Discrete Mathematics 1

Computer Science Tripos, Part 1A

Natural Sciences Tripos, Part 1A, Computer Science

Politics, Psychology and Sociology Part 1, Introduction to Computer Science

Peter Sewell

1A, 8 lectures

2009 – 2010

Introduction

At the start of the Industrial Revolution, we built bridges and steam engines without enough applied maths, physics, materials science, etc.

FIG. 5. AN ILLUSTRATION OF WHAT EXPLOSION DID TO STAYS AND BRACES

Fix: understanding based on continuous-mathematics models — calculus, matrices, complex analysis,...

Introduction

Now, we build computer systems, and sometimes...

[Ariane 501]

Introduction

Now, we build computer systems, and sometimes...

[Ariane 501]

But, computer systems are large and complex, and are *discrete*:

we can’t use approximate continuous models for correctness reasoning.

So, need *applied discrete maths* — logic, set theory, graph theory,

combinatorics, abstract algebra, ...

Logic and Set Theory — Pure Mathematics

Origins with the Greeks, 500–350 BC, philosophy and geometry:

Aristotle, Euclid

Formal logic in the 1800s:

De Morgan, Boole, Venn, Peirce, Frege

Set theory, model theory, proof theory; late 1800s and early 1900s:

Cantor, Russell, Hilbert, Zermelo, Frankel, Goedel, Turing, Bourbaki, Gentzen, Tarski

Focus then on the foundations of mathematics — but what was developed then turns out to be unreasonably effective in Computer Science. This is the core of the applied maths that we need.

Logic and Set Theory — Applications in Computer Science

- modelling digital circuits (1A Digital Electronics, 1B ECAD)
- proofs about particular algorithms and code (1A Algorithms 1, 1B Algorithms 2)
- proofs about what is (or is not!) computable and with what complexity (1B Computation Theory, Complexity Theory)
- proofs about programming languages and type systems (1A Regular Languages and Finite Automata, 1B Semantics of Programming Languages, 2 Types)
- foundation of databases (1B Databases)
- automated reasoning and model-checking tools (1B Logic & Proof, 2 Specification & Verification)

Outline

- Propositional Logic
- Predicate Logic
- Sets
- Inductive Proof

Focus on *using* this material, rather than on metatheoretic study.

More (and more metatheory) in Discrete Maths 2 and in Logic & Proof.

New course last year — feedback welcome.

Supervisons

Not rocket science (?), but *needs practice* to become fluent.

Five example sheets. Many more suitable exercises in the books.

Up to your DoS and supervisor, but I'd suggest 3 supervisons. A possible schedule might be:

1. After the first 2–3 lectures (on or after Thurs 19 Nov)

   Example Sheets 1 and 2, covering Propositional and Predicate Logic

2. After the next 3–4 lectures (on or after Thurs 26 Nov)

   Example Sheets 3 and the first part of 4, covering Structured Proof and Sets

3. After all 8 lectures (on or after Thurs 3 Dec)

   Example Sheet 4 (the remainder) and 5, covering Inductive Proof

Propositional Logic

Propositional Logic

Starting point is informal natural-language argument:

Socrates is a man. All men are mortal. So Socrates is mortal.

Propositional Logic

Starting point is informal natural-language argument:

Socrates is a man. All men are mortal. So Socrates is mortal.

If a person runs barefoot, then his feet hurt. Socrates' feet hurt.

Therefore, Socrates ran barefoot

It will either rain or snow tomorrow. It’s too warm for snow. Therefore, it will rain.

It will either rain or snow tomorrow. It’s too warm for snow. Therefore, it will rain.

Either the butler is guilty or the maid is guilty. Either the maid is guilty or the cook is guilty. Therefore, either the butler is guilty or the cook is guilty.

It will either rain or snow tomorrow. It’s too warm for snow. Therefore, it will rain.

Either the framger widget is misfiring or the wrompal mechanism is out of alignment. I’ve checked the alignment of the wrompal mechanism, and it’s fine. Therefore, the framger widget is misfiring.

Either the framger widget is misfiring or the wrompal mechanism is out of alignment. I've checked the alignment of the wrompal mechanism, and it's fine. Therefore, the framger widget is misfiring.

Either p *or* q. *Not* q. *Therefore,* p

Atomic Propositions

$1 + 1 = 2$

$10 + 10 = 30$

Tom is a student

Is Tom a student? ×

Give Tom food! ×

$x + 7 = 10$ ×

$1 + 2 + \ldots + n = n(n+1)/2$ ×

Atomic Propositions

When we're studying logic, instead of fixing some particular language of atomic propositions, we'll use *propositional variables* p, q, etc. In a particular context, each of these might be true or false (but not 21.5).

Compound Propositions

We'll build more complex *compound propositions* out of those of atomic propositions. Any *propositional variable* p, q, etc., is trivially a compound proposition.

We'll write p, q, etc. for arbitrary propositional variables.

We'll write P, Q, etc. for arbitrary compound propositions.

Building Compound Propositions: Truth and Falsity

We’ll write T for the constant true proposition, and F for the constant false proposition.

Building Compound Propositions: Conjunction

If P and Q are two propositions, $P \wedge Q$ is a proposition.

Pronounce $P \wedge Q$ as ‘P and Q’. Sometimes written with $\&$ or $.$

Definition: $P \wedge Q$ is true if (and only if) P is true and Q is true

Examples:

Tom is a student $\wedge$ Tom has red hair

$(1 + 1 = 2) \wedge (7 \leq 10)$

$(1 + 1 = 2) \wedge (2 = 3)$

$((1 + 1 = 2) \wedge (7 \leq 10)) \wedge (5 \leq 5)$

$(\mathrm{p} \wedge \mathrm{q}) \wedge \mathrm{p}$

Building Compound Propositions: Conjunction

We defined the meaning of $P \wedge Q$ by saying ‘$P \wedge Q$ is true if and only if P is true and Q is true’.

We could instead, equivalently, have defined it by enumerating all the cases, in a *truth table*:

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

According to this definition, is $((1 + 1 = 2) \wedge (7 \leq 10)) \wedge (5 \leq 5)$ true or false?

Building Compound Propositions: Conjunction

We pronounce $P \wedge Q$ as 'P and Q', but not all uses of the English 'and' can be faithfully translated into $\wedge$.

Tom and Alice had a dance.

Grouping

Tom went to a lecture and had lunch.

Temporal ordering?

The Federal Reserve relaxed banking regulations, and the markets boomed.

Causality?

When we want to talk about time or causality in CS, we'll do so explicitly; they are not built into this logic.

Building Compound Propositions: Conjunction

Basic properties:

The order doesn't matter: whatever P and Q are, $P \wedge Q$ means the same thing as $Q \wedge P$.

Check, according to the truth table definition, considering each of the 4 possible cases:

P	Q	$P \wedge Q$	$Q \wedge P$
T	T	T	T
T	F	F	F
F	T	F	F
F	F	F	F

In other words, $\wedge$ is *commutative*

Building Compound Propositions: Conjunction

...and:

The grouping doesn't matter: whatever P, Q, and R are, $P \wedge (Q \wedge R)$ means the same thing as $(P \wedge Q) \wedge R$.

> (Check, according to the truth table definition, considering each of the 8 possible cases).

In other words, $\wedge$ is *associative*

So we'll happily omit *some* parentheses, e.g. writing $P_1 \wedge P_2 \wedge P_3 \wedge P_4$ for $P_1 \wedge (P_2 \wedge (P_3 \wedge P_4))$.

Building Compound Propositions: Disjunction

If P and Q are two propositions, $P \vee Q$ is a proposition.

Pronounce $P \vee Q$ as ‘P or Q’. Sometimes written with $|$ or $+$

Definition: $P \vee Q$ is true if and only if P is true or Q is true

Equivalent truth-table definition:

P	Q	$P \vee Q$
T	T	T
T	F	T
F	T	T
F	F	F

Building Compound Propositions: Disjunction

You can see from that truth table that $\vee$ is an *inclusive* or: $P \vee Q$ if *at least one* of P and Q.

$(2+2=4) \vee (3+3=6)$ is true

$(2+2=4) \vee (3+3=7)$ is true

The English ‘or’ is sometimes an *exclusive* or: $P \text{ xor } Q$ if *exactly* one of P and Q. ‘Fluffy is either a rabbit or a cat.’

P	Q	$P \vee Q$	$P \text{ xor } Q$
T	T	T	F
T	F	T	T
F	T	T	T
F	F	F	F

(we won’t use xor so much).

Building Compound Propositions: Disjunction

Basic Properties

$\vee$ is also commutative and associative:

$P \vee Q$ and $Q \vee P$ have the same meaning

$P \vee (Q \vee R)$ and $(P \vee Q) \vee R$ have the same meaning

$\wedge$ distributes over $\vee$:

$P \wedge (Q \vee R)$ and $(P \wedge Q) \vee (P \wedge R)$ have the same meaning

‘P and either Q or R’ ‘either (P and Q) or (P and R)’

and the other way round: $\vee$ distributes over $\wedge$

$P \vee (Q \wedge R)$ and $(P \vee Q) \wedge (P \vee R)$ have the same meaning

When we mix $\wedge$ and $\vee$, we take care with the parentheses!

