Leftrightarrow \neg A \land \neg B$
Rule of material implication		$A \to B \Leftrightarrow \neg A \lor B$	
Transposition		$A \to B \Leftrightarrow \neg B \to \neg A$	
Absorption		$(A \to B) \Leftrightarrow (A \to (A \land B))$	
Exportation		$A \to (B \to C) \Leftrightarrow ((A \land B) \to C)$	
Falsum introduction	$\bot Intro$	$(A \land \neg A) \Leftrightarrow \bot$	$(A \leftrightarrow \neg A) \Leftrightarrow \bot$
Truth introduction	$\top Intro$	$(A \lor \neg A) \Leftrightarrow \top$	
Negation introdution	$\neg Intro$	$(A \to B) \land (A \to \neg B) \Leftrightarrow \neg A$	$(A \to \bot) \Leftrightarrow \neg A$
Rule of double negation	$\neg Elim$	$\neg\neg A \Leftrightarrow A$	
Truth introduction	$\top Intro$	$(A \lor \neg A) \Leftrightarrow \top$	$(A \ \leftrightarrow A) \Leftrightarrow \top$
Law of excluded middle			
Idempotency		$(A \land A) \Leftrightarrow A$	$(A \lor A) \Leftrightarrow A$

1.2.6 Relationships of conditional statements

Name	Relationship to $p \to q$
Contraposition (the contrapositive)	$\neg q \to \neg p$
Conversion (the converse)	$q \to p$
Inversion (the inverse)	$\neg p \to \neg q$
Negation (the negation)	$\neg(p \to q)$

1.2.7 Self-reference

Propositional logic supports self-reference with the equivalence relation, which can be expressed as a bijunction.
$T \leftrightarrow statement \quad | \, T$ represents the truth value of the statement.

Using this theorem, one can model the **liar paradox** and show that it leads to a contradiction.
$T \leftrightarrow \neg T =$ This sentence is false.
The card paradox, a variation of the liar paradox, also leads to a contradiction:
$T_1 \leftrightarrow T_2 =$ The sentence below is true.
$T_2 \leftrightarrow \neg T_1 =$ The sentence above is false.
Both premises conjugated leads to: $(T_1 \leftrightarrow T_2) \land (T_2 \leftrightarrow \neg T_1)$

Useful rules of inference in self-referring statements:
$T \leftrightarrow (T \to A) \Rightarrow (A \land T)$
$T \leftrightarrow (T \leftrightarrow A) \Rightarrow A$

2 Structural proof theory

Formalism

Structure A structure defines a formula in the following way: $A_1, A_2, ..., A_M \vdash C_1, C_2, ..., C_N$, where A and C are a finite amount of formulas (antecedent & consequent) in any logic (such as propositional logic or first-order logic, which allows conjunction/disjunction), the commas before the $\vdash$ are conjunctions ($\land$) and the commas after the $\vdash$ are disjunctions ($\lor$) and the $\vdash$ itself is a logical implication ($\Rightarrow$).

Hypotheses Γ In order to not write down all axioms/rules of inferences/hypotheses the symbol Γ is introduced. It is a finite (possibly empty) set of formulas which are not relevant to the proof.

Conclusions Δ In order to not write down all possible conclusions the symbol Δ is introduced. It is a finite (possibly empty) set of formulas which are not relevant to the proof.

Judgement A judgement (or assertion) is a metalogical statement. It can be used for assigning values to a proposition in various logics: 'P is true' in propositional logic, 'P is true at time t' in temporal logic, 'P is possibly true' in modal logic. Other uses includes assigning a type to a statement: 'P is a proposition' or 'P is a well-formed formula'. Judgements have a shorthand postfix notation, for e.g.: $(A \land B)\mathit{true}$ or $P\mathit{prop}$ (the brackets in the first example can be ommited).

Note: The material implication in proof theory is more commonly written as $\subset$ instead of $\rightarrow$.

2.1 Natural deduction

Natural deduction proofs transform formulas into another formula using rules (and axioms).
The big difference between sequent calculus and natural deduction is that natural deduction strictly has one conclusion, while sequent calculus can have any finite amount of conclusions. The advantage of having one conclusion is syntactic sugar and allows tree like proofs (see Gentzen-style proofs).
In propositional logic natural deduction transforms propositions into another proposition using rules of inference or replacement. These rules may not change the truth table of the proposition in propositional logic. Generating and checking the entire truth table takes exponential time ($O(2^t)$) and is hard to do for statements with a lot of propositional variables. Natural deduction proofs usually require less steps to verify an equivalence between two statements (see proof sections below) than generating truth tables.
As soon as quantification is introduced to logic it starts to be impossible to fill out a truth table and natural deduction has to be used. Elsewise statements like 'All men are mortal. Socrates is a man. Therefore Socrates is mortal.' cannot be proven formally.

The structure syntax of natural deduction would look like this $J_1, J_2, ..., J_N \vdash J$, however Gentzen-style natural deduction uses a different syntax to describe this. J are judgements:

$$\frac{J_1 \qquad J_2 \qquad \dots \qquad J_N}{J}\,\mathit{name}$$

While the judgements above the line (J_i) is part of the antecedent, J is the consequent. This structure syntax is used for describing *formation rules*, *introduction rules* and *elimination rules* which are rules of inference for judgements. Rules like modus ponens can be used aswell, but they are not

2.1.1 Formation rules

Formation rules are rules of inference for the validity of the formulas. They are generally implied when doing proofs using introduction and elimination rules and are not explicitly stated. Some common formation rules:

$$\frac{A \text{ prop} \qquad B \text{ prop}}{A \land B \text{ prop}}\land F \qquad \frac{A \text{ prop} \qquad B \text{ prop}}{A \lor B \text{ prop}}\lor F \qquad \frac{A \text{ prop} \qquad B \text{ prop}}{A \subset B \text{ prop}}\subset F$$

$$\frac{A \text{ prop}}{\lnot A \text{ prop}}\lnot F \qquad \frac{}{\top \text{ prop}}\top F \qquad \frac{}{\bot \text{ prop}}\bot F$$

2.1.2 Introduction rules

The meaning of a connective is given by its introduction rule in natural deduction. Since in natural deduction there are no truth tables, connectives have to be defined using rules of introduction. The formation rules are not written down, but the full conjunction ($\land$) introduction would look like this:

$$\frac{A \text{ prop} \qquad B \text{ prop} \qquad A \land B \text{ prop} \qquad A \text{ true} \qquad B \text{ true}}{A \land B \text{ true}}\land I$$

Some common introduction rules without assumptions:

$$\frac{A \text{ true} \qquad B \text{ true}}{A \land B \text{ true}}\land I \qquad \frac{A \text{ true}}{A \lor B \text{ true}}\lor I_1 \text{ (or } I_L) \qquad \frac{B \text{ true}}{A \lor B \text{ true}}\lor I_2 \text{ (or } I_R) \qquad \frac{}{\top}\top I$$

A *local reduction* is a special type of deduction where a single formula changes into another using a chain of deduction (using introduction rules). Gentzen-style proofs abreviate this sometimes using $\Longrightarrow_R$ (see below for an example). There also is the *local expansion* which changes a single formula into a large cluster of formulas.

Any statement without a bar on top of it is referred to as a *hypothetical judgement* and is an assumption. It is not referred to as a premise, because it is a judgement. All deductions following after that hypothetical judgement are also hypothetical. See assumption based introductions for the use in a proof. The statement below does not proof that B is true, but it proofs that if $A \land (B \land C)$ is true, then B is true.

$$\cfrac{\cfrac{A \land (B \land C) \text{ true}}{B \land C \text{ true}}\land E_2}{B \text{ true}}\land E_1 \qquad \Longrightarrow_R B \text{ true}$$

2.1.3 Assumption based rules of inference

Tautological (non-hypothetical) proofs need not to start with a $\top$ introduction. Hypothetical judgements are of the form $A_1, A_2, ...A_N \vdash C$, where A_i are assumptions and C is the hypothetical deduction. In propositional logic the above can also be written as a tautology: $\vdash (A_1 \land A_2 ... A_N) \subset C$. This idea can be used to internalize hypothetical judgements into tautological proofs.

Assumptions are introduced in the following way, where a letter references the assumption:

$$\begin{array}{c}\cfrac{}{A \land B \text{ true}}\,u \\ \vdots\end{array}$$

Any deductions following under this line are under the assumption of A. They can only be used in the proof if they are *discharged*, which can be done by using conditional proof ($\subset I$) or an indirect proof / proof by contradiction ($\bot_C$ or $\lnot I$) (however this is not a rule of introduction):

$$\cfrac{\cfrac{\begin{array}{c}\cfrac{}{A \text{ true}}\,u \\ \vdots\end{array}}{B \text{ true}}}{A \subset B}\subset I^u \qquad \cfrac{\cfrac{\begin{array}{c}\cfrac{}{A \text{ true}}\,u \\ \vdots\end{array}}{\bot}}{\lnot A \text{ true}}\lnot I^u \qquad \text{or with an alternative syntax} \qquad \cfrac{\cfrac{\begin{array}{c}\cfrac{[A]^{(1)}}{}\\ \vdots\end{array}}{B \text{ true}}}{A \subset B}\,{}^{(1)} \qquad \cfrac{\cfrac{\begin{array}{c}\cfrac{[A]^{(1)}}{}\\ \vdots\end{array}}{\bot}}{\lnot A \text{ true}}\,{}^{(1)}$$

If a line is deduced from multiple assumptions all assumptions have to be discharged in order to make the prove valid (Only one discharge per line!). See example:

$$\cfrac{\cfrac{\cfrac{\cfrac{}{A \text{ true}}\,u \qquad \cfrac{}{B \text{ true}}\,w}{A \land B \text{ true}}\land I}{B \subset (A \land B) \text{ true}}\subset I^u}{A \subset (B \subset (A \land B)) \text{ true}}\subset I^w$$

2.1.4 Elimination rules

Elimination rules are the opposite of the introduction rules and are important because natural deduction proofs only have one conclusion. In order to seperate the conclusion from the connectives use elimination rules. Some common elimination rules:

$$\dfrac{A \land B \text{ true}}{A \text{ true}} \land E_1 \text{ (or } E_L) \qquad \dfrac{A \land B \text{ true}}{B \text{ true}} \land E_2 \text{ (or } E_R) \qquad \dfrac{A \qquad A \subset B \text{ true}}{B \text{ true}} \subset E \qquad \dfrac{A \text{ true} \qquad \neg A \text{ true}}{\bot} \neg E$$

$$\dfrac{A \lor B \text{ true} \qquad \begin{array}{c} \overline{A \text{ true}}^{\,u} \\ \vdots \\ C \text{ true} \end{array} \qquad \begin{array}{c} \overline{B \text{ true}}^{\,v} \\ \vdots \\ C \text{ true} \end{array}}{C \text{ true}} \lor E^{u,v}$$

2.1.5 Harmony

In order to check whether introduction and elimination rules for a certain connective are sensible (in harmony), they must fullfill two properties:

Local soundness Connectives are sound when their introduction rules have no possibility of generating new truths from a formula.

Local completeness Connectives are complete when their elimination rules can reconstruct the connective.

Both properties can be checked by applying all introduction rules and elimination rules. If we end up with the judgements from the beginning both properties are true. For example the conjunction connective ($\land$):

$$\dfrac{\dfrac{\dfrac{A \text{ true} \qquad B \text{ true}}{A \land B \text{ true}} \land I}{A \text{ true}} \land E_1 \qquad \dfrac{\dfrac{A \text{ true} \qquad B \text{ true}}{A \land B \text{ true}} \land I}{B \text{ true}} \land E_2}{A \land B \text{ true}} \land I$$

Sometimes a communative law is used instead of having 2 $\land$ eliminations and 2 $\lor$ introductions.

2.1.6 Fitch-style proof

Fitch-style is an alternative syntax for natural deduction and is not the only natural deduction syntax used besides the previously shown Gentzen-style (such as Lemmon-style for e.g.).

1	$(A \lor B) \to (C \land D)$	Premise 1
2	$C \to \neg D$	Premise 2
3	$\neg C \lor \neg D$	2, Implication
4	$\neg(C \land D)$	3, De Morgan's Law
5	$\neg(A \lor B)$	1,4, Modens tollens
6	$\neg A \land \neg B$	5, De Morgan's Law
7	$\neg A$	6, Conjunction Elimination

The above proof was a step-by-step proof to show that $((A \lor B) \to (C \land D)) \land A \vdash \neg A$.
Fitch-style natural deduction can also utilize assumptions. By creating a sub-proof with all of the previous statements as premises and additional assumptions (premises in the sub-proof) one can derive new formulas. These new formulas are usually introduced for one of two reasons.

First reason is the *indirect proof* ($\bot I$) also known as proof by contradiction ($\neg E$) where one introduces only a single assumption. If a statement derived from the premises above the introduction now says the exact opposite (the negation) of a statement derived by the assumption then the assumption has been proven incorrect. One then gains a new formula, namely the negation of the assumption. For example:

1	$\neg(A \lor B)$	Premise
2	A	Assumption
3	$(A \lor B)$	2, Conjunction Introduction
4	$\bot$	1,3, Falsum introduction
5	$\neg A$	2-4, Negation introduction

The other common reason to introduce an assumption is a *conditional proof* ($\to$ *introduction*). The assumptions imply that any propositional statement formed from the assumptions is true. This is why one can introduce a conditional. If the assumption ends up being true then the assumption must be true. If the assumption ends up being wrong then the conditional statement is still valid because it is always true if the antecedent is wrong.
$(A_1, A_2 ... A_N) \vdash C$ is the same as $\vdash (A_1 \land A_2 ... A_N) \to C$.

1	$A \to (B \land C)$	Premise 1
2	$(B \lor D) \to E$	Premise 2
3	A	Assumption
4	$B \land C$	1,3, Modus Ponens
5	B	4, Conjunction Elimination
6	$B \lor D$	5, Disjunction introduction
7	E	2,6, Modus ponens
8	$A \to E$	3-7, Conditional proof

After an assumption line stops (you either made a conditional introduction or a falsehood introduction) the line is *discharged.* As an important reminder: It is forbidden to use any line from a discharged assumption because they rely on the assumption. It is perfectly fine to introduce an assumption within an assumption (and so on) as long as you don't discharge two lines at once.

2.2 Sequent calculus

System LK.

Elimination form follows. A true. Sequent means successive according to oxford and basically is a shorthand for (con-)sequent.

Analytical proofs are proofs that don't use the cut theorem.

3 Predicate logic

Predicate logic has all of the features of propositional logic, but uses predicates instead of propositional variables (still variables that can be either $\top$ or $\bot$) and quantification. It uses concepts of naive set theory and vice-versa, which is why the topics of this section and naive set theory overlap.

3.1 Predicates

In english predicates are things which refer to properties. The predicate (sentence) 'x is smooth' is a predicate which expresses the property smooth (not having edges), although it can mean having a non-abrupt movement as well (again english is ambigous). This has the tremendous advantage to automatically generate propositions:
'1 is a natural number'
'2 is a natural number'
...
Instead we can simply create a predicate $p(\_) = \_$ is a natural number | _ is a number between $[1..\infty]$. This is usually done using set-builder notation. $\{\_ | \_ \in \mathbb{T}\}$, where $\mathbb{T}$ is the *truth set* of the predicate. Predicates must not only refer to one thing, predicates can also be relations between objects. For example the predicate 'x is larger than y' requires two objects as input so to speak.

In first-order logic a predicate is formally described as a function with the co-domain $\mathbb{B}$ and is described as $A(p_1, p_2, p_3, ..., p_n) \mapsto \mathbb{B}$, where A is the predicate and p_i symbolizes the i-th parameter of the predicate function. Because not all predicates are bijective the order (and therefore the meaning) of the premises does matter. For example the relation 'father of' is clearly not bijective. Concretely propositional variables are simply predicates, which can either be true or false. The genius thing about abstraction is that propositional variables can represent any predicate. Because not all theories are founded on set theory, predicates are sometimes defined differently.

3.2 Quantifiers

Domain of discourse also known as universe (of discourse) or as the universal set is the set of all entities discussed / used by predicates. *Important:* The universe is never empty!

The following predicate can be filled with natural numbers, but it can also be filled by sets or even swiss cheese. $p(x) = x$ is an even natural number.
$p(Blue\,cheese) = \bot$ This predicate is false.
In order to assign every possible truth statement to the predicate without listing every possible (infinite) element quantifiers are used.

Universal quantifier $\forall x$ The universal quantifier notes a statement for all possible x, yes even blue cheese. In order to reduce claims to a specific set (not the universe) statements can be made like this: $\forall x(x \in \mathbb{N} \rightarrow p(2 * x))$
However this is sometimes implied and left out (which is why the universe is also referred to as the domain of discourse, which does not necessairly include blue cheese). The shorthand form of writing the above would be: $\forall x \in \mathbb{N} p(2 * x)$ and is read as for all natural numbers x, p(2*x) holds true.

Existential quantifier $\exists x$ The existential quantifier notes a statement which is true for at least one element of the universe (possibly blue cheese). The usual way of writing this is $\exists x p(x)$ (there exists an x such that p(x)). If one wants to point out the specific existance of an object in a set one can similairly to the universal quantifier write $\exists x(x \in \mathbb{N} \wedge p(x))$. The shorthand notation is $\exists x \in \mathbb{N} p(x)$.
In order to point out that a statement is true for a certain amount of elements a subscript can be used: $\exists_{=2} x \in \mathbb{Z}(x * x = x)$. To point out uniqueness there is the *uniqueness quantifier*: $\exists_{=1} x = \exists! x$.
$\forall x \forall y \forall z(mother(y, x) \wedge mother(z, x)) \rightarrow y = z$ Leibniz law

Negated quantifiers Negated quantifiers as we use them in english can be simply transformed to one of the above quantifiers.
$\neg \forall x P(x) \Leftrightarrow \exists x \neg P(x)$
$\neg \exists x P(x) \Leftrightarrow \forall x \neg P(x)$

Compound quantifiers (Nested quantifiers) Compound quantifiers are a collection of quantifiers where the order matters! $\forall x \exists y(x < y)$ = For all x there exists a y such that x ¡ y. $\exists x \forall y(x < y)$ There exists and x such that for all y, x ¡ y (This statement is false).

Quantifier seperator Sometimes a . is used after a quantifier instead of brackets. The dot can simply be thought of as brackets from the dot until the end of the expression. For example: $\forall x.x \in A \Leftrightarrow x \in B$ is the same as $(\forall x)(x \in A \Leftrightarrow x \in B)$

3.2.1 Compactness

A set of first-order sentences have a model $\Leftrightarrow$ Every finite subset of the set has a model.
This holds true for all sets (finite, countable and uncountable sets).

3.2.2 Resolution calculus for first order logic

Skolemization.

3.3 Relations

$\forall x \neg elt(x, x)$
$\forall x \forall y \neg(elt(x, y) \wedge elt(y, x))$
$gelt(x, y) :\Leftrightarrow \exists z(elt(x, z) \wedge elt(z, y))$
$geschw(x, y) :\Leftrightarrow \exists z(elt(z, x) \wedge elt(z, y)) \wedge x \neq y$
$onk(x, y) :\Leftrightarrow m(x) \wedge (\exists z(elt(z, x) \wedge gelt(z, y)) \wedge \neg elt(x, y)$
$vorfahr(x, y) :\Leftrightarrow elt(x, y) \vee gelt(x, y)$
$vorfahr(x, y) | (elt(x, y) \rightarrow vorfahr(x, y)) \wedge ((vorfahr(x, y) \wedge vorfahr(y, z)) \rightarrow vorfahr(x, z)) \wedge \forall x \neg vorfahr(x, x)$
$verwandte(x, y) :\Leftrightarrow \exists z(vorfahr(z, x) \wedge vorfahr(z, y)) \wedge x \neq y$

3.4 Quantifier proof methods

$(\forall x)P(x) \vDash (\exists x)P(x)$
$\forall x.(P(x) \wedge Q(x)) \vDash \forall x.P(x) \wedge \forall y Q(y)$
$\exists x.(P(x) \wedge Q(x)) \vDash \exists x.P(x) \wedge \exists y Q(y)$
$\exists y \forall x.P(x, y) \vDash \forall x \exists y.P(x, y)$ ()

3.5 Proof techniques with mathematical examples

Proof techniques from predicate logic can be applied to all predicates and therefore to mathematical statements as well. This section will provide common prove techniques and their examples.

3.5.1 Direct proof

In a direct proof, rules of transformation are used repeatedly.
Example: Proof that two square numbers multiplied with each other always result in another square number:
$a * b = u * u * v * v = u * v * u * v = (uv) * (uv)$

3.5.2 Indirect proof (Modus tollens)

$\neg B \to \neg A \equiv A \to B$
In order to proof $S \Rightarrow T$ simply assume that T is false and then prove that S is false.
Now there is no way that if S is true, T could be false, hence it would contradict the above.

Example:
$x > 0$ is irrational $\Rightarrow \sqrt{x}$ is irrational.
$\sqrt{x}$ is rational $\Rightarrow x$ is rational.
Two rational numbers multiplied together always give a rational number, therefore the above statement is true.
Hence if x is irrational, $\sqrt{x}$ cannot be rational. Therefore $\sqrt{x}$ has to be irrational.

3.5.3 Composition of implications

$(A \to B) \land (B \to C) \vDash A \to C$

3.5.4 Modus Ponens

$A \land (A \to B) \vDash B$
Example:
$2^{3000} \equiv_{3001} 1 \qquad \equiv_{30001} :\Leftrightarrow \%30001$
For each prime number greater than 2 $2^{p-1} \equiv_p = 1$.

$(\forall x \in \ \mathbb{P}) x > 2 \to 2^{x-1} \equiv_p 1$
3001 is a prime number.
$3001 \in \mathbb{P}$
Therefore 2^3000 modulo 3001 is 1.

3.5.5 Case Distinction

$(R_1 \lor R_2 \lor ... \lor R_n) \land (R_1 \to S) \land (R_2 \to S) \land ... \land (R_n \to S) \Rightarrow S$
Example:
If p is prime, then $3|p$ or $3|(p+2)$ or $3|(p+4)$.
$p\%3 = 0 \lor p\%3 = 1 \lor p\%3 = 2$
$(\forall p \in \mathbb{P}) p\%3 = 0 \to 3|p$
$(\forall p \in \mathbb{P}) p\%3 = 1 \to 3|(p+4)$
$(\forall p \in \mathbb{P}) p\%3 = 2 \to 3|(p+2)$

3.5.6 Proof by contradiction

$(\neg A \to B) \land \neg B \ \vDash A$
Alternative syntax: $(A \lor B) \land \neg B \vDash A$
Alternative lemma: $\neg A \to \bot \vDash A$
Famous proof that $\sqrt{2}$ is not rational:
First show that each rational number can be described by $f = \frac{a}{b}$, where a and b are relatively prime (don't share a common divisor). If they would we could simply construct a new fraction $\frac{a/gcd(a,b)}{b/gcd(a,b)}$.
Assume that:
$(\exists a, b \in \mathbb{N}) \sqrt{2} = \frac{a}{b} \land gcd(a,b) = 1$
$(\exists a, b \in \mathbb{N}) 2 = \frac{a*a}{b*b}$
$2 * b^2 = a^2$ (A common mistake of me at least is to write $2 * a * a = b * b$ here).
If a^2 is even, so is $a = 2 * k$.
$2 * b * b = 2 * k * 2 * k = 4 * k^2$
$b^2 = 2 * k^2$, which makes b^2 and therefore b even.
However this would contradict $gcd(a,b) = 1$, because 2 is clearly a factor of both a and b.
So the only assumption we made, must be false.
Therefore $(\neg\exists a, b \in \mathbb{N}) \sqrt{2} = \frac{a}{b} \land gcd(a,b) = 1$

3.5.7 Pidgeonhole principle

Divide n elements into k sets. There must be at least one set which is larger than $\lceil \frac{n}{k} \rceil$. K has to be smaller than n.
Examples:
Let's say you have blue and red socks, so $k = 2$ (set of all red socks and set of all blue socks).
How many socks do you need to have a match of two colors (a set of size 2)?
Or in other words, whats the minimum x, such that: $\lceil \frac{n}{2} \rceil = 2$
$n = 3$. Proof:
n objects are uniquely situated in k partitions. Now assume that the maximum size of the partition set is $\lceil \frac{n}{k} \rceil - 1$. Then there must be $k * (\lceil \frac{n}{k} \rceil - 1)$ objects, which is less than n (Hence $k \lceil \frac{n}{k} \rceil < n + k$ and subtracting k from both sides: $k * \lceil \frac{n}{k} - k < n$) This makes the following statement, the smallest whole number for which the inequality holds: $k \lceil \frac{n}{k} \rceil \geq n$).

3.5.8 Proof by counter-example

$\neg \forall x P(x) \equiv \exists x \neg P(x)$
This is usually the easiest form of proof and in mathematics only requires some algebraic manipulation to solve for x or some trial and error.

3.5.9 Proof by induction

Theorem: $P(0) \land \forall n (P(n) \to P(n+1)) \vDash \forall n n \geq 0 \to P(n)$
Proof: $F \equiv (P(0) \land \forall n (P(n) \to P(n+1)) \to \forall P(n)$
$\neg F \equiv P(0) \land \forall n (P(n) \to P(n+1)) \land \neg \forall n P(n)$
$\neg F \equiv P(0) \land \forall n (P(n) \to P(n+1)) \land \exists n \neg P(n)$
Thus there must be a non empty set $\{P(n) | n \geq 0\}$. Due to the well-ordering principle, this set has a smallest element $P(a)$.
Because $P(0) \land \neg P(a), \quad a \geq 1$. $P(a-1)$ must be true then and according to the rule $(P(a-1) \to P(a) \land P(a-1)) \to P(a)$ due to modus ponens.
$\neg F \to P(a) \land \neg P(a)$.
So F has to be true.
Generally to prove something by induction one writes down 3 steps:
1. Induction hypothesis *(GER: Induktionshypothese)*. The thing you want to prove.
2. Base step *(GER: Induktionsverankerung)*. The starting point of your theorem (above this is $b = 0$).
3. Inductive step *(GER: Induktionsschritt)*. Here it is proven that from $n : n \geq b$ follows $n + 1$.

Fundamental theorem of algebra:
Induction hypothesis: Every prime number has a prime factorization.
Base step: 2 is a prime number and is uniquely factored as 2^1 (Hence no number smaller than 2 is prime).
Inductive step: Assume x has a unique prime factorization.
If $x + 1$ is prime, its prime factorization is $(x+1)^1$ (uniquely).
If $x + 1$ is not prime, by definition $\exists p, q . p * q = x + 1, p \geq 2, q \geq 2$.
Because p and q are smaller than $x + 1$ they already have a prime factorization (using statement of strong induction).
So if $x + 1$ is not prime, $x + 1 =$ (prime factors of p) $*$ (prime factors of q) and has therefore prime factors. Proving uniqueness is a bit harder.

3.5.10 Proof of uniqueness

Technique:

1. Proof that $\exists x P(x)$
2. Now simply do existential elimination twice (Assume $P(x_1)$ and $P(x_2)$).
3. Show that $x_1 = x_2$

4 Logic

4.1 Proof system *(GER: Beweissystem)*

Statement *(GER: Aussagen)* (Proposition)
$\tau : S \mapsto \{0,1\}$
$S = \{0,1\}^*$ (Set of statements, such as $s = 5$ is prime or program s terminates.)
$\tau(S)$ truth function
$\mathcal{P}$ the set of proofs.
For example S could be all primes and P is the set of proofs.
$\phi : S \times P \mapsto \{0,1\}$ (verification function)
$\phi(s,p) = 1$ means that p is a valid proof for statement s in the proof system. *(GER: Validierungsfunktion (Evaluationsfunktion?))*
The proof can be verified in polynomial deterministic time.
Soundness (*(GER: Korrektheit)*):
We should not use existential and quantors here:
$\exists \tau(s) = 0 \Rightarrow$ for all $p\phi(s,p) = 0$
$\phi(s,p) = 1 \Rightarrow \tau(s) = 1$
Completeness *(GER: Vollständigkeit)*:
If $\tau(s) = 1$, then exists a $p \in P : \phi(s,p) = 1$
There exists s such that $\tau(s) = 1 \Rightarrow$, then there exists a p such that $\in P : \phi(s,p) = 1$

The problem of verifying whether a hamilton path exists can be checked in P-time.
Given a graph and proof whether there exists a hamilton path. This is not a proof, which can be done in EXP time.
However it does work for prime numbers cleverly.
Sudoku: How to prove a sudoku has no solution (This can normally only be done in exponential time, unless you find a simple contradiction).