Building Compound Propositions: Negation

If P is some proposition, $\neg P$ is a proposition.

Pronounce $\neg P$ as ‘not P’. Sometimes written as $\sim P$ or $\overline{P}$

Definition: $\neg P$ is true if and only if P is false

Equivalent truth-table definition:

P	$\neg P$
T	F
F	T

Building Compound Propositions: Implication

If P and Q are two propositions, $P \Rightarrow Q$ is a proposition.

Pronounce $P \Rightarrow Q$ as 'P implies Q'. Sometimes written with $\longrightarrow$

Definition: $P \Rightarrow Q$ is true if (and only if), whenever P is true, Q is true

Equivalent truth-table definition:

P	Q	$P \Rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

Building Compound Propositions: Implication

That can be confusing. First, the logic is not talking about causation, but just about truth values.

$(1 + 1 = 2) \Rightarrow (3 < 4)$ is true

Second, $P \Rightarrow Q$ is vacuously true if P is false.

‘If I’m a giant squid, then I live in the ocean’

For that to be true, either:

(a) I really am a giant squid, in which case I must live in the ocean, or

(b) I’m not a giant squid, in which case we don’t care where I live.

$P \Rightarrow Q$ and $(P \wedge Q) \vee \neg P$ and $Q \vee \neg P$ all have the same meaning

Building Compound Propositions: Implication

Basic properties:

$P \Rightarrow Q$ and $\neg Q \Rightarrow \neg P$ have the same meaning

$\Rightarrow$ is not commutative: $P \Rightarrow Q$ and $Q \Rightarrow P$ do not have the same meaning

$P \Rightarrow (Q \wedge R)$ and $(P \Rightarrow Q) \wedge (P \Rightarrow R)$ have the same meaning

$(P \wedge Q) \Rightarrow R$ and $(P \Rightarrow R) \wedge (Q \Rightarrow R)$ do not

$(P \wedge Q) \Rightarrow R$ and $P \Rightarrow Q \Rightarrow R$ do

Building Compound Propositions: Bi-Implication

If P and Q are two propositions, $P \Leftrightarrow Q$ is a proposition.

Pronounce $P \Leftrightarrow Q$ as 'P if and only if Q'. Sometimes written with $\leftrightarrow$ or $=$.

Definition: $P \Leftrightarrow Q$ is true if (and only if) P is true whenever Q is true, and vice versa

Equivalent truth-table definition:

P	Q	$P \Leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

The Language of Propositional Logic

Summarising, the formulae of propositional logic are the terms of the grammar

$$P, Q ::= p \mid \mathsf{T} \mid \mathsf{F} \mid \neg P \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid P \Leftrightarrow Q$$

where p ranges over atomic propositions p, q, etc., and we use parentheses (P) as necessary to avoid ambiguity.

For any such formula P, assuming the truth value of each atomic proposition p it mentions is fixed (true or false), we've defined whether P is true or false.

Example Compound Truth Table

Given an arbitrary formula P, we can calculate the meaning of P for all possible assumptions on its atomic propositions by enumerating the cases in a truth table.

For example, consider $P \overset{\mathrm{def}}{=} ((\mathrm{p} \vee \neg \mathrm{q}) \Rightarrow (\mathrm{p} \wedge \mathrm{q}))$. It mentions two atomic propositions, p and q, so we have to consider 2^2 possibilities:

p	q	$\neg \mathrm{q}$	$\mathrm{p} \vee \neg \mathrm{q}$	$\mathrm{p} \wedge \mathrm{q}$	$(\mathrm{p} \vee \neg \mathrm{q}) \Rightarrow (\mathrm{p} \wedge \mathrm{q})$
T	T	F	T	T	T
T	F	T	T	F	F
F	T	F	F	F	T
F	F	T	T	F	F

Notice that this calculation is *compositional* in the structure of P.

The Binary Boolean Functions of one and two variables

$2^{(2^1)}$ functions of one variable

P	T	P	$\neg P$	F
T	T	T	F	F
F	T	F	T	F

$2^{(2^2)}$ functions of two variables

P	Q	T	$\vee$		P	$\Rightarrow$	Q	$\Leftrightarrow$	$\wedge$	nand	xor						F
T	T	T	T	T	T	T	T	T	T	F	F	F	F	F	F	F	F
T	F	T	T	T	T	F	F	F	F	T	T	T	T	F	F	F	F
F	T	T	T	F	F	T	T	F	F	T	T	F	F	T	T	F	F
F	F	T	F	T	F	T	F	T	F	T	F	T	F	T	F	T	F

(what are the complete subsets of those functions?) (why stop at 2?)

A Few More Equivalences

Identity:

$P \wedge$ T and P have the same meaning

$P \vee$ F and P have the same meaning

Complement:

$P \wedge \neg P$ and F have the same meaning

$P \vee \neg P$ and T have the same meaning

De Morgan:

$\neg(P \wedge Q)$ and $\neg P \vee \neg Q$ have the same meaning

$\neg(P \vee Q)$ and $\neg P \wedge \neg Q$ have the same meaning

Translating away $\Leftrightarrow$:

$P \Leftrightarrow Q$ and $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$ have the same meaning

Tautologies

Say P is a *tautology*, or is *valid*, if it is always true — i.e., if, whatever assumption we make about the truth values of its atomic propositions, then P is true.

When we say 'P and Q have the same meaning', we really mean 'whatever assumption we make about the truth values of their atomic propositions, P and Q have the same truth value as each other'.

We write that as P iff Q

(Strictly, this P iff Q is a meta-statement about two propositions, not itself a proposition. But P iff Q if and only if $P \Leftrightarrow Q$ is a tautology.)

Equational Reasoning

Tautologies are really useful — because they can be used anywhere.

In more detail, this P iff Q is a proper notion of equality. You can see from its definition that

- it's *reflexive*, i.e., for any P, we have P iff P
- it's *symmetric*, i.e., if P iff Q then Q iff P
- it's *transitive*, i.e., if P iff Q and Q iff R then P iff R

Moreover, if P iff Q then we can replace a subformula P by Q in any context, without affecting the meaning of the whole thing. For example, if P iff Q then $P \wedge R$ iff $Q \wedge R$, $R \wedge P$ iff $R \wedge Q$, $\neg P$ iff $\neg Q$, etc.

Equational Reasoning

Now we’re in business: we can do equational reasoning, replacing equal subformulae by equal subformulae, just as you do in normal algebraic manipulation (where you’d use $2 + 2 = 4$ without thinking).

This complements direct verification using truth tables — sometimes that’s more convenient, and sometimes this is. Later, we’ll see a third option — structured proof.

Some Collected Tautologies, for Reference

For any propositions P, Q, and R

Commutativity:

$P \land Q$ iff $Q \land P$ (and-comm)

$P \lor Q$ iff $Q \lor P$ (or-comm)

Associativity:

$P \land (Q \land R)$ iff $(P \land Q) \land R$ (and-assoc)

$P \lor (Q \lor R)$ iff $(P \lor Q) \lor R$ (or-assoc)

Distributivity:

$P \land (Q \lor R)$ iff $(P \land Q) \lor (P \land R)$ (and-or-dist)

$P \lor (Q \land R)$ iff $(P \lor Q) \land (P \lor R)$ (or-and-dist)

Identity:

$P \land \mathsf{T}$ iff P (and-id)

$P \lor \mathsf{F}$ iff P (or-id)

Unit:

$P \land \mathsf{F}$ iff F (and-unit)

$P \lor \mathsf{T}$ iff T (or-unit)

Complement:

$P \land \neg P$ iff F (and-comp)

$P \lor \neg P$ iff T (or-comp)

De Morgan:

$\neg(P \land Q)$ iff $\neg P \lor \neg Q$ (and-DM)

$\neg(P \lor Q)$ iff $\neg P \land \neg Q$ (or-DM)

Defn:

$P \Rightarrow Q$ iff $Q \lor \neg P$ (imp)

$P \Leftrightarrow Q = (P \Rightarrow Q) \land (Q \Rightarrow P)$ (bi)

Equational Reasoning — Example

Suppose we wanted to prove a 3-way De Morgan law

$\neg(P_1 \wedge P_2 \wedge P_3)$ iff $\neg P_1 \vee \neg P_2 \vee \neg P_3$

We could do so either by truth tables, checking 2^3 cases, or by equational reasoning:

$$\begin{array}{lcll} \neg(P_1 \wedge P_2 \wedge P_3) & \text{iff} & \neg(P_1 \wedge (P_2 \wedge P_3)) & \text{choosing an } \wedge \text{ association} \\ & \text{iff} & \neg P_1 \vee \neg(P_2 \wedge P_3) & \text{by (and-DM)} \end{array}$$

(and-DM) is $\neg(P \wedge Q)$ iff $\neg P \vee \neg Q$. Instantiating the metavariables P and Q as

$$\begin{array}{lcl} P & \mapsto & P_1 \\ Q & \mapsto & P_2 \wedge P_3 \end{array}$$

we get exactly the $\neg(P_1 \wedge (P_2 \wedge P_3))$ iff $\neg P_1 \vee \neg(P_2 \wedge P_3)$ needed.