Example: 6.4
Given a statement s in S $s = q$ is a prime number. Show that it is not a prime number (Can be done easily by showing the prime factors).
In fact one can prove the complement as well. It is a prime number as well (Pratt 1973?):
1. Give all prime factors of $n-1$. 2. A recursive proof of primality for each (claimed) factor of $n-1$. 3. A generator g of the group $\mathbb{Z}_n^*$. To verify it must hold that $g^{n-1} \equiv_n 1$. Also obviously $Ord(g) = n-1$, otherwise we could just claim 1 is a generator.
This can be done efficiently by using a prime factor of $n-1$: $g^{\frac{n-1}{p_i}} \not\equiv_n 1$. Then we know that $ord(g)$ is $n-1$ and not a divisor of $ord(\mathbb{Z}_n^*)$

Yet another example:
$F_1 : A \vee Z \rightarrow C$
$F_2 : B \rightarrow A$
$F_3 : E \rightarrow Z$
$F_4 : A \wedge C \rightarrow D \vee E$
$F_5 : D \wedge A \rightarrow Z$
$F_6 : A \wedge B$
$F_1 \ \wedge F_2 \wedge F_3 \wedge F_4 \wedge F_5 \wedge F_6 \models Z$
$\{F_1, ..., F_6\} \models Z$
Intuitive proof: F_6 is true, therefore A is true. Due to F_1 C is true. Then F_4 is true and either f_3 or F_5 is true, either way Z is true.
It can also be solved using propositional truth tables.
However this time we use a proof system: $\Pi = (\mathcal{S}, \mathcal{P}, \tau, \phi)$.
$R_1 : F \wedge G \vdash_{R1} F$
$R_2 : F \vdash_{R2} F \vee G$
$R_3 : \{F,\ F \rightarrow G\} \vdash_{R3} G$ (If i find both F and $F \rightarrow G$ in database, I can find G.)
$R_4 : \{F, G\} \vdash_{R4} F \wedge G$
$R_5 : \{F \wedge G \rightarrow H, G\} \vdash_{R5} F \rightarrow H$
$R_6 : \{F \vee G, F \rightarrow H, G \rightarrow H\} \vdash_{R6} H$
In this case we don't want redundant rules.

Now we generate new true statements in this system:
$F_6 \vdash_{R1} A(F_7)$
$F_7 \vdash_{R2} A \vee Z(F_8)$ (für $G = Z$)
$\{F_8, F_1\} \vdash_{R3} C(F_9)$
$\{F_7, F_9\} \vdash_{R4} A \wedge C(F_{10})$
$\{F_{10}, F_4\} \vdash_{R3} D \vee E$ (für $F = A \wedge C, G = D \vee E$)
$\{F_5, F_7\} \vdash_{R5} D \rightarrow Z(F_{12})$
$\{F_{11}, F_{12}, F_3\} \vdash_{R6} Z(F_{13})$

4.1.1 Terms of logic

Using babyexample of propositional logic.

Syntax The syntax of a logic defines an alphabet (of allowed symbols) and specifies which strings (over the alphabet) are (syntactically) well-formed formulas.

Atomic formulas *(GER: Atomare Formeln)* $A_i, (i = 1,2,3)$
$A_1 = \neg F, A_2 = (F \ \wedge G), A_3 = (F \vee G)$
Formulas with binary connectives usually are required to have brackets around them, in order to parse it easily.

Interpretation / Structure A formula can have an interpretation *(GER: Wahrheitsbelegung oder Struktur)*

Semantic The semantics of a logic is a (partial) function $\sigma(F, \mathcal{A}) \in \{0,1\}$ which takes a formula and an interpretation A. In other places this function is written as $\mathcal{A}(F)$.
For example: $\mathcal{A}((F \wedge G)) = 1$ if and only if $\mathcal{A}(F) = 1$ und $\mathcal{A}(G) = 1$
$A(\neg F) = 1$ if and only if $\mathcal{A}(F) = 0$

Syntax writing $(A \wedge \neg B) \wedge \neg(A \wedge C)$
Can also be written as a syntax tree or in the form (i)

6.6.1 Syntax of predicate logic

Variable symbol $x_i, i \in \mathbb{N}$ (usually x_0 is not used), usually $a, b, c, ...z$

Predicate symbol $P_i^{(k)}, i, k \in \mathbb{N}$, where k denotes the number of arguments of the predicate.

Term is defined inductively: A variable is a term, and if $t_1, ..., t_k$ are terms, then

Formula So a formula as a tree (not term) is a graph where the nodes are: $[\forall y]- > [\wedge]- > [P(x,y)]- > [Q(f(a,x))]$

Interpretation.

Model $\mathcal{A} \vDash F$.
A is an interpretation.
$A \vDash F$, F is true in A.
This can also be a set:
$A \vDash M = \{F_1, ..., F_k\}$ F_i is a formula.
Model assigns truth values to variable symbols.

Satisfiability A formula is erfüllbar, if there exists an interpretation A, such that A is true.

Tautology F is a tautology iff $\neg F$ is unsatisfiable. $A \vDash F$ is tautology
$\vDash F$ (is equivalent to the above statement)!

Logical implication ***(GER: Logische Folgerung)***
For each interpretation A, if F is true, then G is true.
$F \vDash G$
So $F \leq G$.

Logical equivalence ***(GER: Logische Äquivalenz)***
For each interpretation A, if F is true so is M. If F is false, M is false.
$F \equiv M$

Lemmas about logical equivalence
The 3 statements are equivalent (non-logical).

1. $\{F_1, ..., F_k\} \vDash G$. Die Menge ist dabei selber keine Formel.
2. $(F_1 \wedge ... \wedge F_k) \rightarrow G$ is a tautology.
3. $\{F_1, ..., F_k, \neq G\}$ is unsatisfiable.

A proof is only possible in words. The proof of this lemma can't be described in logic.
If $\mathcal{A}$ is an interpretation for $\{F_1, F_2, ..., F_k, G\}$

1. Bedeutet: Wenn A Modell ist für $\{F_1, ..., F_k\}$, dann auch Modell für G.
2. Bedeutet: Wenn A Modell ist für $F_1 \wedge F_2 \wedge ... \wedge F_k$, dann auch für G. Jedes Modell für $\{F_1, ..., F_k\}$ ist auch Modell für G.
3. Bedeutet: Wenn A Modell ist für $\{F_1, F_2, ..., F_k\}$, dann nicht für $\neg G$ (ist genügend).

Basic equivalences:
$\neg(F \wedge G) \equiv \neg F \vee \neg G$
$A(\neg(F \wedge G)) = 1 \Leftrightarrow A(\neg F \vee F \neg G)$ alternative:
$A((F \wedge G)) = 0 \Leftrightarrow A(F) = 0$ oder $A(G) = 0$
$\Leftrightarrow A(\neg F) = 1$ oder $A(\neg G) = 1$
$\Leftrightarrow A(\neg F \vee \neg F)$

CNF Form
CNF Form (conjunctive normal form) (written as a tree):
A literal is either an atomic formula or the negation of an atomic formula. CNF looks like: $\wedge(\vee(l_i, \neg l_j, l_k), \vee(l_q, l_i))$
$\wedge(\vee(A, \neg B, \neg A), \vee(B, \neg C, D), \vee(A, B, C, D, E, F))$

DNF Form
DNF Form (disjunctive normal form) (written as a tree):
Just like CNF, but with $\wedge$ exchanged with $\vee$.

Function table
Each formula can be written in at least one disjunctive and conjunctive normal form (which is not directly obvious).
Constructive proof, look at proof table. Then look at the first instance, where the equation is true.
$(A \ \wedge \neg B) \vee (B \wedge \neg C)$. Will be rewritten as:
It is true if $A = 0, B = 1, C = 0$
So the DNF can be generated from any truth table:
$F \equiv (\neg A \wedge B \wedge \neg C) \vee (A \wedge \neg B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$
The CNF form can be generated from all false statements:
$F \equiv (A \vee B \vee C) \wedge (A \vee B \vee \neg C) \wedge (A \vee \neg B \vee \neg C) \wedge (\neg A \vee \neg B \vee \neg C)$

The problem with this form, is that it can turn exponential in size!
The DNF/CNF needn't to be generated in this fashion. Any DNF/CNF conform equation is valid.

Common mistakes
Also brackets are missing?
$\forall x P(x) \vee \exists y P(x,y)$ is not a formula, hence $P(x)$ and $P(x,y)$ have a different amount of parameters.
$f(f(f(a)))$ is not a formula, it is a term. It is not evaluated.
$\forall x P(f(f(f(a))))$ is a formula, it doesn't matter whether x is in the formula after the quantifier.
$P(x,y) \vee P(y,z) \rightarrow P(x,z)$ is a formula ($\rightarrow$ is replaced with $(\neg F \vee G)$
Semantics of $P(a)$ may be equivalent to $\forall P(a)$
There are **bound** and **unbound** *(GER: freie)* variables.
The same variable representation can be bound and unbound at two different places.
Missing brackets.

A good example $(\forall x \exists y P(x, g(y, f(a))) \vee \neg Q(z)) \vee \neg \forall x R(x, y, a)$
Again in a syntax tree each leaf node is a term and the other terms are connectives.
$\vee[\vee[\forall x[\exists y[P(x, g(y, f(a)))]], \neg[Q(z)]], \neg[\forall x[R(x, y, a)]]]$

All variables in a quantifier is bounded.
So x, y on the left hand-side are bounded and x on the right-hand side.
z is unbounded and y on the right-hand side is bounded aswell.

Define an interpretation in predicate logic (Semantics) Evaluate the term i.

1. **Universe** First define the universe of discourse. The universe of this interpretation is sometimes defined as $U^{\mathcal{A}}$
2. **Define function symbols (in words)** $\phi(f) : U \mapsto U$. So ϕ is a function which interprets a function f, which has to go from f to g.
   $\phi(g) : U \times U \mapsto U$
3. **Define predicates** $\psi(P) : U \times U ... \times U \mapsto \{0, 1\}$, which in the interpretation $\mathcal{A}$ is written as $P^{\mathcal{A}}$
   This defines variables of bounded variables.

4. **Define unbounded variables** ***(GER: freie Variabeln)***

$\eta(y) : U \mapsto \{0, 1\}$
Probably these are written as $y^{\mathcal{A}}$

This is written as $\mathcal{A} = (U, \phi, \psi, \eta)$
Above example:
$\phi(f) : U \mapsto U$, so f in the interpretation $\mathcal{A}$ is written as: $f^{\mathcal{A}}$
$\phi(g) : U \times U \mapsto U$

Formal semantics

$\mathcal{A}(t)$ is recursively defined as follows:
If t is a variable, then $\mathcal{A}(t) = \eta(t)$

The $\mathcal{A}_{[x \to u]}$ is a slightly modified interpretation of $\mathcal{A}$, which changes η at position x: $\eta(x) = u$ for all u.

$\mathcal{A}(\forall x G) = \begin{cases} 1 & \text{if } \mathcal{A}_{[x \to u]}(G) = 1 \text{ for all } u \in U. \\ 0 & \text{sonst} \end{cases}$ $\mathcal{A}(\exists x G) = \begin{cases} 1 & \text{if } \mathcal{A}_{[x \to u]}(G) = 1 \text{ for some } u \in U. \\ 0 & \text{sonst} \end{cases}$

Example 6.20

$F = \forall x(P(x) \vee P(f(x, a)))$
$U^{\mathcal{A}} = \mathbb{N}$
$a^{\mathcal{A}} = 3$
$f^{\mathcal{A}}(r, s) = r + s$
$g^{\mathcal{A}}(r) = r + 1$ (not necessairy for the interpretation)
$P^{\mathcal{A}}(r) =$ 1, wenn r gerade, sonst 0 (Alternative $P^{=}$ Geradheit)
(GER: Die Struktur ist passend) From the above definitions follows (after computing an infinite amount of elements of the universe of discourse.)
$\mathcal{A}(F) = 1$

Forcing a model A which forces the universe to be larger or equal than 3

The size of a model = size of the universe.
When we talk about the size of A, we talk about the size of its defined unbound variables.
Now we are searching for a formula, for which each interpretation $|U^{\mathcal{A}}| \leq 3$
Couldn't one simply choose: $(P(a) \vee \neg P(a)) \wedge (Q(b) \vee \neg Q(b)) \wedge (R(c) \vee \neg R(c))$
This is one way to do it:
$F = \forall x P(x, x) \wedge \exists x \exists y \exists z(\neg P(x, y) \wedge \neg P(x, z) \wedge \neg P(y, z))$

Forcing a model A which forces the universe to be infinite

$G = \forall x P(x, f(x)) \wedge \forall x \neg P(x, x) \wedge \forall x \forall y \forall z(P(x, y) \;\wedge P(y, z) \to P(x, z))$
The formula G has no finite model.
Take for example the interpretation f successor function.
$u_0 = u$
$u_1 = f(u)$
$u_i = f(u_{i-1})$ for $i = \{1, 2, 3, ..., N\}$
Falls U^A endlich $\Rightarrow$ es gibt $r, s, r \neq s, u_r = u_s$
Dies ist im Widerspruch: $P(u, u_{i+1}) = 1$ und somit $P(u_i, u_j) = 1$ (for all $i, ji \neq j$)
$\Rightarrow P(u_r, u_s) = 1$ but also $P(u_r, u_s) = 0$
$\Rightarrow U^A$ is infinite.
Other example:
This forces every model to be an equivalence relation.
$P(x, x) \wedge (P(x, y) \leftrightarrow P(y, x)) \wedge (P(x, y) \wedge P(y, z) \to P(x, z))$
If you don't want to force the universe to have 3 variables:
Using universal quantifiers every equivalence relation in a modell is true iff:
$\forall x \forall y \forall z P(x, x) \wedge (P(x, y) \leftrightarrow P(y, x)) \wedge (P(x, y) \wedge P(y, z) \to P(x, z))$

$\vee(\vee(\ \forall x(\exists y(P(x, g(y, f(a)))))), \neg(Q(z))), \neg(\forall x(R(x, y, a)))$
A model must define: $P^{\mathcal{A}}, Q^{\mathcal{A}}, f^{\mathcal{A}}, a^{\mathcal{A}}, g^{\mathcal{A}}, y^{\mathcal{A}}, z^{\mathcal{A}}$ in order to be *(GER: passend).*

Theorem 6.6.10

$\neg \exists x(\forall y(P(y, x) \leftrightarrow \neg P(y, y)))$ is a tautology (so it is true for every interpretation).