$\neg(P_1 \wedge P_2 \wedge P_3)$ iff $\neg(P_1 \wedge (P_2 \wedge P_3))$ choosing an $\wedge$ association

iff $\neg P_1 \vee \neg(P_2 \wedge P_3)$ by (and-DM)

iff $\neg P_1 \vee (\neg P_2 \vee \neg P_3)$ by (and-DM)

(and-DM) is $\neg(P \wedge Q)$ iff $\neg P \vee \neg Q$. Instantiating the metavariables P and Q as

$$\begin{array}{ccc} P & \mapsto & P_2 \\ Q & \mapsto & P_3 \end{array}$$

we get $\neg(P_2 \wedge P_3)$ iff $\neg P_2 \vee \neg P_3$. Using that in the context $\neg P_1 \vee \ldots$ gives us exactly the equality $\neg P_1 \vee \neg(P_2 \wedge P_3))$ iff $\neg P_1 \vee (\neg P_2 \vee \neg P_3)$.

iff $\neg P_1 \vee \neg P_2 \vee \neg P_3$ forgetting the $\vee$ association

So by transitivity of iff, we have $\neg(P_1 \wedge P_2 \wedge P_3)$ iff $\neg P_1 \vee \neg P_2 \vee \neg P_3$

There I unpacked the steps in some detail, so you can see what’s really going on. Later, we’d normally just give the brief justification on each line; we wouldn’t write down the boxed reasoning (instantiation, context, transitivity) — but it should be clearly in your head when you’re doing a proof.

If it’s not clear, write it down — *use* the written proof as a tool for thinking.

Still later, you’ll use equalities like this one as single steps in bigger proofs.

Equational reasoning from those tautologies is *sound*: however we instantiate them, and chain them together, if we deduce that P iff Q then P iff Q.

Pragmatically important: if you've faithfully modelled some real-world situation in propositional logic, then you can do any amount of equational reasoning, and the result will be meaningful.

Is equational reasoning from those tautologies also *complete*? I.e., if P iff Q, is there an equational proof of that?

Yes (though proving completeness is beyond the scope of DM1).

Pragmatically: if P iff Q, and you systematically explore all possible candidate equational proofs, eventually you'll find one. But there are infinitely many candidates: at any point, there might be several tautologies you could try to apply, and sometimes there are infinitely many instantiations (consider T iff $P \vee \neg P$).

...so naive proof search is not a decision procedure (but sometimes you can find short proofs).

In contrast, we had a terminating algorithm for checking tautologies by truth tables (but that’s exponential in the number of propositional variables).

Satisfiability

Recall P is a *tautology*, or is *valid*, if it is always true — i.e., if, whatever assumption we make about the truth values of its atomic propositions, then P is true.

Say P is a *satisfiable* if, under *some* assumption about the truth values of its atomic propositions, P is true.

$\mathrm{p} \wedge \neg \mathrm{q}$ satisfiable (true under the assumption $\mathrm{p} \mapsto \mathsf{T}$, $q \mapsto \mathsf{F}$)

$\mathrm{p} \wedge \neg \mathrm{p}$ unsatisfiable (not true under $\mathrm{p} \mapsto \mathsf{T}$ or $\mathrm{p} \mapsto \mathsf{F}$)

P unsatisfiable iff $\neg P$ valid

Object, Meta, Meta-Meta,...

We’re taking care to distinguish the connectives of the object language (propositional logic) that we’re studying, and the informal mathematics and English that we’re using to talk about it (our meta-language).

For now, we adopt a simple discipline: the former in symbols, the latter in words.

Later, you’ll use logic to talk about logic.

Application: Combinational Circuits

Use T and F to represent high and low voltage values on a wire.

Logic gates (AND, OR, NAND, etc.) compute propositional functions of their inputs.

Notation: T, F, $\wedge$, $\vee$, $\neg$ vs 0, 1, $.$, $+$, $\overline{}$

SAT solvers: compute satisfiability of formulae with 10 000’s of propositional variables.

Predicate Logic

Predicate Logic

(or Predicate Calculus, or First-Order Logic)

Socrates is a man. All men are mortal. So Socrates is mortal.

Predicate Logic

(or Predicate Calculus, or First-Order Logic)

Socrates is a man. All men are mortal. So Socrates is mortal.

Can we formalise in propositional logic?

Write p for *Socrates is a man*

Write q for *Socrates is mortal*

$\mathrm{p} \qquad \mathrm{p} \Rightarrow \mathrm{q} \qquad \mathrm{q}$

?

Predicate Logic

Often, we want to talk about properties of things, not just atomic propositions.

All lions are fierce.

Some lions do not drink coffee.

Therefore, some fierce creatures do not drink coffee.

[Lewis Carroll, 1886]

Let x range over creatures. Write $\mathrm{L}(x)$ for 'x is a lion'. Write $\mathrm{C}(x)$ for 'x drinks coffee'. Write $\mathrm{F}(x)$ for 'x is fierce'.

$$\forall\, x.\mathrm{L}(x) \Rightarrow \mathrm{F}(x)$$

$$\exists\, x.\mathrm{L}(x) \wedge \neg\mathrm{C}(x)$$

$$\exists\, x.\mathrm{F}(x) \wedge \neg\mathrm{C}(x)$$

Predicate Logic

So, we extend the language.

Variables x, y, etc., ranging over some specified domain.

Atomic predicates $\mathrm{A}(x)$, $\mathrm{B}(x)$, etc., like the earlier atomic propositions, but with truth values that depend on the values of the variables. Write $A(x)$ for an arbitrary atomic predicate. E.g.:

> Let $\mathrm{A}(x)$ denote $x + 7 = 10$, where x ranges over the natural numbers. $\mathrm{A}(x)$ is true if $x = 3$, otherwise false, so $\mathrm{A}(3) \wedge \neg A(4)$
>
> Let $\mathrm{B}(n)$ denote $1 + 2 + \ldots + n = n(n+1)/2$, where n ranges over the naturals. $\mathrm{B}(n)$ is true for any value of n, so $\mathrm{B}(27)$.

Add these to the language of formulae:

$$P, Q ::= A(x) \mid \mathsf{T} \mid \mathsf{F} \mid \neg P \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid P \Leftrightarrow Q$$

Predicate Logic — Universal Quantifiers

If P is a formula, then $\forall\ x.P$ is a formula

Pronounce $\forall\ x.P$ as ‘for all x, P’.

Definition: $\forall\ x.P$ is true if (and only if) P is true for all values of x (taken from its specified domain).

Sometimes we write $P(x)$ for a formula that might mention x, so that we can write (e.g.) $P(27)$ for the formula with x instantiated to 27.

Then, if x is ranging over the naturals,
$\forall\ x.P(x)$ if and only if $P(0)$ and $P(1)$ and $P(2)$ and ...

Or, if x is ranging over $\{\mathrm{red}, \mathrm{green}, \mathrm{blue}\}$,then
$(\forall\ x.P(x))$ iff $P(\mathrm{red}) \wedge P(\mathrm{green}) \wedge P(\mathrm{blue})$.

Predicate Logic — Existential Quantifiers

If P is a formula, then $\exists\ x.P$ is a formula

Pronounce $\exists\ x.P$ as ‘exists x such that P’.

Definition: $\exists\ x.P$ is true if (and only if) there is at least one value of x (taken from its specified domain) such that P is true.

So, if x is ranging over $\{\mathrm{red}, \mathrm{green}, \mathrm{blue}\}$, then $(\exists\ x.P(x))$ iff $P(\mathrm{red}) \vee P(\mathrm{green}) \vee P(\mathrm{blue})$.

Because the domain might be infinite, we don’t give truth-table definitions for $\forall$ and $\exists$.

Note also that we don’t allow infinitary formulae — I carefully *didn’t* write $(\forall\ x.P(x))$ iff $P(0) \wedge P(1) \wedge P(2) \wedge \ldots$ ×

The Language of Predicate Logic

Summarising, the formulae of predicate logic are the terms of the grammar

$$\begin{aligned} P, Q \quad ::= \quad & A(x) \mid \mathsf{T} \mid \mathsf{F} \mid \neg P \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid \\ & P \Leftrightarrow Q \mid \forall\, x.P \mid \exists\, x.P \end{aligned}$$

Convention: the scope of a quantifier extends as far to the right as possible, so (e.g.) $\forall\, x.\mathrm{A}(x) \wedge \mathrm{B}(x)$ is $\forall x.(\mathrm{A}(x) \wedge \mathrm{B}(x))$, not $(\forall\, x.\mathrm{A}(x)) \wedge \mathrm{B}(x)$.

(other convention — no dot, always parenthesise: $\forall\, x(P)$)

Predicate Logic — Extensions

n-ary atomic predicates $\mathrm{A}(x, y)$, $\mathrm{B}(x, y, z)$,...

(regard our old p, q, etc. as 0-ary atomic predicates)

Equality as a special binary predicate $(e = e')$ where e and e' are some mathematical expressions (that might mention variables such as x), and similarly for $<, >, \leq, \geq$ over numbers.

$(e \neq e')$ iff $\neg(e = e')$

$(e \leq e')$ iff $(e < e') \vee (e = e')$

Predicate Logic — Examples

What do these mean? Are they true or false?

$\exists\, x.(x^2 + 2x + 1 = 0)$ where x ranges over the integers

$\forall\, x.(x < 0) \vee (x = 0) \vee (x \geq 0)$ where x ranges over the reals

$\forall\, x.(x \geq 0) \Rightarrow (2x > x)$ where x ranges over the reals

Predicate Logic — Examples

Formalise:

If someone learns discrete mathematics, then they will find a good job. (*)

Let x range over all people.

Write $\mathrm{L}(x)$ to mean 'x learns discrete mathematics'

Write $\mathrm{J}(x)$ to mean 'x will find a good job'

Then $\forall\ x.\mathrm{L}(x) \Rightarrow \mathrm{J}(x)$ is a reasonable formalisation of (*).

Is it true? We'd need to know more...

Predicate Logic — Nested Quantifers

What do these mean? Are they true?

$\forall\, x.\forall\, y.(x + y = y + x)$ where x, y range over the integers

$\forall\, x.\exists\, y.(x = y - 10)$ where x, y range over the integers

$\exists\, x.\forall\, y.(x \geq y)$ where x, y range over the integers

$\forall\, y.\exists\, x.(x \geq y)$ where x, y range over the integers

$\exists\, x.\exists\, y.(4x = 2y) \wedge (x + 1 = y)$ where x, y range over the integers

Predicate Logic — Examples

Formalise:

Every real number except 0 has a multiplicative inverse

$\forall\ x.(\neg(x = 0)) \Rightarrow \exists\ y.(x\ \ y = 1)$ where x ranges over the reals

Predicate Logic — Examples

Formalise:

Everyone has exactly one best friend.

Let x, y, z range over all people.

Write $\mathrm{B}(x, y)$ to mean y is a best friend of x

Then $\forall\ x.\exists\ y.\mathrm{B}(x, y) \wedge \forall\ z.\mathrm{B}(x, z) \Rightarrow z = y$ is one reasonable formalisation.

Equivalently $\forall\ x.\exists\ y.\mathrm{B}(x, y) \wedge \forall\ z.(\neg(z = y)) \Rightarrow \neg\mathrm{B}(x, z)$.

Um. what about $y = x$?

Predicate Logic — Basic Properties

De Morgan laws for quantifiers:

$(\neg\forall\, x.P)$ iff $\exists\, x.\neg P$

$(\neg\exists\, x.P)$ iff $\forall\, x.\neg P$

Distributing quantifiers over $\wedge$ and $\vee$:

$(\forall\, x.P \wedge Q)$ iff $(\forall\, x.P) \wedge (\forall\, x.Q)$

$(\exists\, x.P \wedge Q)$ $\not\text{iff}$ $(\exists\, x.P) \wedge (\exists\, x.Q)$ × (left-to-right holds)

$(\forall\, x.P \vee Q)$ $\not\text{iff}$ $(\forall\, x.P) \vee (\forall\, x.Q)$ × (right-to-left holds)

$(\exists\, x.P \vee Q)$ iff $(\exists\, x.P) \vee (\exists\, x.Q)$

Predicate Logic — Free and Bound Variables

A slightly odd (but well-formed) formula:

$$\mathrm{A}(x) \wedge (\forall\, x.\mathrm{B}(x) \Rightarrow \exists\, x.\mathrm{C}(x, x))$$

Really there are 3 different x's here, and it'd be clearer to write

$\mathrm{A}(x) \wedge (\forall\, x'.\mathrm{B}(x') \Rightarrow \exists\, x''.\mathrm{C}(x'', x''))$ or

$\mathrm{A}(x) \wedge (\forall\, y.\mathrm{B}(y) \Rightarrow \exists\, z.\mathrm{C}(z, z))$

Say an occurrence of x in a formula P is *free* if it is not inside any $(\forall\, x....)$ or $(\exists\, x....)$

All the other occurrences of x are *bound* by the closest enclosing $(\forall\, x....)$ or $(\exists\, x....)$

The *scope* of a quantifier in a formula $...(\forall\, x.P)...$ is all of P (except any subformulae of P of the form $\forall\, x....$ or $\exists\, x....$).

Truth Semantics

Whether a formula P is true or false might depend on

1. an interpretation of the atomic predicate symbols used in P (generalising the ‘assumptions on its atomic propositions’ we had before)

2. the values of the free variables of P

Often 1 is fixed (as it is for $e = e'$)

Application: Databases

Proof

Proof

We've now got a rich enough language to express some non-trivial conjectures, e.g.

$$\forall\ n.(n \geq 2) \Rightarrow \neg\exists\ x, y, z.x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \wedge x^n + y^n = z^n$$

(where n ranges over the naturals)

Is that true or false?

Proof

$$\forall\, n.(n \geq 2) \Rightarrow \neg\exists\, x, y.x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \wedge x^n + y^n = z^n$$

We have to be able to reason about this kind of thing, to *prove* that it’s true (or to disprove it — to prove its negation...).

This course: ‘informal’ rigorous proof (normal mathematical practice). A proof is a rigorous argument to convince a very skeptical reader. It should be completely clear, and the individual steps small enough that there’s no question about them.

(Later, study ‘formal’ proofs, as mathematical objects themselves...)

Non-Proofs

There are *lots*.

‘I have discovered a truly remarkable proof which this margin is too small to contain.’

‘I’m your lecturer, and I say it’s true’

‘The world would be a sad place if this wasn’t true’

‘I can’t imagine that it could be false’

Statements

Theorem 1 [associativity of $+$] $\forall\ x, y, z. x + (y + z) = (x + y) + z$

Often leave top-level universal quantifiers implicit (but only in these top-level statements):

Theorem 2 $x + (y + z) = (x + y) + z$

Proposition — a little theorem

Lemma — a little theorem written down as part of a bigger proof

Corollary — an easy consequence of some theorem

any of those should come with a proof attached

Conjecture $x \mod 2 = 0 \vee x \mod 3 = 0 \vee x \mod 5 = 0$

Structured Proof

The truth-table and equational reasoning from before is still sound, but we need more, to reason about the quantifiers. And truth tables aren’t going to help there.

Going to focus instead on the structure of the formulae we’re trying to prove (and of those we can use).

Practice on statements about numbers — not that we care about these results particularly, but just to get started.

Example

Theorem? The sum of two rationals is rational.

Example

Theorem? The sum of two rationals is rational.

Clarify the logical form:

Theorem? $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$

Theorem? The sum of two rationals is rational.

Clarify the logical form:

Theorem?

$$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$$

and the definitions:

Say $\mathrm{Rational}(x)$ iff $\exists\, n, m.(x = n/m)$

where x and y range over real numbers and n and m range over integers.

Sometimes this clarification is a major intellectual activity (and the subsequent proof might be easy); sometimes it's easy to state the problem (but the proof is very hard).

How *far* we have to clarify the definitions depends on the problem — here I didn't define the reals, integers, addition, or division.

$$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$$

1. Consider an arbitrary real x

now we aim to prove $\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$

$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x
2. Consider an arbitrary real y

now we aim to prove $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x
2. Consider an arbitrary real y
3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\, x. \forall\, y. (\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x
2. Consider an arbitrary real y
3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$
4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x
2. Consider an arbitrary real y
3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$
4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination
5. $\mathrm{Rational}(y)$ from 3 by $\wedge$-elimination

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x
2. Consider an arbitrary real y
3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$
4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination
5. $\mathrm{Rational}(y)$ from 3 by $\wedge$-elimination
6. $\exists\ n, m.(x = n/m)$ from 4 by unfolding the defn of $\mathrm{Rational}$
7. $\exists\ n, m.(y = n/m)$ from 5 by unfolding the defn of $\mathrm{Rational}$

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x

2. Consider an arbitrary real y

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination

5. $\mathrm{Rational}(y)$ from 3 by $\wedge$-elimination

6. $\exists\ n, m.(x = n/m)$ from 4 by unfolding the defn of $\mathrm{Rational}$

7. $\exists\ n, m.(y = n/m)$ from 5 by unfolding the defn of $\mathrm{Rational}$

8. For some actual (integer) n_1 and m_1, $x = n_1/m_1$

   from 6 by $\exists$-elimination

9. For some actual (integer) n_2 and m_2, $y = n_2/m_2$

   from 7 by $\exists$-elimination

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x

2. Consider an arbitrary real y

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

...

8. For some actual (integer) n_1 and m_1, $x = n_1/m_1$

from 6 by $\exists$-elimination

9. For some actual (integer) n_2 and m_2, $y = n_2/m_2$

from 7 by $\exists$-elimination

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $= \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $= \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x

2. Consider an arbitrary real y

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

...

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $\qquad = \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $\qquad = \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

13. $\exists\ n, m. x+y = n/m$ from 10–12, witness $n = n_1\ m_2 + m_1\ n_2$

$m = m_1\ m_2$

now we aim to prove $\mathrm{Rational}(x+y)$

$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$

1. Consider an arbitrary real x

2. Consider an arbitrary real y

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

...