Prove of lemma 6.6.2

$\neg \exists x F \equiv \forall x \neg F$
$\mathcal{A}$ is a model, which is *(GER: passend).*
$\mathcal{A}(\neg \exists x F) = 1 \Leftrightarrow$ Es gibt kein $u \in U$ mit $\mathcal{A}_{[x \to u]}(F) = 1$
$\Leftrightarrow$ Für alle u gilt $\mathcal{A}_{[x \to u]}(F) = 0$
$\Leftrightarrow$ Für alle u gilt $\mathcal{A}_{[x \to u]}(\neg F) = 1$
$\Leftrightarrow \mathcal{A}(\forall x \neg F) = 1$

Lemma

Sei F eine Formel mit x frei.
F ist Tautologie (gültig) $\Leftrightarrow F[x/a]$ ist Tautologie (wo a eine Konstante ist).
$\Leftrightarrow \forall x F$ ist Tautologie.

Group axioms in predicate logic (Wonderful example)

$f(x, y) : x \circ y$
$g(x) : x^{-1}$
The equal sign should be a predicate.
$G_1 : \forall x f(x, e) = x$
$G_2 : \forall x \forall y \forall z f(x, f(y, z)) = f(f(x, y), z)$
$G_3 : \forall x f(x, g(x)) = a$

GA = Gruppenaxiome.
$GA = \{G_1, G_2, G_3\}$
$GA \vDash F$
A model of GA (formulas) are groups.
$GA \vDash \forall x f(e, x) = x$
$GA \vDash \forall x f(g(x), x) = a$

Peano axioms in predicate logic

$\mathbb{N}$ is a model of the 5 axioms. S would be the successor function.
$s(x)$: Sucessor function in model.
$\mathbb{N} \vDash PA \;\vDash F \Rightarrow \mathbb{N} \vDash F$
Therefore every model, which fullfils the 5 axioms (this could be the modell $\mathbb{R}$), F must hold.

Fun Lemma:
$(\exists x P(x)) \to Q(y) \equiv \forall x(P(x) \to Q(y))$
$(\exists x P(x) \to Q(x)) \equiv (\neg \exists x P(x) \vee Q(x)) \equiv (\forall x \neg P(x) \vee Q(x)) \equiv \forall x(\neg P(x) \vee Q(x)) \equiv \forall x(P(x) \to Q(x))$

Prenex normal form

Achtung beim hochblubbern eines Quantors dürfen keine freie Variabeln mit selben Namen vorkommen.
Also einfach gebundene Variabeln umbennenen (wenn keine anderen Vorkommen bezeichnet mit $F[y/a]$)
In the syntax tree we write next to the quantifier $[y/u]$

Lemma: $\exists xG \equiv \exists yG[x/y]$
Bereinigte Form = Keine gebundene Variabeln mit gleichem Namen.

$\forall x(\exists u(\exists v(\vee(P(x, g(u, f(x))), \neg(Q(z)), \neg(R(v, y, a))))))$ ($\exists v$ could go in front of $\forall x$ in this special configuration.)

Note about cool theorem

F is a theorem $\Leftrightarrow$ F is a tautology. (under maybe some axioms).
$\models \neg\exists x\forall y(P(y, x) \leftrightarrow \neg P(y, y))$
Concretization:
Corollary: Russels paradox.
Proof: $U :\Leftrightarrow$ all sets.
$P(x, y) = 1 :\Leftrightarrow x \in y$
$\neg\exists R\forall S(S \in R \leftrightarrow S \not\in S)$
Another corollary: $\{0, 1\}^\infty$ ist überabzählbar.
Proof: $U :\Leftrightarrow \mathbb{N}$ (Fixe Aufzählung Elementen von $\{0, 1\}^\infty$).
$P(x, y) = 1 :\Leftrightarrow y$-te Bit der x-ten Folge ist 1.
Another corollary: Es gibt nicht berechenbare Funktionen. $\mathbb{N} \to \{0, 1\}$
Proof: $U :\Leftrightarrow \mathbb{N}$
$P(y, x) :\Leftrightarrow$ Output von Programm mit Index x, für den Input y. (falls nichts rauskommt oder 0).
$\neg\exists x\forall y(P(y, x) \leftrightarrow \neg P(y, y))$
$y \mapsto \neg P(y, y)$ A concrete function, which is not computable.

Proving the cool theorem:
$\neg\exists\forall y(P(y, x) \leftrightarrow \neg P(y, y)) \equiv \forall\neg\forall y(P(y, x) \leftrightarrow \neg P(y, y)) \equiv \forall\exists y\neg(P(y, x) \leftrightarrow \neg P(y, y)) \equiv \forall\exists y(P(y, x) \leftrightarrow P(y, y))$
Proof that tautology:
$\mathcal{A}$ a *(GER: passende)* Struktur $(U^\mathcal{A}, p^\mathcal{A})$
$\mathcal{A}_{[x\to u]}(\exists yP(y, x) \leftrightarrow P(y, y))$ für alle $u \Rightarrow \mathcal{A}(\forall x, ...) = 1$ (tautology)
$\mathcal{A}_{[x\to u][y\to v]}(P(y, x) \leftrightarrow P(y, y)) = 1$ für ein v.
$v = u$
Hence $P(u, u) = P(u, u)$ (u don't say)

Techniques for proving tautologies

$\forall x\exists y(P(x, f(x)) \vee \neg P(x, y))$
This is a tautology when $y = f(x)$, because $f(x), y \in U$.
$\forall x\exists y(P(y, x) \leftrightarrow P(y, y))$
Same argument as above. $y = x$
Couldn't I just use existential elimination as example.

Hilbert calculus

A *derivation rule* is a rule for deriving a formula from formulas (called the precondition), written as:
Calculus: $\{R_1, ..., R_m\}$.
$\{F_1, .., F_k\} \vdash_R G$ G can be derived from $\{F_1, ..., F_k\}$ using rules R.
Example calculus:
$R = \{$
$\{F, G\} \vdash_{R1} F \wedge G$
$\{F\} \vdash_{R2} F \vee \neg F(TND)\}$
$F \vdash_{R3} \neg\neg F$
$\{F \vee G, \neg G\} \vdash_{R4} F$
$\{F \to g, G \to F\}$ }

The law of the excluded middle is also shortened as TND (tertium non datur).
Applying rules on set:
$M = \{D \wedge \neg(A \wedge C), A \vee C, E \to A\}$
Rule R_1 with $F = A \vee C$ and $G = E \ \to A$ will yield: $(A \vee C) \wedge (E \to A)$

Often such derivations are written in a gentzen like style:
$\frac{F_1 F_2 ... F_k}{G}$

Correctness:
$M \vdash_R F \Rightarrow M \models F$
Completeness:
$M \vdash_R F \Leftarrow M \models F$
The rules R1,...,R5 are not complete, since we can't show: $A \vee B \models B \vee A$

Resolution calculus

First the formula has to be brought into conjuctive normal form (one way of doing this is by computing truth tables, but that takes forever).
$(A \vee \neg B \vee C) \wedge (\neg A \vee C \vee D) \wedge (\neg A \vee \neg D)$
$MK = \{\{A, \neg B, C\}, \{\neg A, C, D\}, \{\neg A, \neg D\}\}$
Such a list of formulas is called a *(GER: Klausel)* (the comma is now a $\vee$!)
MK is a *(GER: Klauselmenge).*
$\{A, \neg B, C\}, \{\neg A, C, D\}$
Now I can resolve the above two clauses $\{\neg B, C, D\}$
This is not yet complete.
Such a resolution step is done in the following way:
$K_1 = \{A, C, E, ...Z, L\}$ und $K_2 = \{A, B, C, ..., Z, \neg L\}$
K_1 resoluted with K_2 is $\{K_1 \setminus L, K_2 \setminus \neg L\}$.
If at the end of this process we get $\{\}$, we get an unsatisfiable formula.
In order to prove a tautology, we can just show F is tautology $\Leftrightarrow \neg F$ is unsatisfiable.
$\{F_1, ..., F_k\} \models G \Leftrightarrow \{F_1, ..., F_k, \neg G\}$ unsatisfiable.

Example:
$\{\{A, B\}, \{\neg A, \neg B\}\}$
(Is satisfiable $A = 1, B = 0$)
You are not allowed to remove 2 at once!
$\{B, \neg B\}$

Example:
$(A \vee Z) \to C \equiv \neg(A \vee Z) \vee C \equiv (\neg A \wedge \neg Z) \vee C \equiv (\neg A \vee C) \wedge (\neg Z \vee C)$, $KM : \{\{\neg A, C\}, \{\neg Z, C\}\}$
$B \to A \equiv \neg B \vee A$, $\mathcal{K}_1 = \{\neg B, A\}$
$E \to Z \equiv \neg E \vee Z$, $\mathcal{K}_2 = \{\neg E, Z\}$

$A \wedge C \rightarrow D \vee R \equiv \neg(A \wedge C) \vee D \vee E \equiv \neg A \vee \neg C \vee D \vee E$, $\mathcal{K}_3 = \{\neg A, \neg C, D, E\}$

$D \wedge A \rightarrow Z$, $\mathcal{K}_4 = \{\neg A, \neg D, Z\}$ $A \wedge B \equiv$, $\mathcal{K}_5 : \{A\}, \{B\}$

Wir wollen zeigen, dass Z immer gilt, also muss $\neg Z$ unerfüllbar sein (unter Voraussetzung das die Regeln stimmen).

In order to prove this is a tautology: $M \vDash F \Leftrightarrow M \vee \{\neg F\}$ unsatisfiable.

$\{\{\neg A, C\}, \{\neg Z, C\}, \{\neg B, A\}, \{\neg E, Z\}, \{\neg A, \neg C, D, E\}, \{\neg A, \neg D, Z\}, \{A\}, \{B\}, \{\neg Z\}\}$

1. $\{\neg E, Z\}, \{\neg A, \neg C, D, E\} \Rightarrow \{\neg A, \neg C, D, Z\}$
2. $\{\neg A, \neg C, D, Z\}, \{\neg A, \neg D, Z\} \Rightarrow \{\neg A, \neg C, Z\}$
3. $\{\neg A, \neg C, Z\}, \{A\} \Rightarrow \{\neg C, Z\}$
4. $\{\neg A, C\}, \{A\} \Rightarrow \{C\}$
5. $\{C\}, \{\neg C, Z\} \Rightarrow \{Z\}$
6. $\{Z\}, \{\neg Z\} \Rightarrow \varnothing$

Also unerfüllbar.

Lemma. $\mathcal{K} \vdash_{Res} K \Rightarrow \mathcal{K} \vDash K$

Proof correctness: We prove that the rules res are correct.

It suffices, because $\{K_1, K_2\} \vdash_{res} K \Rightarrow \{K_1, K_2\} \vDash K$

then also $\{K_1, K_2\} \vdash_{res} \mathcal{K}(\subseteq \{K_1, K_2\}) \vDash K$

Let $\mathcal{A}$ be any interpretation which is *(GER: passend)* for $K_1, K_2(undK)$.

$\mathcal{A} \vDash \{K_1, K_2\}$

$K_1 = \{\ldots, \ldots, L\}$

$K_2 = \{\ldots, \ldots, \neg L\}$

If $\mathcal{A}$ is a model of $\{L\}$, $\mathcal{A}(L) = 1$, then $A \vDash K_2 \setminus \{\neg L\}$

If not $A \vDash (K_1 \setminus \{L\})$.

$\Rightarrow \mathcal{A} \vDash (K_1 \setminus \{L\}), \vee (K_2 \setminus \{\neg L\})$

Proof completeness of unsatisfiability:

Theorem: $\mathcal{K}$ unsatisfiable $\Leftrightarrow \mathcal{K} \vdash_{res} \varnothing$

Not every statement can be shown by reduction calculus, for example the valid formula:

$\{A\}, \{B\} \vDash \{A, B\}$, but not $\{A\}, \{B\} \cancel{\vdash_{res}} \{A, B\}$

$A \wedge B \Rightarrow A \vee B$

Proof using induction:

Base case: $n = 1 : \mathcal{K} \vDash \varnothing \Leftrightarrow \{A_i\} \in \mathcal{K}$ und $\{\neg A_i\} \in \mathcal{K}$

Induktionshypothese:

For any klauselmenge $\mathcal{K}$ mit F gilt $A_1, ..., A_n$ gilt:

$\mathcal{K}$ unsatisfiable $\Rightarrow \mathcal{K} \vdash_{res} \varnothing$

Induktionsschritt: $\mathcal{K}$ enthalte $A_1, ..., A_{n+1}$ und ist unerfüllbar.

$\mathcal{K} = \{\{...\}, \{..., A_{n+1}\}, \{...\}, \{...., \neg A_{n+1}\}, \{...., A_{n+1}\}, ...\}$

Wenn $A_{n+1} = 0$, dann führt es zu $\mathcal{K}_0$ (remove all clauses from A_{n+1}, which contain $\neg A_{n+1}$, hence they are valid one hundret percent and cannot contribute to unsatisfiability. We can remove the term A_{n+1} from every clausel, hence it is always wrong and can therefore be removed.).

Wenn $A_{n+1} = 1$, dann führt es zu $\mathcal{K}_1$.

$\mathcal{K}$ ist unerfüllbar mit $A_{n+1} = 0 \Leftrightarrow K_0$ unerfüllbar $\Rightarrow \mathcal{K}_0 \vdash_{res} \varnothing$

$\mathcal{K}$ ist unerfüllbar mit $A_{n+1} = 0 \Leftrightarrow K_1$ unerfüllbar $\Rightarrow \mathcal{K}_1 \vdash_{res} \varnothing$

This can be used constructively from the bottom up, however there are exponential possibilities.

We can solve this using recursion (or bottom-up) exponentially, but we can verify it in polynomial time.

4.1.2 Natural type deduction (Sequent style calculus)

Syntactic objects: (M, F)

$(M, F) \vdash_K (M', F') \Rightarrow (M \vDash F \Rightarrow M' \vDash F')$.

However it is not as beautiful, hence you have to store everything.

5 Naive set theory

Naive set theory was originally invented by Georg Cantor.