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $= \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $= \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

13. $\exists\ n, m.x + y = n/m$ from 10–12, witness $n = n_1\ m_2 + m_1\ n_2$

$m = m_1\ m_2$

14. $\mathrm{Rational}(x + y)$ from 13, folding the defn of $\mathrm{Rational}$

now we aim to prove $\mathrm{Rational}(x + y)$ — but we have! so:

$\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x

2. Consider an arbitrary real y

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$

...

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $= \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $= \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

13. $\exists\ n, m.x+y = n/m$ from 10–12, witness $n = n_1\ m_2 + m_1\ n_2$

$m = m_1\ m_2$

14. $\mathrm{Rational}(x+y)$ from 13, folding the defn of $\mathrm{Rational}$

15. $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$ by $\Rightarrow$-I, 3–14

now we aim to prove $\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

$\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$

1. Consider an arbitrary real x.

2. Consider an arbitrary real y.

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$.

4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination

5. $\mathrm{Rational}(y)$ from 3 by $\wedge$-elimination

6. $\exists\, n, m.(x = n/m)$ from 4 by unfolding the defn of $\mathrm{Rational}$

7. $\exists\, n, m.(y = n/m)$ from 5 by unfolding the defn of $\mathrm{Rational}$

8. For some actual (integer) n_1 and m_1, $x = n_1/m_1$ from 6 by $\exists$-elimination

9. For some actual (integer) n_2 and m_2, $y = n_2/m_2$ from 7 by $\exists$-elimination

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $= \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $= \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

13. $\exists\, n, m.x + y = n/m$ from 10–12, witness $n = n_1\ m_2 + m_1\ n_2$

$m = m_1\ m_2$

14. $\mathrm{Rational}(x+y)$ from 13, folding the defn of $\mathrm{Rational}$

15. $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$ by $\Rightarrow$-introduction, from 3–14

16. $\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$ by $\forall$-introduction, from 2–15

17. $\forall\, x.\forall\, y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x+y)$ by $\forall$-introduction, from 1–16

Theorem $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$

Proof

1. Consider an arbitrary real x.

2. Consider an arbitrary real y.

3. Assume $\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)$.

4. $\mathrm{Rational}(x)$ from 3 by $\wedge$-elimination

5. $\mathrm{Rational}(y)$ from 3 by $\wedge$-elimination

6. $\exists\ n, m.(x = n/m)$ from 4 by unfolding the defn of $\mathrm{Rational}$

7. $\exists\ n, m.(y = n/m)$ from 5 by unfolding the defn of $\mathrm{Rational}$

8. For some actual (integer) n_1 and m_1, $x = n_1/m_1$ from 6 by $\exists$-elimination

9. For some actual (integer) n_2 and m_2, $y = n_2/m_2$ from 7 by $\exists$-elimination

10. $x + y = (n_1/m_1) + (n_2/m_2)$ from 8 and 9, adding both sides

11. $= \frac{n_1\ m_2}{m_1\ m_2} + \frac{m_1\ n_2}{m_1\ m_2}$ from 10, by arithmetic

12. $= \frac{n_1\ m_2 + m_1\ n_2}{m_1\ m_2}$ from 11, by arithmetic

13. $\exists\ n, m.x + y = n/m$ from 10–12, witness $n = n_1\ m_2 + m_1\ n_2$

$m = m_1\ m_2$

14. $\mathrm{Rational}(x + y)$ from 13, folding the defn of $\mathrm{Rational}$

15. $(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$ by $\Rightarrow$-introduction, from 3–14

16. $\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$ by $\forall$-introduction, from 2–15

17. $\forall\ x.\forall\ y.(\mathrm{Rational}(x) \wedge \mathrm{Rational}(y)) \Rightarrow \mathrm{Rational}(x + y)$ by $\forall$-introduction, from 1–16 □

What is a Proof (in this stylised form)?

A list of lines, each of which is either:

- a formula of predicate logic, with a justification (‘P, from ... by ...’)
- an assumption of some formula (‘Assume P’)
- an introduction of a arbitrary variable (‘Consider an arbitrary x (from the appropriate domain)’)
- an introduction of some actual witness variables and a formula (‘For some actual n, P’)

When we make an assumption, we open a box. We have to close it before we can discharge the assumption (by $\Rightarrow$-introduction at step 15).

(Actually also for introductions of arbitrary and witness variables. But if these are just at the top level, and we do $\forall$-introduction on them at the end, we might not draw them.)

What is a Proof (in this stylised form)?

Lines are numbered

Introduced variables must be *fresh* (not free in any preceeding formula).

The justifications must not refer to later lines (no circular proofs, please!)

1. P by ... from 15 ×

...

15. Q by ... from 1

What is a Proof (in this stylised form)?

The justifications must not refer to lines inside any earlier box

1. Assume P

...

15. U from ... by ...

...

27. Q from ... by ...

28. $P \Rightarrow Q$ by $\Rightarrow$-introduction, from 1–27

29. Assume R

...

1007. ... from 15 by ... ×

(earlier in an enclosing box is ok)

What is a Justification (in this stylised form)?

Back to the Connectives — And

To use a conjunction: if we know $P \wedge Q$, then we can deduce P, or we can deduce Q (or both, as often as we like)

...

m. $P \wedge Q$ from ...

...

n. P from m by $\wedge$-elimination

or

...

m. $P \wedge Q$ from ...

...

n. Q from m by $\wedge$-elimination

What is a Justification (in this stylised form)?

Back to the Connectives — And

To prove a conjunction: we can prove $P \wedge Q$ by proving P and proving Q.

...

l. P from ...

...

m. Q from ...

...

n. $P \wedge Q$ from l and m by $\wedge$-introduction

(it doesn’t matter in what order l and m are in)

What is a Justification (in this stylised form)?

Back to the Connectives — Or

To prove a disjunction: to prove $P \vee Q$, we could prove P, or we could prove Q. (could even use $\neg Q$ or $\neg P$ resp.)

...

m. P from ...

...

n. $P \vee Q$ from m by $\vee$-introduction

or

...

m. Q from ...

...

n. $P \vee Q$ from m by $\vee$-introduction

What is a Justification (in this stylised form)?

Back to the Connectives — Or

To use a disjunction: if we know $P \vee Q$, and by assuming P we can prove R, and by assuming Q we can prove R, then we can deduce R (a form of case analysis).

l. $P \vee Q$ from ... by ...

...

m_1. Assume P

...

m_2. R

...

n_1. Assume Q

...

n_2. R

...

o. R from l, m_1–m_2, n_1–n_2 by $\vee$-elimination

(it doesn't matter what order l, m_1–m_2, and n_1–n_2 are in)

What is a Justification (in this stylised form)?

Back to the Connectives — Implication

To prove an implication: to prove $P \Rightarrow Q$, assume P, prove Q, and discharge the assumption.

...

m. Assume P

...

n. Q from ... by ...

$n + 1$. $P \Rightarrow Q$ from m–n, by $\Rightarrow$-introduction

What is a Justification (in this stylised form)?

Back to the Connectives — Implication

To use an implication: if we know $P \Rightarrow Q$, and we know P, we can deduce Q

...

l. $P \Rightarrow Q$ by ...

...

m. P by ...

...

n. Q from l and m by $\Rightarrow$-elimination

(also known as *modus ponens*)

What is a Justification (in this stylised form)?

Back to the Connectives — Negation

To prove a negation: to prove $\neg P$, assume P, prove F, and discharge the assumption.

...

m. Assume P

...

n. F from ... by ...

$n+1$. $\neg P$ from m–n, by $\neg$-introduction

That's a lot like $\Rightarrow$-introduction (not a surprise, as $\neg P$ iff $(P \Rightarrow \mathsf{F})$).

What is a Justification (in this stylised form)?

Back to the Connectives — Negation

To use a negation: if we know $\neg P$, and we know P, we can deduce F

...

l. P by ...

...

m. $\neg P$ by ...

...

n. F from l and m by $\neg$-elimination

What is a Justification (in this stylised form)?

Back to the Connectives — Truth

To prove T: nothing to do

...

n. T

That’s not very useful, though... because:

To use T: you can’t do anything with it.

What is a Justification (in this stylised form)?

Contradiction

To prove P by contradiction: if, from assuming $\neg P$, we can prove F, then we can deduce P

...

m. Assume $\neg P$

...

n. F from ... by ...

$n+1$. P from m–n, by contradiction

Note that in the other rules either a premise (for elimination rules) or the conclusion (for introduction rules) had some particular form, but here the conclusion is an arbitrary P.

What is a Justification (in this stylised form)?

Contradiction$'$

To prove P by contradiction: if we can deduce F, then we can deduce any P

...

m. F from ... by ...

...

n. P from m, by contradiction

(hopefully this would be under some assumption(s)...)