Set *(GER: Menge)* A set is atomically defined as a collection of things. These things are called **elements** or **members** of the set. Sets are usually noted as *upper-case letters* such as A, P and so on. Sets do not have an order and must contain each element at most once. The sizes of sets can be infinite.
Naive set theory claims that every definable set exists.
According to naive set theory even sets which contain themselves are allowed, although they lead to contradictions.
Russels paradox:
The set of all sets that are not members of themselves:
$R = \{A|A \notin A\}, R \in R \Rightarrow R \notin R, R \notin R \Rightarrow R \in R$
The existence of R therefore leads to a contradiction, which disproofs the axiom of naive set theory, that every definable set exists.
However not all self containing sets lead to a contradiction, such as the sets of all sets which have at least 7 elements: $R = \{A|\ |A| \geq 7\}$.

Notations The *roster notation* explicitly lists out every element of the set seperated by commas and surrounded by brackets.
$A = \{2, 4, 6, 8\}$
The roster notation gets inaccurate with sets of infinite size:
$A = \{1, 2, 3, 4, ...\}$

The *set-builder notation* is of the form:
$A = \{x \mid \Phi(x)\}$ where $\Phi(x)$ is a predicate on the dummy variable x which can be evaluated as either true or false. x is an element of A iff the predicate $\Phi(x)$ is true. The *vertical bar* $|$ can be translated to the word 'where'. The set A of x, where x is an even number. Alternatively the *column* : is used (although it is also a place-holder without mathematical significance).
Set-builder notation can be defined non-atomically by using second-order quantification over the set to be defined: $\exists A \forall x (x \in A \Leftrightarrow \Phi(x))$
$\Phi(x)$ might use other (global) sets as predicates so it can use quantification statements:
$\forall t_1..\forall t_k \exists A \forall x (x \in A \Leftrightarrow \Phi(x))$
The above equation is the axiom of unrestricted comprehension, however when you take $\Phi(x) = \neg(x \in x)$ you have a contradiction. Therefore the law is inconsistent.
It is also common to write $\{x \in A | \Phi(x)\}$, which is equivalent to stating:
$\{x | x \in A \wedge \Phi(x)\}$

Indexing set If every element of the set is a number and has a smallest element, elements in the set can be eummerated (and therefore receive an index). The index set $\mathcal{I}$ is usually the natural numbers and the set to be enumerated is A. $x_i | x \in A\}$

Empty set or *null set* is the set without members and is typically represented as $\{\}$ or $\varnothing$ (LaTeX: \varnothing).

Class A class is a collection of sets (depending on the theory these are just sets of sets or are atomically defined objects). They are usually noted as upper-case curvy letters such as $\mathcal{F}$ (LaTeX: $\mathcal{A-Z}$)

Tuple Tuples are an ordered list of elements (see ordered set). Their elements are noted within brackets, seperated by commas: $t = (e_1, e_2, ..., e_n)$. In geometry they can be used for coordinates.
Contrary to sets, tuples can have multiple elements of the same kind and the order matters.

Size of tuple	Name	Shorthand
1	Singleton	1-tuple
2	Ordered pair	2-tuple
3	Ordered triple	3-tuple
4	Ordered quadruple	4-tuple

Ordered set An *ordered set* is also referred to as a *sequence* and is discussed in another sectios. They can be represented as sets by using an ordered pair (2-tuple) where the first part is a number (the *index*) and the other part is the actual element. Note that this definition also allows for multisets and is therefore identical to the definition of a sequence.

Multiset Multisets are collections that can contain an element more than once. They can be represented as sets by using an ordered pair (2-tuple) where the first part is the element and the second part is the count (a natural number). No two ordered pairs may however exist with the same first part.

5.1 Set definitions

Name	Notation	Set theoretic interpretation
Equality	$A = B$	$\forall x (x \in A \leftrightarrow x \in B)$, $(A \subseteq B) \wedge (B \subseteq A)$ (important for proofs)
Inequality	$A \neq B$	$\exists x \neg (x \in A \leftrightarrow x \in B)$
Subset	$A \subseteq B$	$\forall x (x \ \in A \rightarrow x \in B)$
Subset exclusive	$A \subsetneq B$	$\forall x (x \in A \rightarrow x \in B \wedge \neg x \in A \leftrightarrow x \in B)$
Union	$A \cup B$	$\{x | x \in A \vee x \in B\}$
Intersection	$A \cap B$	$\{x | x \in A \wedge x \in B\}$
Big Union	$\bigcup \mathcal{Z}$	$\{x | \exists A \in \mathcal{Z} (x \in A)\}$ (where $\mathcal{Z}$ is a class)
Big intersection	$\bigcap \mathcal{Z}$	$\{x | \forall A \in \mathcal{Z} (x \in A)\}$ (where $\mathcal{Z}$ is a class)
Difference	$A - B, A \setminus B$	$\{x \in A | x \notin B\}$
Symmetric difference	$A \triangle B, A \oplus B$ ($\oplus$ is xor)	$\{x \in A \text{ XOR } x \in B\}$, $A - B \wedge B - A$
Power set	$\mathcal{P}(A)$	$\{S | S \subseteq A\}$
Complement	$\overline{A}$ (sometimes A^c)	$\{x \in U | x \notin A\}$ (where $U =$ universe of discourse)
Ordered pair	(a, b)	$\{\{a\}, \{a, b\}\}$
Cartesian product	$A \times B$	$\{(a, b) | a \in A \wedge b \in B\}$
N-ary product	$\bigtimes_{i=1}^{k} A_i$	$\{(a_1, ..., a_k) | a_i \in A_i$ for $1 \leq i \leq k\ \}$
Cartesian exponentiation	A^n	$\bigtimes_{i=1}^{n} A$
		Note: The cartesian product is neither commutative nor associative.
		The n-ary product defines an n-tuple, not pairs of pairs.

5.2 Proving style

Proving $a \in \{x | \Phi(x)\}$, by showing $P(a)$ is true.
Proving $a \in \{x \in \mathbb{E} | \Phi(x)\}$, by showing $a \in \mathbb{E}$ and showing $\Phi(a)$
Proving $A \subseteq B$ directly:
Suppose $a \in A \Rightarrow \ldots \Rightarrow a \in B$. Thus $a \in A$ implies $a \in B$, so it follows $A \subseteq B$.
Proving $A \subseteq B$ by modens tollens:
Suppose $a \notin B \Rightarrow \ldots \Rightarrow a \notin B$. Thus $a \notin B$ implies $a \notin B$, so it follows (by modens tollens) that $A \subseteq B$.
Proving $A = B$ by proving $A \subseteq B$ and proving $B \subseteq A$. Since $A \subseteq B$ and $B \subseteq A$, it follows that $A = B$.

5.3 Properties

Useful properties of these relations:

Name	Rule 1	Rule 2
Idempotence	$A \cap A = A$	$A \cup A = A$
Commutativity	$A \cap B = B \cap A$	$A \cup B = B \cup A$
Associativity	$A \cap (B \cap C) = (A \cap B) \cap C$	$A \cup (B \cup C) = (A \cup B) \cup C$
Absoption	$A \cap (A \cup B) = A$	$A \cup (A \cap B) = A$
Distributivity	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
Complementarity	$A \cap \overline{A} = \varnothing$	$A \ \cup \overline{A} = \mathcal{U}$
De Morgan's law	$\overline{A \cap B} = \overline{A} \cup \overline{B}$	$\overline{A \cup B} = \overline{A} \cap \overline{B}$
Consistency	$A \subseteq B \Leftrightarrow A \cap B = A \Leftrightarrow A \cup B = B$	
Power set laws	$a \in \mathcal{P}(A) \Leftrightarrow a \subseteq A$	$A \in \mathcal{P}(A)$
Subset law	$\{a\} \subseteq A \Leftrightarrow a \in A$	

5.4 Testing knowledge

$\{\{\varnothing\}\} = \{\{\varnothing\}, \{\varnothing, \varnothing\}\} \subseteq \{\varnothing, \{\varnothing\}\}$
$\{\} \in \mathcal{P}(A)$
$\forall A \{\} \in A$
$\{\} \times A = \{\}$

5.5 Pitfalls I had

A need not to be a subset of $\mathcal{P}(A)$.
It's rather the exception, for example: $\{\varnothing\} \subseteq \{\varnothing, \{\varnothing\}\}$ $A \not\subseteq \mathcal{P}(A)$

5.6 Relations

A relation is a subset of the n-ary cartesian product: $\rho \subseteq \bigtimes_{i=1}^{k} A_i$. For binary relations $\rho \subseteq A \times B$.
There are $2^{|A|^2}$ binary relations ($\subseteq A \times A$) on a finite set A.

Elements of binary relations

$a\rho b :\Leftrightarrow (a,b) \in \rho, \qquad a \in A, b \in B, \rho \subseteq A \times B$
$a \not\rho b :\Leftrightarrow \neg((a,b) \in \rho), \qquad a \in A, b \in B, \rho \subseteq A \times B$
The set A is called the *domain* and the set B the *codomain* of the binary relation ρ.

Matrix representation

A binary relation ρ can be easily represented as a Boolean $|A| \times |B|$ matrix M^ρ, where each tuple of the relation set is mapped with either a 0 or 1.

$$M^\rho_{a,b} = \begin{cases} 1 & \text{if } a\rho b \\ 0 & \text{otherwise} \end{cases} \qquad a \in A, b \in B$$

Identity relation

$id_A = \{(a,a) | a \in A\}$ (the set is commonly left out id).
$id = \widehat{id}$

Complete/Empty relation

The empty relation is $\varnothing$. No ordered tuple is in the relation.
The complete binary relation is $A \times B$. Every ordered pair is in the relation.

Inverse relation

$a\rho b \Leftrightarrow b\widehat{\rho}a \Leftrightarrow b\rho^{-1}a$

Composition of relations

$a\rho \circ \sigma c :\Leftrightarrow \exists b \in B(a\rho b \wedge b\sigma c), \quad \rho \subseteq X \times Y, \sigma \subseteq Y \times Z.$
The $\circ$ operator is sometimes left out $a\rho\sigma c$.
The composition of relations is associative! $\rho \circ (\sigma \circ \tau) = (\rho \circ \sigma) \circ \tau$

Exponating relations The n-fold composition of a relation ρ on a set A is denoted:

$\rho^n = \underbrace{\rho \circ \rho \circ ... \circ \rho}_{\text{n-times}}, \qquad \rho \subseteq A \times A.$
$\rho^n \subseteq \rho$
ρ is transitive $\Leftrightarrow \rho^2 \subseteq \rho$.

Transitive closure The relation of each possible path of a transitive relation ρ from a_i to a_j.

$p^* = \bigcup_{n=1}^{\infty} \rho^n$

Special relations of type $\rho \subseteq \mathbb{Z} \times \mathbb{Z}$.

Property:

Smaller than	$a < b \Leftrightarrow (a,b) \in <$	
Smaller or equal than	$a \leq b \Leftrightarrow (a,b) \in \leq$	
Greater than	$a > b \Leftrightarrow (a,b) \in >$	
Greater or equal than	$a \geq b \Leftrightarrow (a,b) \in \geq$	
Equality	$a = b \Leftrightarrow (a,b) \in =$	$id_{\mathbb{Z}}$
Inequality	$a \neq b \Leftrightarrow (a,b) \in \neq$	
Divides	$a \vert b \Leftrightarrow (a,b) \in \vert$	$\exists k \in \mathbb{Z}(a/b = k)$
Does not divide	$a \nmid b \Leftrightarrow (a,b) \in \nmid$	$\forall k \in \mathbb{Z}(a/b \neq k)$
Congruence	$a \equiv_m b$ (a and b have the same reminder when divided by m)	$\exists k(a - b = km)$

The congruence relation $a \equiv_m b$ is equivalent of stating that a and b have the same reminder when divided by m.

Properties: $\leq \cup \geq \Leftrightarrow$ complete relation.
$\leq \cap \geq \Leftrightarrow id$
$\equiv_m \cap \equiv_n \Leftrightarrow \equiv_{lcm(m,n)}$

5.6.1 Special properties of relations

The matrix representing the relations between nodes, can be thought of as an adjacency matrix of a directed graph. The properties of relations might be easier to think about in terms of digraphs.

reflexive

$\forall a(a\rho a), \quad a \in A, \rho \subseteq A \times A$
Alternate definition: $id_A \subseteq \rho$
Matrix definition: $diag(M) = 1, M_{a,a} = 1$
In order for a relation to reflexive graphically, each node has to form a directed loop with itself.
Examples: $\leq, \geq, | \subseteq \mathbb{Z}^2$

irreflexive
$\forall a(a \not\rho a), \quad a \in A, \rho \subseteq A \times A$
Alternate defintion: $\rho \cap id = \varnothing$
Matrix definition: $diag(M) = 0, M_{a,a} = 0$
In order for a relation to be irreflexive graphically, no node may have a directed loop with itself.
If ρ is irreflexive, ρ is not reflexive.

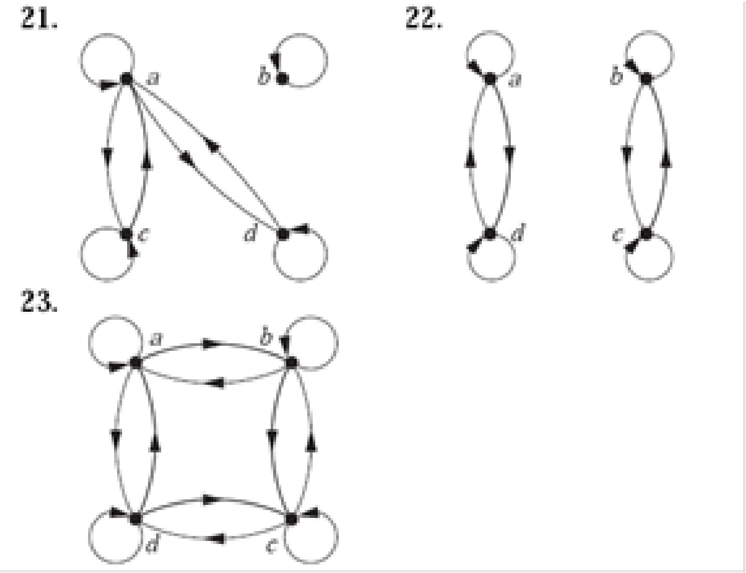
symmetrical
$\forall a \forall b(a\rho b \leftrightarrow b\rho a), \quad a \in A, b \in A, \rho \subseteq A \times A$
Alternate definition: $\rho = \widehat{\rho}$
Matrix definition of symmetric: $\forall a \forall b M_{a,b} = M_{b,a}$.
In order for a relation to be symmetric graphically: If theres a path from a to b, there must be a path from b to a.
Examples: $\equiv_m$, $=$, $\neq$, married to, relative of

antisymmetrical
$\forall a \forall b(a\rho b \wedge b\rho a) \rightarrow a = b, \quad a \in A, b \in A, \rho \subseteq A \times A$
Alternate definition: $\rho \cap \widehat{\rho} \subseteq id$
Matrix definition of antisymmetric: $\forall a \forall b(a \neq b) \ \rightarrow M_{a,b} \neq M_{b,a}$
In order for a relation to be assymetric graphically, there must be no strongly connected components of size greater than 1. If there is a path from a to b, there may not be a path from b to a ($a \neq b$). Loops such as $a\rho a$ are allowed.
Examples: $\leq$, $\geq$, $<$, $>$, $| \subseteq \mathbb{N}^2$, BUT NOT $| \subseteq \mathbb{Z}^2$ due to (-a — a).

transitive
$\forall a \forall b(a\rho b \wedge b\rho c) \rightarrow a\rho c, \quad a \in A, b \in A, c \in A, \rho \subseteq A \times A$
Alternate definition: $\rho^2 \subseteq \rho$
In order for a relation to be transitive graphically: If there is some indirect path from a to b, there must also be a direct path from a to b.
Examples: $\leq, \geq, <, >, |, \equiv_m$, ancestor of.