Example

Theorem $(P \wedge Q) \Rightarrow (P \vee Q)$

Proof:

1. Assume $P \wedge Q$
2. P from 1 by $\wedge$-elim
3. $P \vee Q$ from 2 by $\vee$-intro
4. $(P \wedge Q) \Rightarrow (P \vee Q)$ from 1–3 by $\Rightarrow$-intro

□

Example

Theorem ? $(P \vee Q) \Rightarrow (P \wedge Q)$

Proof ?:

1. Assume $P \vee Q$

2. ...use $\vee$-elim somehow? prove by contradiction?

????

$n-2.$ P from ? by ?

$n-1.$ Q from ? by ?

$n.$ $(P \wedge Q)$ from $n-1$, $n-2$ by $\wedge$-intro

$n+1.$ $(P \vee Q) \Rightarrow (P \wedge Q)$ from 1–n by $\Rightarrow$-intro

Counterexample? Prove negation?

What is a Justification (in this stylised form)?

Back to the Connectives — For all

To use a universally quantified formula: if we know $\forall x.P(x)$, then we can deduce $P(v)$ for any v (of the appropriate domain)

...

m. $\forall\ x.P(x)$ from ...

...

n. $P(v)$ from m by $\forall$-elimination

What is a Justification (in this stylised form)?

Back to the Connectives — For all

To prove a universally quantified formula $\forall\ x.P(x)$, consider an arbitrary fresh variable x (ranging over the appropriate domain) and prove $P(x)$, then discharge the assumption.

...

m. Consider an arbitrary x (from domain ...)

...

n. $P(x)$ by ...

$n + 1$. $\forall\ x.P(x)$ from m–n by $\forall$-introduction

What is a Justification (in this stylised form)?

Back to the Connectives — Exists

To prove an existentially quantified formula $\exists\ x.P(x)$, prove $P(v)$ for some witness v (from the appropriate domain).

...

m. $P(v)$

...

n. $\exists\ x.P(x)$ from m by $\exists$-introduction with witness $x = v$

What is a Justification (in this stylised form)?

Back to the Connectives — Exists

To use an existentially quantified formula $\exists\, x.P(x)$, introduce a fresh variable (ranging over the appropriate domain) x_1, about which we know only $P(x_1)$

...

m. $\exists\, x.P(x)$

...

n. For some actual x_1, $P(x_1)$ from m by $\exists$-elimination

That’s a special case of this more general rule:

l. $\exists\, x.P(x)$

...

m. For some actual x_1, $P(x_1)$

...

n. Q (where x_1 not free in Q)

...

o. Q from l, m–n, by $\exists$-elimination

Example

Many theorems have a similar top-level structure, e.g.

$\forall\, x, y, z.(P \wedge Q \wedge R) \Rightarrow S$

1. Consider an arbitrary x, y, z.

2. Assume $P \wedge Q \wedge R$.

3. P from 2 by $\wedge$-elimination

4. Q from 2 by $\wedge$-elimination

5. R from 2 by $\wedge$-elimination

...

215. S by ...

216. $(P \wedge Q \wedge R) \Rightarrow S$ from 2–215 by $\Rightarrow$-introduction

217. $\forall\, x, y, z.(P \wedge Q \wedge R) \Rightarrow S$ by $\forall$-introduction, from 1–216

What is a Proof (in this stylised form)?

NB This particular stylised form is only one way to write down rigorous paper proofs. It's a good place to start, but its not always appropriate. Later, you'll sometimes take bigger steps, and won't draw the boxes.

But however they are written, they *have* to be written down clearly — a proof is a communication tool, to persuade. Each step needs a justification.

In questions, we'll say specifically "by structured proof", "by equational reasoning", "by truth tables", or, more generally "prove".

(This is basically the 'box and line' proofs from Bornat 2005, which are a linear notation for *natural deduction* proofs. More on that in 1B Logic & Proof. If you want to try mechanised proofs, see `jape.org.uk` or `prover.cs.ru.nl` (experiments! — let me know how it goes))

Soundness and Completeness?

Are these proof rules *sound*? (i.e., are all the provable formulae valid?)

Are these proof rules *complete*? (i.e., are all valid formulae provable?)

Think about *proof search*

Aside: Writing Discrete Maths

By hand

In ASCII

```
P ::= T | F | p | A(x) | P /\ Q | P \/ Q
    | P=>Q | P<=>Q | !x.P | ?x.P
```

In LaTeX (but don’t forget that typesetting is *not* real work)

Pragmatics

Given some conjecture:

1. Ensure the statement is well-defined, and that you know the definitions of whatever it uses.
2. Understand intuitive what it’s saying. Verbalize it.
3. Intuitively, why is it true? (or false?)
4. What are the hard (or easy) cases likely to be?
5. Choose a strategy — truth tables, equational reasoning, structured proof, induction, ...
6. Try it! (but be prepared to backtrack)
7. Expand definitions and make abbreviations as you need them.
8. Writing — to communicate, and to help you think.

9. Choose variable names carefully; take care with parentheses

10. Use enough words and use enough symbols, but keep them properly nested. Don't use random squiggles ("$\Rightarrow$" or "$\because$") for meta-reasoning.

11. If it hasn't worked yet... either

    (a) you've make some local mistake (mis-instantiated, re-used a variable name, not expanded definitions enough, forgotten a useful assumption). Fix it and continue.

    (b) you've found that the conjecture is false. Construct a simple counterexample and check it.

    (c) you need to try a different strategy (different induction principle, strengthened induction hypothesis, proof by contradictions,...)

    (d) you didn't really understand intuitively what the conjecture is saying, or what the definitions it uses mean. Go back to them again.

12. If it has worked: read through it, skeptically. Maybe *re-write* it.

13. Finally, give it to someone else, as skeptical and careful as you can find, to see if they believe it — to see if they believe that *what you've written down is a proof*, not that they believe that *the conjecture is true*.

...more fallacies

Set Theory

Set Theory

Now we've got some reasoning techniques, but not much to reason *about*.

Let's add *sets* to our language.

What is a set? An unordered collection of *elements*:

$$\{0, 3, 7\} = \{3, 0, 7\}$$

might be empty:

$$\{\} = \emptyset = \varnothing$$

might be infinite:

$$\begin{aligned} \mathbb{N} &= \{0, 1, 2, 3...\} \\ \mathbb{Z} &= \{..., -1, 0, 1, ...\} \\ \mathbb{R} &= \text{...all the real numbers} \end{aligned}$$

Some more interesting sets

the set of nodes in a network (encode with $\mathbb{N}$?)

the set of paths between such nodes (encode ??)

the set of polynomial-time computable functions from naturals to naturals

the set of well-typed programs in some programming language (encode???)

the set of executions of such programs

the set of formulae of predicate logic

the set of valid proofs of such formulae

the set of all students in this room (?)

the set of all sets ✗

Basic relationships

membership $x \in A$

$3 \in \{1, 3, 5\}$

$2 \notin \{1, 3, 5\}$

(of course $(2 \notin \{1, 3, 5\})$ iff $\neg(2 \in \{3, 5, 1\})$)

equality between sets $A = \mathrm{B}$ iff $\forall\, x . x \in A \Leftrightarrow x \in \mathrm{B}$

$\{1, 2\} = \{2, 1\} = \{2, 1, 2, 2\} \qquad \{\} \neq \{\{\}\}$

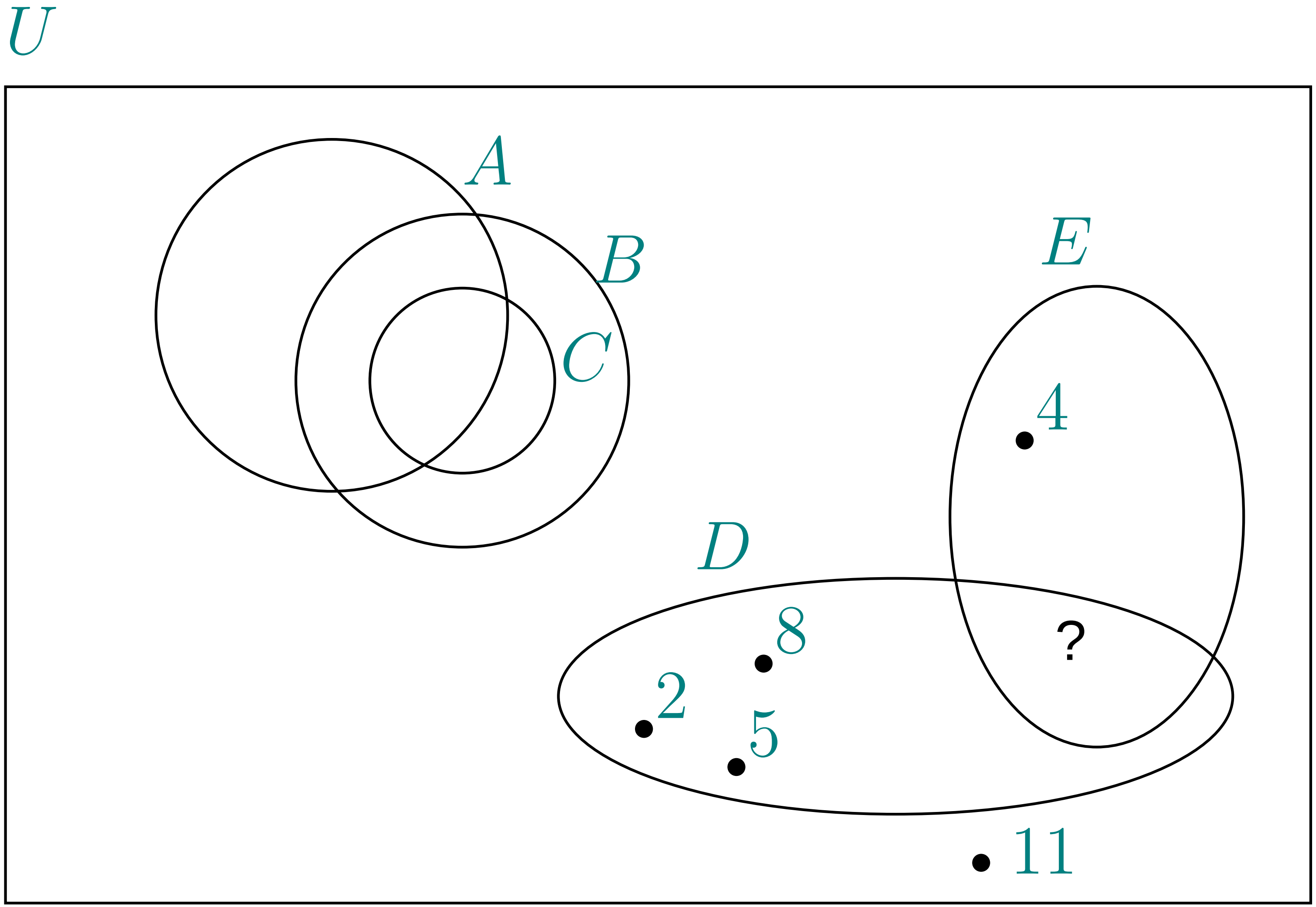
inclusion or *subset* $A \subseteq \mathrm{B}$ iff $\forall\, x . x \in A \Rightarrow x \in \mathrm{B}$

Properties: $\subseteq$ is reflexive, transitive,

and antisymmetric ($(A \subseteq \mathrm{B} \wedge \mathrm{B} \subseteq A) \Rightarrow A = \mathrm{B}$)

but not total: $\{1, 2\} \not\subseteq \{1, 3\} \not\subseteq \{1, 2\}$

Venn Diagrams



Bounded Quantifiers

Write

$\forall\ x \in A . P$ for $\forall\ x . x \in A \Rightarrow P$

$\exists\ x \in A . P$ for $\exists\ x . x \in A \wedge P$

where A is a subset of the domain that x ranges over.

Define Even to be the set of all even naturals

Then can write $\forall\ n \in \mathrm{Even} . \exists\ m \in \mathbb{N} . n = 3m$

Building interesting subsets with set comprehension

$$\text{Even} \overset{\text{def}}{=} \{n \mid \exists\ m \in \mathbb{N}. n = 2m\}$$

$$\{x \mid x \in \mathbb{N} \wedge \neg\exists\ y, z \in \mathbb{N}. y > 1 \wedge z > 1 \wedge y\ z = x\}$$

$$\{x \mid x \in \mathbb{N} \wedge \forall\ y \in \mathbb{N}. y > x\}$$

$$\{2\ x \mid x \in \mathbb{N}\}$$

From sets to predicates, and back again

From sets to predicates: given a set A, can define a predicate

$$P(x) \overset{\mathrm{def}}{=} x \in A$$

From predicates to sets: given $P(x)$ and some set U, can build a set

$$A \overset{\mathrm{def}}{=} \{x \mid x \in U \wedge P(x)\}$$

(in some logics we'd really identify the two concepts – but not here)

Property of comprehensions: $x \in \{y \mid P(y)\}$ iff $P(x)$

Building new sets from old ones: union, intersection, and difference

$$A \cup \mathrm{B} \overset{\text{def}}{=} \{x \mid x \in A \vee x \in \mathrm{B}\}$$

$$A \cap \mathrm{B} \overset{\text{def}}{=} \{x \mid x \in A \wedge x \in \mathrm{B}\}$$

$$A - \mathrm{B} \overset{\text{def}}{=} \{x \mid x \in A \wedge x \notin \mathrm{B}\}$$

A and B are *disjoint* iff $A \cap \mathrm{B} = \{\}$ (symm, not refl or tran)

Building new sets from old ones: union, intersection, and difference

$$\{1,2\} \cup \{2,3\} = \{1,2,3\}$$

$$\{1,2\} \cap \{2,3\} = \{2\}$$

$$\{1,2\} - \{2,3\} = \{1\}$$

Properties of union, intersection, and difference

Recall $\vee$ is associative: $P \vee (Q \vee R)$ iff $(P \vee Q) \vee R$

Theorem $A \cup (\mathrm{B} \cup \mathrm{C}) = (A \cup \mathrm{B}) \cup \mathrm{C}$

Proof

$A \cup (\mathrm{B} \cup \mathrm{C})$

1. $= \{x \mid x \in A \vee x \in (\mathrm{B} \cup \mathrm{C})\}$ unfold defn of union
2. $= \{x \mid x \in A \vee x \in \{y \mid y \in \mathrm{B} \vee y \in \mathrm{C}\}\}$ unfold defn of union
3. $= \{x \mid x \in A \vee (y \in \mathrm{B} \vee y \in \mathrm{C})\}$ comprehension property
4. $= \{x \mid (x \in A \vee y \in \mathrm{B}) \vee y \in \mathrm{C}\}$ by $\vee$ assoc
5. $= (A \cup \mathrm{B}) \cup \mathrm{C}$ by the comprehension property and folding defn of union twice □

Some Collected Set Equalities, for Reference

For any sets A, B, and C, all subsets of U

Commutativity:

$A \cap \mathrm{B} = \mathrm{B} \cap A$ (∩-comm)

$A \cup \mathrm{B} = \mathrm{B} \cup A$ (∪-comm)

Associativity:

$A \cap (\mathrm{B} \cap \mathrm{C}) = (A \cap \mathrm{B}) \cap \mathrm{C}$ (∩-assoc)

$A \cup (\mathrm{B} \cup \mathrm{C}) = (A \cup \mathrm{B}) \cup \mathrm{C}$ (∪-assoc)

Distributivity:

$A \cap (\mathrm{B} \cup \mathrm{C}) = (A \cap \mathrm{B}) \cup (A \cap \mathrm{C})$ (∩-∪-dist)

$A \cup (\mathrm{B} \cap \mathrm{C}) = (A \cup \mathrm{B}) \cap (A \cup \mathrm{C})$ (∪-∩-dist)

Identity:

$A \cap U = A$ (∩-id)

$A \cup \{\} = A$ (∪-id)

Unit:

$A \cap \{\} = \{\}$ (∩-unit)

$A \cup U = U$ (∪-unit)

Complement:

$A \cap (U - A) = \{\}$ (∩-comp)

$A \cup (U - A) = U$ (∪-comp)

De Morgan:

$U - (A \cap \mathrm{B}) = (U - A) \cup (U - \mathrm{B})$ (∩-DM)

$U - (A \cup \mathrm{B}) = (U - A) \cap (U - \mathrm{B})$ (∪-DM)

Example Proof

Theorem $\{\} \subseteq A$

Proof

$$\{\} \subseteq A$$

1. iff $\forall x . x \in \{\} \Rightarrow x \in A$ unfolding defn of $\subseteq$
2. iff $\forall x .$F $\Rightarrow x \in A$ use defn of $\in$
3. iff $\forall x .$T equational reasoning with (F $\Rightarrow P$) iff T
4. iff T using defn of $\forall$ □

Another Proof of the Same Theorem

Theorem $\{\} \subseteq A$

Another Proof (using the structured rules more explicitly)

1. $\{\} \subseteq A$ iff $\forall x . x \in \{\} \Rightarrow x \in A$ unfolding defn of $\subseteq$

We prove the r.h.s.:

> 2\. Consider an arbitrary x
>
> > 3\. Assume $x \in \{\}$
> >
> > 4\. F by defn of $\in$
> >
> > 5\. $x \in A$ from 4, by contradiction
>
> 6\. $x \in \{\} \Rightarrow x \in A$ from 3–5, by $\Rightarrow$-introduction

7. $\forall x . x \in \{\} \Rightarrow x \in A$ from 2–6, by $\forall$-introduction □

Building new sets from old ones: powerset

Write $\mathcal{P}(A)$ for the set of all subsets of a set A.

$$\mathcal{P}\{\} = \{\{\}\}$$

$$\mathcal{P}\{7\} = \{\{\}, \{7\}\}$$

$$\mathcal{P}\{1, 2\} = \{\{\}, \{1\}, \{2\}, \{1, 2\}\}$$

$$A \in \mathcal{P}(A)$$

(why ‘power’ set?)

Building new sets from old ones: product

Write (a, b) (or sometimes $\langle a, b\rangle$) for an ordered pair of a and b

$$A \times \mathrm{B} \overset{\text{def}}{=} \{(a, b) \mid a \in A \wedge b \in \mathrm{B}\}$$

Similarly for triples $(a, b, c) \in A \times \mathrm{B} \times \mathrm{C}$ etc.