5.7 Equivalence relations

An equivalence relation is a relation, which is *reflexive, symmetrical* and *transitive.*
Commonly equivalence relations are represented as $\sim$. Example: $\equiv_m \subseteq \mathbb{Z}^2$ is an equivalence relation. Graphically they all look like this:



Equivalence class For an equivalence relation $\theta \subseteq A \times A$ and an element $a \in A$, the *equivalence class* is denoted as:
$[a]_\theta := \{b \in A | b\theta a\}$
As you can see beautifully from the above picture, a graph theoretic interpretation would be, that every connected component of a relation, is a complete digraph (with loops). Each element of a component is equivalent to the other. Example: $\equiv_m$ is an equivalence class $\forall m$.
$[0]_{\equiv_3} = \{..., -6, -3, 0, 3, 6, ...\}$
$[1]_{\equiv_3} = \{..., -5, -2, 1, 4, 7, ...\}$
$[2]_{\equiv_3} = \{..., -4, -1, 2, 5, 8, ...\}$
For the congruence relation the smallest non-negative number is the principal representative, which is the remainder after the division by m.

Quotient set A quotient set is uniquely defined by a set A and an equivalence relation $\theta \subseteq A^2$. (The slash below is not a setminus)
$A/\theta = \{[a]_\theta | a \in A\}$
It is also referred to as A *modulo* θ, A mod θ.
In other words, the collection of all equivalence classes forms exactly the quotient set, which is also a partition of A (see proof section).

Partition S is a partition (class of disjoint subsets, which unified result in A) of A and S_i is the i-th subset. By definition: $S_i \cap S_j = \varnothing (i \neq j)$
Also the union of all these subsets form S again. $\bigcup_{i \in I} S_i = A$. A partition is not only defined for equivalence classes, but is simply a set of sets, which share no element and are therefore mutually disjoint. Equivalence classes just show this off beautifully.

Rational numbers Relation of rational numbers $(a, b) \sim (c, d) :\Leftrightarrow a * d = b * c$
$\sim \subseteq (\mathbb{Z} \times (\mathbb{Z} - 0))^2$
Is reflexive: $a * b = b * a$
Is symmetric: $a * d = b * c \Rightarrow c * b = d * a$
Is transitive: $(a, b) \sim (c, d) \wedge (c, d) \sim (e, f) \Rightarrow a * d = b * c \wedge c * f = d * e \Rightarrow a * d * c * f = b * c * d * e \Rightarrow a * c * f = b * c * e \wedge d \neq 0$ which must be true, due to it being in $\mathbb{Z} - 0 \Rightarrow a * c * f = b * c * e \Rightarrow$
If $c \neq 0$ we can cancel c : $a * f = b * e$.
If $c = 0$, then $a = 0$ (due to $a * d = b * 0$ and $d \neq 0$). Similairly $e = 0$ and thus $a * f = b * e$.
Therefore the $\sim$ relation is transitive. 2/4 and 1/2 form the same equivalence class $[(1, 2)]_\sim = [(2, 4)]_\sim$. Therefore $\mathbb{Q}$ is the quotient set of the relation $\sim$: $\mathbb{Q} := \mathbb{Z} \times (\mathbb{Z} - 0)/\sim$.

5.8 Partial order relations

A partial order are relations which are reflexive, antisymmetric and transitive (Sometimes the word quasi order is used for relations which are reflexive and transitive).
A partial order set (shorthand: *poset*) is denoted as $(A; \preceq), \quad \preceq \subseteq A^2$
Graphically it is a DAG with loops on every node and no cycles.
Examples:

$\leq, \geq$ are partial orders (and total orders) on $\mathbb{R}$ noted as $(\mathbb{R}; \leq)$.

$| \subseteq \mathbb{N}^2$ The division relation $|$ is a partial order (but no total order) only in $\mathbb{N}$ (counter-example for $\mathbb{Z}$, $-2|2 \wedge 2| - 2$, which is against the antisymmetric property).

$(\mathcal{P}(S); \subseteq)$ has a partial order (and no total order if $|A| \geq 2$) and its Hasse diagram has each subset of different sizes on unique levels. Fun fact: The power set of a set of cardinality 3 looks like a cube, see picture.

Polygon cutting Given 2-d paper polygons, $a \preceq b$ if b can be transformed into a by using only straight cuts and transformations.
It is partially ordered, but not totally ordered, hence there is are triangles, that cannot be made into squares with 3/4 edge length and vice-versa.

Comparable Two elements $a \in A, b \in A$ where $a\rho b \vee b\rho a$ are called *comparable.* An example of incomparability would be the subset relation of the power set $\mathcal{P}(\{a,b,c\})$ and the elements $\{a\},\{b,c\}$. Graphically there is no path from a to b and no path from b to a.

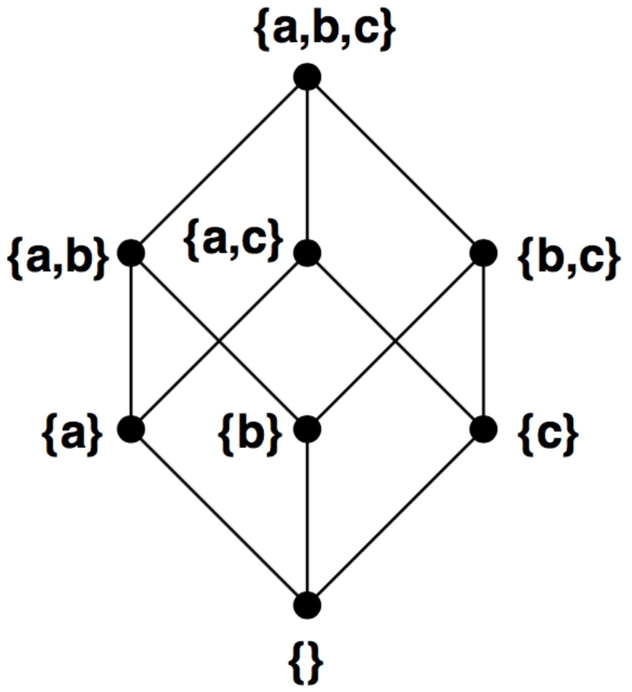
Total order If any two elements of a poset $(A;\preceq)$ are comparable, then it is called *totally ordered (linearly ordered).*

The relation $\prec$

$a \prec b :\Leftrightarrow a \preceq b \wedge a \neq b$
Proof if $a \prec b$ is transitive:
$a \preceq b$ is transitive: $a \prec b \wedge b \prec c \rightarrow a \prec c$
$\Rightarrow (a \overset{1}{\preceq} b \wedge a \overset{2}{\neq} b \wedge b \overset{3}{\preceq} c \wedge b \overset{4}{\neq} c) \rightarrow a \overset{5}{\preceq} c \wedge a \overset{6}{\neq} c$
We can show that 5 is proven by 1,3 by using transitivity of $\preceq$.
Wrong proof for 6 using transitivity: $(a \neq b \wedge b \neq c \not\Rightarrow a \neq c)$
Wrong proof for 6: $\prec \subseteq \preceq$ (true statement, but doesn't prove $a \neq c$).
Proof using contradiction and symmetry:
$a = c \Rightarrow c \preceq b \wedge b \preceq c$
This is a contradiction, because $\preceq$ is antisymmetric!
Therefore $a \neq c$.

Well-ordered A poset $(A;\preceq)$ is well-ordered if it is totally ordered and if every non-empty subset of A has a least element. Not true for $\leq, \geq$, true for $| \subseteq \mathbb{N}^2, \subseteq \mathcal{P}(A)$.

Hasse diagram An element b is said to *cover* a if $a \prec b$. A Hasse diagram represents this covering from top to bottom, where if $a \prec b$, a is below b and there is a path from b to a. Also no transitive relations of $\prec$ are drawn. Always the right-hand side is written above. So for the overlaying relation $A \sqsubseteq B$, B the smaller element is written above A.



Graphically a Hasse diagram describes a topological ordering, where each layer describes another dependency.

Order relation The order relation defined on two posets $(A;\preceq), (B;\sqsubseteq)$ forms a poset: $(A \times B;\leq))$ and is defined as:
$(a_1,b_1) \leq (a_2,b_2) :\Leftrightarrow a_1 \preceq a_2 \wedge b_1 \sqsubseteq b_2$ (also $\leqslant$ is alternatively used).

Lexicographical order A lexicographical order for two given posets $(A;\preceq), (B;\sqsubseteq)$ forms a poset: $(A \times B;\leq_{lex})$ and is defined as:
$(a_1,b_1) \leq_{lex}' (a_2,b_2) :\Leftrightarrow a_1 \prec a_2 \vee (a_1 = a_2 \wedge b_1 \sqsubseteq b_2)$
Attention: (a_1,b_1) not (a_1,a_2)
It can be generalized over k-tuples over some alphabet Σ (denoted Σ^k) and more generally on finite-length strings noted as Σ^*. Proof lexicographical order for two given posets is itself aposet:
$(a_1,b_1) \leq_{lex} (a_2,b_2) :\Leftrightarrow a_1 \prec a_2 \vee (a_1 = a_2 \wedge b_1 \sqsubseteq b_2)$

Reflexive:
$(a_1 \prec a_1 \vee a_1 = a_1 \wedge b_1 \sqsubseteq b_1)$
The left-hand side of the disjunction of the above statement is false, however the right-hand side correctly states that $a_1 = a_1$ and $b_1 \sqsubseteq b_1$ holds, hence it is reflexive.
Antisymmetric:

$$(a_1 \overset{1}{\prec} a_2 \vee (a_1 = a_2 \overset{2}{\wedge} b_1 \sqsubseteq b_2)) \wedge (a_2 \overset{3}{\prec} a_1 \vee (a_2 = a_1 \overset{4}{\wedge} b_2 \sqsubseteq b_1)) \rightarrow (a_1,b_1) = (a_2,b_2)$$
$$((a_1 \preceq a_2 \overset{1}{\wedge} a_1 \neq a_2) \vee (a_1 = a_2 \overset{2}{\wedge} b_1 \sqsubseteq b_2)) \wedge ((a_2 \preceq a_1 \overset{3}{\wedge} a_2 \neq a_1) \vee (a_2 = a_1 \overset{4}{\wedge} b_2 \sqsubseteq b_1)) \rightarrow (a_1,b_1) = (a_2,b_2)$$

The left-hand side of this implication can only be true, when 2 and 4 are true.

1	3	cannot be true, hence they are antisymmetric and non-reflexive
1	4	$a_1 \neq a_2 \wedge a_1 = a_2 \rightarrow \perp$
2	3	$a_1 \neq a_2 \wedge a_1 = a_2 \rightarrow \perp$
2	4	$a_1 = a_2$ follows from 2 and $b_1 \sqsubseteq b_2 \wedge b_2 \sqsubseteq b_1 \rightarrow b_1 = b_2$

Transitivity:

$$(a_1 \overset{1}{\prec} a_2 \vee (a_1 = a_2 \overset{2}{\wedge} b_1 \sqsubseteq b_2)) \wedge (a_2 \overset{3}{\prec} a_3 \vee (a_2 = a_3 \overset{4}{\wedge} b_2 \sqsubseteq b_3)) \rightarrow (a_1 \overset{5}{\prec} a_3 \vee (a_1 = a_3 \overset{6}{\wedge} b_1 \sqsubseteq b_3))$$

The only time the above formula can be false is if the left-hand side is true and the right-hand side is false.

1	3	5 directly follows from transitive property of $\prec$
1	4	5 follows from $a_1 \prec a_2 \wedge a_2 = a_3 \rightarrow a_1 \prec a_3$
2	3	5 follows from $a_1 = a_2 \wedge a_2 \prec a_3 \rightarrow a_1 \prec a_3$
2	4	6 follows from $a_1 = a_2 = a_3 \wedge$ transitive property $b_1 \sqsubseteq b_2 \wedge b_2 \sqsubseteq b_3 \rightarrow b_1 \sqsubseteq b_3$

For all the below definitions, the following holds: $(A;\preceq), S \subseteq A$

Minimal(maximal) element A minimal element is an element in a subset S, such that $\neg\exists b \in S.b \prec a$ (there can be multiple minimal elements) $a \in S, b \in S$.

Least(greatest) element A least (sometimes called smallest) element is a minimal element in a subset S, if there are no other minimal elements in that subset. (Therefore there must be a transitive relation between every other element in S and the least element). $\forall b \in S a \preceq b$, where $a \in S$ is the least element.
Example:
$(\mathbb{N};|)$ has 1 as a least element, but has no greatest element.

Lower bound ***(GER: Untere Schranke)*** $a \in A$ is a lower bound a subset S if $\forall s \in S(a \preceq s)$. The same thing is for upper bounds *(GER: Obere Schranke)* (Keep in mind that a can be in S).

Greatest lower bound (least upper bound) Sometimes noted as $glb(S), lub(S)$.
This is the greatest element in $a \in A$ for which all other elements in the subset S are greater equal to (Keep in mind that a can be in S).
Example:
The subset $3, 6, 9$ has a least upper bound defined by the poset $(\mathbb{N}, |)$, which is $lcm(3, lcm(6, 9)) = 36$.
The subset $3, 6, 9$ has a greatest lower bound defined by the poset $(\mathbb{N}, |)$, which is $gcd(3, gcd(6, 9)) = 3$.

Lattice
If a subset of size 2 $\{a, b\} \subseteq A$ has a greatest lower bound, then that element is called the *meet* or *2-size infimum* of a and b, often noted as $a \wedge b$.
If a subset of size 2 $\{a, b\} \subseteq A$ has a least upper bound, then that element is called the *join* or *2-size supremum* of a and b, often noted as $a \vee b$.
A poset for which every possible pair in A $a \in A, b \in B\{a, b\}$ has a meet and a join, is called a *lattice (GER: Verband)*.
Example:
$(\mathbb{N}, \leq), (\mathbb{N}, \geq)$ are lattices. Their meet and join are simply the smallest or greatest element of the pair.
$(\mathbb{N}, |)$ is a lattice. Its meet is always $gcd(a, b)$, and its join is $lcm(a, b)$. Of course if $lcm(a, b)$ does not exist in a particular set A of the poset $(A; |)$, it has no lattice.
$(\mathcal{P}(A); \subseteq)$ is a lattice. Its meet is always $a \cap b$ and its join $a \cup b$.

5.8.1 Function

The relation $f \subseteq A \times B$ is a function, if it is totally defined and well-defined:

Totally defined $\forall a \in A \exists b \in B$: a f b (f is totally defined)

Well-defined $\forall a \in A \forall (b, b') \in B$: a f b $\wedge$ a f b' $\rightarrow b = b'$ (f is well-defined).

A is called the **domain** *(GER: Definitionsbereich)* and B is called the **codomain** *(GER: Wertebereich)*.
$f(A)$ is called the **image** *(GER: Bild)* $Im(f)$ of f.
$f^{-1}(T)$ is called the inverse, where T is an *inverse image* or *preimage*.
$f : A \mapsto B, T \subseteq B, f^{-1}(T) := \{a \in A | \; f(a) \in T\}$
A function can also be formally described as a mapping:
$f : a \mapsto$ "expression, which can use a"
$f : a \mapsto a^2$
A *partial function* is a relation, which is only well-defined.
The set of all possible functions can be called B^A and its size is $|B|^{|A|}$
Additional properties that functions can have:

Injective A function is injective, if there are no collisions: $\forall a \in A, !\exists b \in B : a\, f\, b$
$\Leftrightarrow \forall a \in A, !\exists b \in B : b = f(a)$

Surjective A function is surjective, if there is: $\forall b \in B, \exists a \in A : a\, f\, b$
$\Leftrightarrow \forall b \in B, \exists a \in A : b = f(a)$

Bijective A function is bijective, if it is both injective and surjective.

Composition The composition of a function $f : A \mapsto B$ and a function $g : B \mapsto C$ is defined as:
$g \circ f :\equiv g(f(a))$.
Function composition is associative, the same way relation composition is associative.

5.8.2 Countability of sets

Equivalent cardinality
$A \sim B :\Leftrightarrow |A| = |B| :\Leftrightarrow$there exists a bijective function $A \mapsto B$

Cardinality of B at least cardinality of A
$A \preceq B \Leftrightarrow |A| < |B| :\Leftrightarrow$ there exists an injective function $A \mapsto B$
Alternate: $A \preceq B :\Leftrightarrow A \mapsto C$(bijective)$\wedge C \subseteq B$
Transitive: $A \preceq B \wedge B \preceq C \Rightarrow A \preceq C$
Proof: There is an injenction from A to B and an injection from B to C, then A to C has an injection.
$A \subseteq B \Rightarrow A \preceq B$
Proof: id_A is an injection from A to B.
Schröder-Bernstein theorem: $A \preceq B \wedge B \preceq A \Rightarrow A \sim B$, which is seemingly hard to prove.

Countable
$A \preceq \mathbb{N}$.
Alternate: A set A is countable if and only if it is finite or if $A \sim \mathbb{N}$.
Proof see below.
Uncountable if otherwise.

Example:
$[-1, 3] \sim \mathbb{R}$
The below function maps every rational number to a number in the range $[-1, 3]$

$$\begin{cases} r + 2 \text{ if } r > -1 \wedge r < 1 \\ \frac{1}{r} & \text{elsewise} \end{cases}$$

The below function maps every number in the range $[-1, 3]$ to a rational number.

$$\begin{cases} p - 2 \text{ if } r > 1 \wedge r < 3 \\ \frac{1}{p} & \text{elsewise} \end{cases}$$

Hence $[-1, 3] \preceq \mathbb{R} \wedge \mathbb{R} \preceq [1, -3] \Rightarrow [-1, 3] \sim \mathbb{R}$.
Example:
$\mathbb{Z} \sim \mathbb{N}$
$(-1)^a + \lceil \frac{a}{2} \rceil$

There is no cardinality between a finite cardinality and $|\mathbb{N}|$. Alternatively: A set A is countable iff finite or $A \sim \mathbb{N}$
$A \preceq \mathbb{N} \Rightarrow |A| = c \vee A \sim |\mathbb{N}|$
$C = f(A) \subseteq \mathbb{N}$ C = Wertemenge
Since $f(A)$ is well-ordered, there must be a smallest element in A. Therefore we can construct a bijunction, from the least element to the next greatest element.
$g : C \mapsto \mathbb{N}$
$g \circ f$ is a bijection $A \mapsto \mathbb{N}$
Important countable sets:

1.
$\{0, 1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, ...\} \sim \; \mathbb{N}$
The injective function is $2^{|s|} + s.toInt()$, where s is the string representation of the bit sequence, which maps to $\mathbb{N}$.
$\{0, 1\}^{\infty} \not\sim \in \mathbb{N}$ is not countable.

2.

$\mathbb{N} \times \mathbb{N} \sim \mathbb{N}$
This can be shown using **cantors pairing function**.
Cantors pairing function for $\mathbb{N}_0 \times \mathbb{N}_0 \mapsto \mathbb{N}_0$:
$\pi_0(x, y) = \frac{1}{2}(x + y)(x + y + 1) + y$
Cantors pairing function for $\mathbb{N} \times \mathbb{N} \mapsto \mathbb{N}$:
$\pi(x, y) = \frac{1}{2}(x + y - 2)(x + y - 1) + y$

3.

$A \preceq \mathbb{N} \land B \preceq \mathbb{N} \Rightarrow A \times B \preceq \mathbb{N}$
Concrete you can take the injective function $f : A \mapsto \mathbb{N}$ and $g : B \mapsto \mathbb{N}$ and just modify f to $2 * f$ and g to $2 * g + 1$.

4.

$A^n \preceq \mathbb{N}, \forall n \in \mathbb{N}$
This can be done inductively:
$A^n \times A$ is countable.
Base step: A is countable.
Inductive step: $A^{n+1} = A^n \times A$ is countable, hence $\pi(A^n, A) \mapsto \mathbb{N}$.

5.

$\mathcal{A} = \bigcup_{i \in \mathbb{N}} A_i \sim \mathbb{N}, \mathcal{A} \sim \mathbb{N}$
Proof:
Assume that no A_i shares an element with an other set (Elsewise consider the sets $A'_1 = A_1, A'_2 = A_2 - (A_1 \cap A_2), A'_3 = A_3 - (A_1 \cap A_2 \cap A_3)...$)
Now the elements can be written in a table (first column have the elements of A'_1, second of A'_2).
Hence $\mathbb{N} \times \mathbb{N}$ is countable, $\mathbb{N}$ is countable.

6.

$A^* \sim \mathbb{N}$, where A^* are finite sequences of elements from A. Proof using combination of 4 and 5.

Uncountable set $\{0,1\}^\infty \not\sim \mathbb{N}$

Proof using contradiction:
Assumption: There exists a bijunction $f^{-1} : \{0,1\}^\infty \mapsto \mathbb{N}$.
$f : \mathbb{N} \mapsto \{0,1\}^\infty$
$f(0) = 0110110001...$
$f(1) = 1001101001...$
$f(2) = 0011001010...$
$\beta_{m,n} = \text{n-th bit in } f(m)$.
$\alpha := \overline{\beta_{0,0}}, \overline{\beta_{1,1}}, \overline{\beta_{2,2}}...$
Now α does not exist in $f(i)$, hence α is at least different by one digit from each $f(i), \forall i \in \mathbb{N}$.
The same argument can be used with the irrational numbers between 0 and 1 exclusive in binary.

Uncomputable functions

A function $f : \mathbb{N} \mapsto \{0,1\}$ is called computable, if there is a program that for every $n \in \mathbb{N}$ outputs $f(n)$.
Such a computable function would be $prime(n)$. All inputs listed: $0110101000101000101000101000001... \in \{0,1\}^\infty$

There are countably infinite programs, every program is of type $\{0,1\}^*$.
There are countably infinite programs which compute a function of the form $\mathbb{N} \mapsto \{0,1\}$.
Hence the programs are countable, we can have an order of these functions:
$f_1(0) = 0110101000101..$ (prime numbers)
$f_2(0) = 1110100100001..$ (fibonacci numbers)
$f_3(0) = 0101010101010..$ (even numbers)
$\vdots$
But now we can use the diagonalization argument to show that not every function of the form $\mathbb{N} \mapsto \{0,1\}$ is contained in this list.
If it is not in the list of countable functions, it is not computable. Therefore there exist uncomputable functions. (Concretely: Halting problem).

5.9 Defining natural numbers

$0 := \varnothing$
$1 := \{\varnothing\}$
$2 := \{\varnothing, \{\varnothing\}\}$
$s(n) = n + 1 := n \cup \{n\}$

6 Number theory

Strictly number theory is the study of the universe $\mathbb{Z}$. Whenever quantification is used, it is used over the whole numbers.

Divides

$a|b :\Leftrightarrow \exists e(b = e * a)$
b is divisible by a.
b is a multiple of a (All multiples are $\{b : a|b\}$)
a is a divisor (or factor) of b (All divisors are $\{a : a|b\}$).
Properties:

1. $a|b \Rightarrow \forall k\, a|(k * b)$
2. $c|a \wedge c|b \Rightarrow c|(a \pm b)$
3. $c|a \wedge c|b \Rightarrow c|(a * b)$
4. $a|b \wedge b|c \Rightarrow a|c$ (transitive property)

Euclids division (Euclidean ring)

$\forall a \forall b \neq 0 \exists q \exists r : (a = b * q + r) \wedge 0 \leq r < |d|$
a is the dividend, b is the divisor and q is called the quotient.
$\frac{a}{b} = q$ rest $r = q + \frac{r}{b} = q + R_b(a)$
Negative division is uniquely defined, since the remainder stays positive.
$\frac{-26}{3} = 3 * (-9) + 1$

Greatest common divisor ***(GER: Grösster gemeinsamer Teiler)***

Universe of discourse $= \mathbb{Z}$.
d is a greatest common divisor, if:
$d|a \wedge d|b \wedge \forall c(c|a \wedge c|b \rightarrow c|d)$
Example: -5 and 5 are greatest common divisors of 15 and 25.
However the gcd is always positive, so $gcd(15, 25) = 5$ and is defined as:
$\exists u, b\ gcd(a, b) = u * a + v * b$ Two numbers a and b are called *relatively prime* or *coprime* if $gcd(a, b) = 1$. This means they do not share common divisors.

1. $gcd(m, n) = gcd(n, m - q * n)$
   Partial Proof:
   $d|m \wedge d|n \Leftrightarrow d|n \wedge d(m - qn)$
   $n = fd \Rightarrow (m - qn) = ed - qfd = c(e - qf)d$
2. $\forall a, b \exists d(a, b) = (d)$

   Universe for (a, b).
   Without proof: $(d) = gcd(a, b)$
   Existence Proof:
   $(a, b) \neq \varnothing, (a, b)\ \cap \mathbb{N} \neq \varnothing$, falls $a \neq 0$ und $b \neq 0$
   $d > 0$, d the smallest element.
   $(a, b) \neq \varnothing$
   Behauptung:
   $(d) = (a, b)$
   $\Leftrightarrow (d) \subseteq (a, b) \wedge (a, b) \subseteq (d)$
   Beweis von $(a, b) \subseteq (d)$
   Sei $c \in (a, b)$ (beliebig)
   $c = dq + r, \quad 0 \leq r < d$
   $r = c - dq \in (a, b)$
   Da d das kleinste Element ist:
   $r = 0$
3. $\forall a, b \exists u, v gcd(a, b) = ua + vb$
   Corollary from 2.

Ideal

$a, b \in \mathbb{Z}$

$(a) := \{ua|\ u \in \mathbb{Z}\}$
$(a, b) := \{ua + vb|\ u, v \in \mathbb{Z}\}$ Properties:

1. $\forall a, b \in \mathbb{Z} \exists d \in \mathbb{Z}\ (a, b) = (d)$
   This is a greatest common divisor of a and b, which is $\pm gcd(a, b)$ or 0 if $a = b = 0$.

Example: $(4, 7)$
$(4, 7) = \{1 = 2 * 4 - 1 * 7, 2 = 4 * 4 - 2 * 7\}$.
So we can express each number as a multiple of 1: $k = 2 * k * 4 - k * 7$.

Gaussian numbers are complex numbers, where $a + bi, a \in \mathbb{Z} \wedge b \in \mathbb{Z}$

GCD algorithm simple example

Calculating $gcd(309, 21)$:
$309 = 21 * 14 + 15$
$21 = 15 * 1 + 6$
$15 = 6 * 2 + 3$
$6 = 3 * 2 + 0$
$gcd(309, 21) = 3$
However the greatest common divisor can be uniquely expressed as $\forall a \forall b(gcd(m, n) = a * m + b * n)$.
In order to find out m and n one can simply solve the above equations for the reminder and replace all summands with their new remainder representation from the bottom-up. At the end only factors of m and n should be present:
$3 = 15 - 6 * 2$
$3 = 15 - (21 - 15 * 1) * 2$
$3 = (309 - 21 * 14) - (21 - (309 - 21 * 14)) * 2 = 3 * 309 - 44 * 21$
The gcd can also be found by writing down all divisors of both numbers and choosing the greater one of both.
The gcd can also be found by factoring both numbers into primes and taking all their exponents.
$a = p_1^{e_1} p_2^{e_2} p_3^{e_3} \dots$
$b = p_1^{f_1} p_2^{f_2} p_3^{f_3} \dots$
$gcd(a, b) = p_1^{min(e_1, f_1)} p_2^{min(e_2, f_2)} \dots$
$lcm(a, b) = p_1^{max(e_1, f_1)} p_2^{max(e_2, f_2)} \dots$

Extended GCD

```
s_1 = a; s_2 = b;
u_1 = 1; u_2 = 0;
v_1 = 1; v_2 = 0;
while(s_2 > 0) {
    q := s_1 | s_2; // q = s_1 % s_2
    r := s_1 - q s_2;
    s1 := s_2;
```

```
        s2 = r;
        t = u_2;
        u_2 = u_1 - q u_2;
        u_1 = t; //old u_2
        t := v_2;
        v_2 := v_1 - q * v_2;
        v_1 := v_2; //old v_2
    }
    gcd = s_1; u := u_1; v := v_1;
```

Primes

$prime(p) :\Leftrightarrow \forall x(x \in \mathbb{N} \land x \neq \pm 1 \land x \neq \pm p \rightarrow (x \not| p))$
$prime(p) :\Leftrightarrow \forall x \in \mathbb{N}_{\geq 2}(x|p) \rightarrow x = 1 \lor x = p$
$prime(p) :\Leftrightarrow \forall a \forall b(p|(a * b) \Rightarrow p|a \lor p|b)$

Prime factorization uniqueness

Every positive integer greater than 1 can be expressed by an unique prime factorization.
$a = p_1^{e1} p_2^{e2} p_3^{e3} \ldots$. For most exponents $e_i = 0$.
Proof not mentioned, due to size.

Irrationality of Roots

Every positive integer has an irrational root, except square numbers (for example $4 = \sqrt{16}$). Proof:
Suppose n is not a square:

$\sqrt{n} = \frac{a}{b}$
$a^2 = nb^2$
If n is not a square, it contains at least one prime factor p with an odd power. a^2, b^2 only have prime factors of even degree. Due to the uniqueness of prime factorization, it cannot be that one side has an odd power and the other not.

Least common multiple

The least common multiple divides all other multiples of a and b.
$l = lcm(a, b) \Leftrightarrow a|l \land b|l \land \forall m((a|m \land b|m) \rightarrow l|m)$
$lcm(a, b) = \frac{a*b}{gcd(a,b)}$

Diophantine equations $x^3 + 1 = y^2$ has a solution, $x = 2, y = 3$

$x^3 + x + 1 = y^2 + y$ has no solutions. Proof by contradiction:
$x^3 + x + 1$ is always odd.
$y^2 + y$ is always even. So no solution exists.
$a * x + b * y = z, gcd(a, b) = u * a + v * b$.
By multiplying $u * a + v * b$ by $\frac{z}{gcd(a,b)}$ you get one solution.
A solution of the above form only has an integer solution if $gcd(a, b)|z$.
For more complicated equations, more complicated solutions can be found:
$x^3 - x - 1 = y^2$
We can show that the left-hand side modulo 3 always results in 2: $R_3(x^3 - x - 1) = 2$.
The right-hand has modulo 0 or 1. $R_3(y^2) \in \{0, 1\}$
$x^5 + 6 = y^2$ with mod 11.

Fermats big theorem $x^n + y^n = z^n$

$n = 2$ has solution.
Wiles: $n > 2$ has no solutions.

Congruence modulo

$a, b, m \in \mathbb{Z}, m \geq 2, a \equiv_m = b :\Leftrightarrow m|(a - b)$.
Alternatively: $a \equiv_m = b :\Leftrightarrow a = k * m + b$
m is called the *modulus*. a is *congruent* to b.
Alternatively the congruence relation is written as:
$a \equiv b(\text{mod } m)$ and even as $a \equiv b(m)$
$a \equiv_2 0$ results in a is even.
Sample congruence proof: $a = b \Rightarrow a \equiv_m b$. Proof: $m|(a - a) = m|0 = \top$
Proof of equivalence relation:
Proof of equivalence of $\equiv_m$:
Reflexive: $m|0 \Rightarrow m|(a - a) \Rightarrow a \ \equiv_m a$ Symmetric:
$a \equiv_m b \Rightarrow b \equiv_m a$
$m|(a - b) \Rightarrow \exists k a - b = k * m$
$\Rightarrow \exists k b - a = (-k) * m$
$\Rightarrow \exists \widetilde{k} b - a = \widetilde{k} * m$
$\Rightarrow m|(b - a)$
Transitive:
$a \equiv_m b \land b \equiv_, c \Rightarrow a \equiv_m c$
$m|(a - b) \land m|(b - c)$
$\Rightarrow m|(a - b + b - c)$
$\Rightarrow m|(a - c)$

Properties:

1. $a \equiv_m b \Rightarrow k * a \equiv_m k * b$
2. If $gcd(k, m) = 1$, then $k * a \equiv_m k * b \Rightarrow a \equiv_m b$
3. $a \equiv_m b \land c \equiv_m d \Rightarrow a + c \equiv_m b + d$
   Proof: $m|(a - b) \land m|(c - d)$
   $\Rightarrow m| \ (a - b) + (c - d) \Rightarrow ((a + c) - (b + d))$
4. $a \equiv_m b \land c \equiv_m d \Rightarrow a * c \equiv_m b * d$
   Proof:
   $m|(a - b) \Leftrightarrow m = e * (a - b) \Leftrightarrow b = a - \frac{m}{e}$ (1)
   $m|(c - d) \Leftrightarrow m = f * (c - d) \Leftrightarrow d = c - \frac{m}{f}$ (2)
   $(a * c - b * d) \Rightarrow_{(1)\&(2)} (a * c - (a - \frac{m}{e})(c - \frac{m}{f})) \Rightarrow (a * c - a * c + a * \frac{m}{f} + c * \frac{m}{e} - \frac{m}{e*f})$
   $(m * (\frac{a}{f} + \frac{c}{f} - \frac{1}{ef})) \Rightarrow m|(a * d - b * c)$

4b $a \equiv_m b \Rightarrow a^n \equiv_m b^n (n \in \mathbb{Z}_{\geq 0})$
5 $a \not\equiv_m b \Rightarrow a \neq b$
6 Given two multi-variate polynomials $p = f(a_1, a_2, ..., a_n)$ and $q = f(b_1, b_2, ..., b_n)$ and $a_i \equiv_m b_i \Rightarrow p(a_1, a_2, ..., a_n) \equiv_m q(b_1, b_2, ..., b_n)$

Remainder

The remainder of a division by m can be written as: $R_m(a)$
Properties:

1. $a \equiv_m R_m(a)$
   Proof:

$a = k * m + R_m(a)$
$a - R_m(a) = k * m$
$a \equiv_m R_m(a)$

2. $a \equiv_m b \Leftrightarrow R_m(a) \equiv_m R_m(b)$
   Proof:
   $a = k * m + R_m(a)$
   $b = q * m + R_m(b)$
   $R_m(a) - R_m(b) = m|(...)$
   $\Rightarrow R_m(a) \equiv_m R_m(b)$
   $\Rightarrow R_m(a) = R_m(b)$

3. $R_m(a) \subseteq [a]_{\equiv_m}$

4. $R_m(R_m(a)) = R_m(a)$
5. $R_m(x) = 0 \Leftrightarrow x$ has a prime factorization which includes m.

Division test by 3 can be proven easily:
A number $ABC \equiv_3$ can be expressed as: $10^2 A + 10B + C \equiv_3$
So $R_m(10^2)A + R_m(10) + C \equiv_3$
Since $10^n \equiv_3 1$, the number can be easily calculated by the sum of the digits mod 3.

Modular arithmetic $a, b, m \in \mathbb{Z}$
Properties:

1. $R_m(a + b) = R_m\left(R_m(a) + R_m(b)\right)$

2. $R_m(a * b) = R_m\left(R_m(a) * R_m(b)\right)$

Proof 1:
$a + b \equiv_m R_m(a) + R_m(b)$
$R_m(a + b) \equiv_m R_m(R_m(a) + R_m(b))$

Proof 2:
$a = k * m + R_m(a)$
$b = q * m + R_m(b)$
$R_m(a * b) = R_m(k * q * m^2 + k * m * R_m(b) + q * m * R_m(a) + R_m(a) * R_m(b))$
$\Rightarrow R_m(a * b) = R_m(R_m(a) * R_m(b))$

Techniques for fast modulo exponentiation
$R_{19}(7^6) = R_{19}(7^{2*3}) = R_{19}((7^3)^2) = R_{19}(R_{19}(343)^2) = R_1 91^2 = 1$
$R_{19}(7^{11}) = R_{19}(7^{8+2+1}) = R_{19}(7^8) * R_{19}(7^2) * R_{19}(7^1)$

Linear congruence
A linear congruence, with x being an unknown is of the form $ax \equiv_m b$
If x_0 is a solution, then $\forall k \in \mathbb{Z}(x_0 + k * m)$ is a solution (or all elements in $[x_0]_{\equiv_m}$).
Still every number up to m has to be checked. Example:
$3 * x \equiv_9 3, x = [1]_9 \cup [4]_9 \cup [7]_9$
The solution must span $gcd(a, m)$ equivalence classes!
There is only a solution if $gcd(a, m)|b$.
All solutions are $\forall i x_i = x_0 + \frac{m}{d} * i$

Multiplicative inverse $ax \equiv_m 1$ has a solution iff $gcd(a, m) = 1$ (a,m are coprime).
The solution is unique in $\mathbb{Z}_m$!
The multiplicative inverse is also noted as: $x \equiv_m a^{-1}$ or $x \equiv_m \frac{1}{a}$.
It can be found by calculating the extended gcd: $gcd(a, m) = u * a + v * m$. u is the inverse!
Proof:
$\Rightarrow$ assume x is a solution.
$ax = k * m + 1, k \in \mathbb{Z}$
$gcd(a, m)|a, \quad gcd(a, m)|m$
$\Rightarrow gcd(a, m)|(ax - km)$
$gcd(a, m)|1$
$gcd(a, m) = 1$
$\Leftarrow$ assuming $gcd(a, m) = 1$
$\Rightarrow u * a + v * m = 1$
$\Rightarrow u * a \equiv_m 1$
$\Rightarrow R_m(u) * a \equiv_m 1 \in \mathbb{Z}$
Now we have to prove that there is more than 1 solution.
x, x' are two solutions.
$ax - ax' \equiv_m 0$
$m|\ (a(x - x'))$
$m|\ (x - x')$
$\Rightarrow m|(x - x')$ (Because gcd(a,m) is 1).
$\Rightarrow R_m(x) = R_m(x')$
$\Rightarrow x = x'$
If $\exists x$ then $ax \equiv_m 1$ such that the multiplicative inverse of $a\, mod\, m$
$\mathbb{Z}_m^* = \{a \in \mathbb{Z}_m | gcd(a, m) = 1\}$

Example:
$7 * x \equiv_1 51$
$gcd(7, 15) = 1$

Chinese remainder theorem
Given n equations of the form:
$x \equiv_{m_1} a_1$
$x \equiv_{m_2} a_2$
$\vdots$
$x \equiv_{m_n} a_n$
The chinese remainder theorem states that there is a unique solution for $x \in \mathbb{Z}_m$.
$x = R_M\left(M_1 * a_1 * X_1 + M_2 * a_2 * X_2 + \ldots + M_n * a_n * X_n\right)$
$x + M * k$ are also solutions.
The intuition behind using M_i is that for solving X_i, the other summands will turn out to be 0.
$x = M_1 * a_1 * X_1 \; M = \prod m_i$
$m_1, m_2, ..., m_r$ which are *(GER: teilerfremd)* pair-wise coprime.
$a_1, a_2, ..., a_r, \qquad 0 \leq a_i \leq m_i$
The system of equations has exactly one solution in $[0, M)$.
$M_i = \frac{M}{m_i}$
X_i can be found by solving $M_i * X_i \equiv_{m_i} = 1$ using modular inverse or trial & error.

Due to Property 1, the solutions are only $[X_i]_{\equiv_{m_i}}$
Property 1: $\Rightarrow gcd(M_i, m_i) = 1$
Property 2: $\Rightarrow (i \neq j)\ M_i * X_i \equiv_{m_j} 0$
Property 3: $\sum_{i=1}^{r} a_i M_i N_i \equiv_{m_k} a_k$

Techniques for solving chinese remainder theorem problems

Example:
$x \equiv_3 4 \Rightarrow x \equiv_3 1$
$x \equiv_4 2$
$x \equiv_5 3$

1. Either calculate the result using the above construction.
$x = R_{60}(\frac{60}{3} * 1 * X_1 + \frac{60}{4} * 2 * X_2 + \frac{60}{5} * 3 * X_3)$
$x = R_{60}(20 * 1 * X_1 + 15 * 2 * X_2 + 12 * 3 * X_3)$
$X_1 : 20 * X_1 \equiv_3 1 \Rightarrow X_1 = 2$
$X_2 : 15 * X_2 \equiv_4 1 \Rightarrow X_2 = 3$
$X_3 : 12 * X_3 \equiv_5 1 \Rightarrow X_3 = 3$
$x = R_{60}(20 * 1 * 2 + 15 * 2 * 3 + 12 * 3 * 3) = R_{60}(238) = 58$ 2. If the numbers are small, find x using trial & error and then use the chinese remainder property to find out.
According to 3: The last number is either 3 or 8.
According to 2: The last number is even. Then the last number can only be 8.
x must be in the range of $0, 59$:
Remaining candidates:
$8, 18, 28, 38, 48, 58$
According to 1: Only $28, 58$ are possible solutions.
According to 2 again: 58 is the only number dividable by 4.
Only 58 remains and must be the solution!

Diffie-Hellman Key-agreement

Diffie-Hellman Key-agreement is used in order to generate symmetric keys.
Using Alice and Bob's key agreement, both generate a private x_* / public key y_*.
In order to do this they need a prime number p and some generator g, which spans the subgroup: $< g >= \{g^x | x \in \mathbb{Z}\}$. The generator does not need to span $\mathbb{Z}_p^*$.
Every generator will work, however the order should be as large as possible.
The diffie-hellman key-agreement works for all groups, however it is only safe if the *discrete logarithm problem* is hard (solving x_A in the equation: $y_A = g^{x_A}$).
For integers a prime number p and a generator (element in $\mathbb{Z}_p^*$) are publicly chosen. Then Alice and Bob choose their integer private key in the range $\{0..p - 2\}$.
Now each of them compute their public key and send it to the other person:
$y_A = R_p(g^{x_A})$
$y_B = R_p(g^{x_B})$
In order to compute their shared key both can compute:
$k_{AB} = y_B^{x_A} = g^{x_B} x_A = y_A^{x_B}$