Pairing is *non-commutative*: $(a, b) \neq (b, a)$ unless $a = b$

Pairing is *non-associative* and distinct from 3-tupling etc:

$(a, (b, c)) \neq (a, b, c) \neq ((a, b), c)$ and

$A \times (\mathrm{B} \times \mathrm{C}) \neq A \times \mathrm{B} \times \mathrm{C} \neq (A \times \mathrm{B}) \times \mathrm{C}$

Why ‘product’?

$$\{1, 2\} \times \{\mathrm{red}, \mathrm{green}\} = \{(1, \mathrm{red}), (2, \mathrm{red}), (1, \mathrm{green}), (2, \mathrm{green})\}$$

We know $(a, b) = (b, a) \Rightarrow a = b$ for pairs

so why not lift the result to set product?

Theorem ? $(A \times \mathrm{B} = \mathrm{B} \times A) \Rightarrow A = \mathrm{B}$

Proof?

The first components of the pairs in $A \times \mathrm{B}$ are from A.

The first components of the pairs in $\mathrm{B} \times A$ are from B.

If $A \times \mathrm{B} = \mathrm{B} \times A$ then these must be the same, so $A = \mathrm{B}$.

Theorem ? $(A \times \mathrm{B} = \mathrm{B} \times A) \Rightarrow A = \mathrm{B}$

Proof?

1. Assume $A \times \mathrm{B} = \mathrm{B} \times A$

We prove $A = \mathrm{B}$, i.e. $\forall\, x . x \in A \Leftrightarrow x \in \mathrm{B}$

> 2. Consider an arbitrary x.
>
> We first prove the $\Rightarrow$ implication.
>
> > 3. Assume $x \in A$.
> >
> > 4. Consider an arbitrary $y \in \mathrm{B}$.
> >
> > 5. $(x, y) \in A \times \mathrm{B}$ by defn $\times$
> >
> > 6. $(x, y) \in \mathrm{B} \times A$ by 1
> >
> > 7. $x \in \mathrm{B}$ by defn $\times$
>
> 8. $x \in A \Rightarrow x \in \mathrm{B}$ from 3–7 by $\Rightarrow$-introduction
>
> 9. The proof of the $\Leftarrow$ implication is symmetric

10. $\forall\, x . x \in A \Leftrightarrow x \in \mathrm{B}$ from 2–9 by $\forall$-introduction

Theorem $(A \times \mathrm{B} = \mathrm{B} \times A) \wedge A \neq \emptyset \wedge \mathrm{B} \neq \emptyset \Rightarrow A = \mathrm{B}$

Proof

0. Assume $A \neq \emptyset$ and $\mathrm{B} \neq \emptyset$

1. Assume $A \times \mathrm{B} = \mathrm{B} \times A$

We prove $A = \mathrm{B}$, i.e. $\forall\, x . x \in A \Leftrightarrow x \in \mathrm{B}$

> 2. Consider an arbitrary x.
>
> We first prove the $\Rightarrow$ implication.
>
> > 3. Assume $x \in A$.
> >
> > 4. Consider an arbitrary $y \in \mathrm{B}$ using 0
> >
> > 5. $(x, y) \in A \times \mathrm{B}$ by defn $\times$
> >
> > 6. $(x, y) \in \mathrm{B} \times A$ by 1
> >
> > 7. $x \in \mathrm{B}$ by defn $\times$
>
> 8. $x \in A \Rightarrow x \in \mathrm{B}$ from 3–7 by $\Rightarrow$-introduction
>
> 9. The proof of the $\Leftarrow$ implication is symmetric

10. $\forall\, x . x \in A \Leftrightarrow x \in \mathrm{B}$ from 2–9 by $\forall$-introduction □

Theorem $(A \times \mathrm{B} = \mathrm{B} \times A) \wedge A \neq \emptyset \wedge \mathrm{B} \neq \emptyset \Rightarrow A = \mathrm{B}$

or equivalently

Theorem $(A \times \mathrm{B} = \mathrm{B} \times A) \Rightarrow A = \mathrm{B} \vee A = \emptyset \vee \mathrm{B} = \emptyset$

using $((P \wedge R) \Rightarrow Q)$ iff $(P \Rightarrow Q \vee \neg R)$ and De Morgan

Aside

Let $A \overset{\text{def}}{=} \{n \mid n = n + 1\}$

Is $\forall\ x \in A.x = 7$ true?

Or $\forall\ x \in A.x = x + 1$? Or $\forall\ x \in A.1 = 2$?

Is $\exists\ x \in A.1 + 1 = 2$ true?

Relations, Graphs, and Orders

Using Products: Relations

Say a (binary) *relation* R between two sets A and B is a subset of all the (a, b) pairs (where $a \in A$ and $b \in \mathrm{B}$)

$R \subseteq A \times \mathrm{B}$ (or, or course, $R \in \mathcal{P}(A \times \mathrm{B})$)

Extremes: $\varnothing$ and $A \times \mathrm{B}$ are both relations between A and B

$1_A \stackrel{\text{def}}{=} \{(a, a) \mid a \in A\}$ is the *identity relation* on A

$\varnothing \subseteq 1_A \subseteq A \times A$

Sometimes write infix: $a \; R \; b \stackrel{\text{def}}{=} (a, b) \in R$

Relational Composition

Given $R \subseteq A \times \mathrm{B}$ and $S \subseteq \mathrm{B} \times \mathrm{C}$, their *relational composition* is

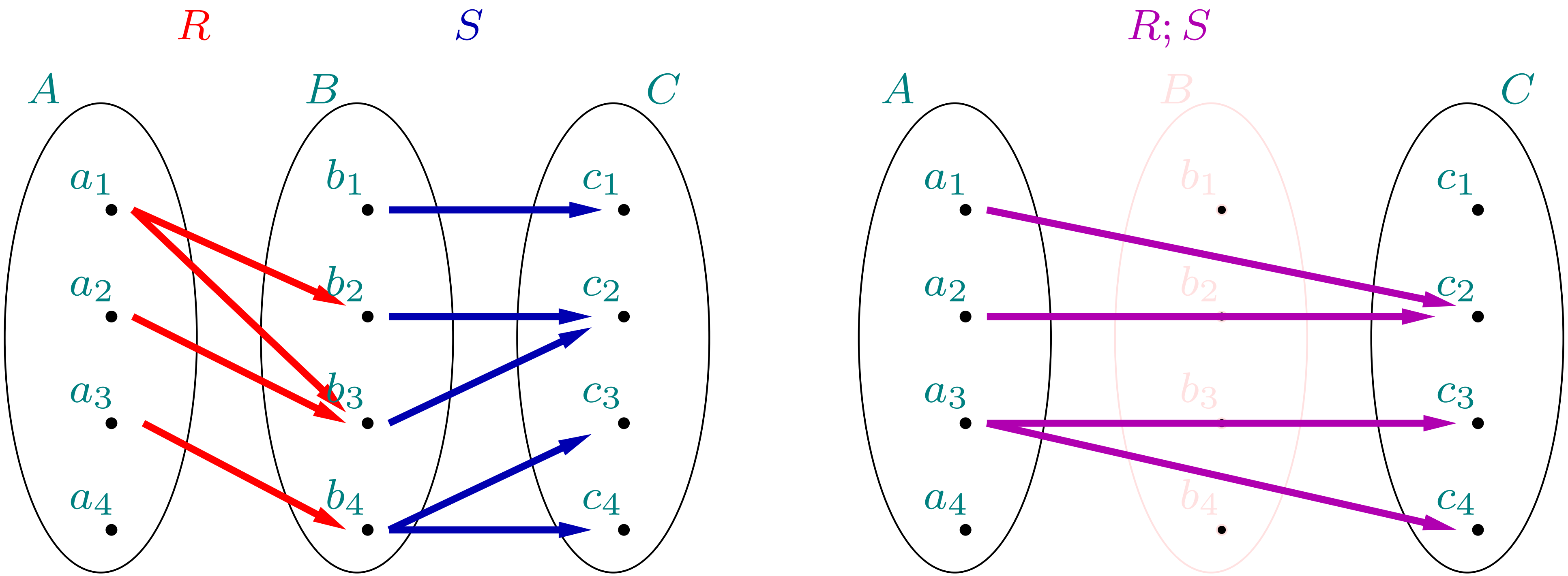
$$R;S \overset{\mathrm{def}}{=} \{(a,c) \mid \exists\, b.(a,b) \in R \wedge (b,c) \in S\}$$

$$R;S \subseteq A \times \mathrm{C}$$

Sometimes write that the other way round: $S \circ R \overset{\mathrm{def}}{=} R;S$

(to match function composition)

Relational Composition



$A \overset{\text{def}}{=} \{a_1, a_2, a_3, a_4\}$ B $\overset{\text{def}}{=} \{b_1, b_2, b_3, b_4\}$ C $\overset{\text{def}}{=} \{c_1, c_2, c_3, c_4\}$

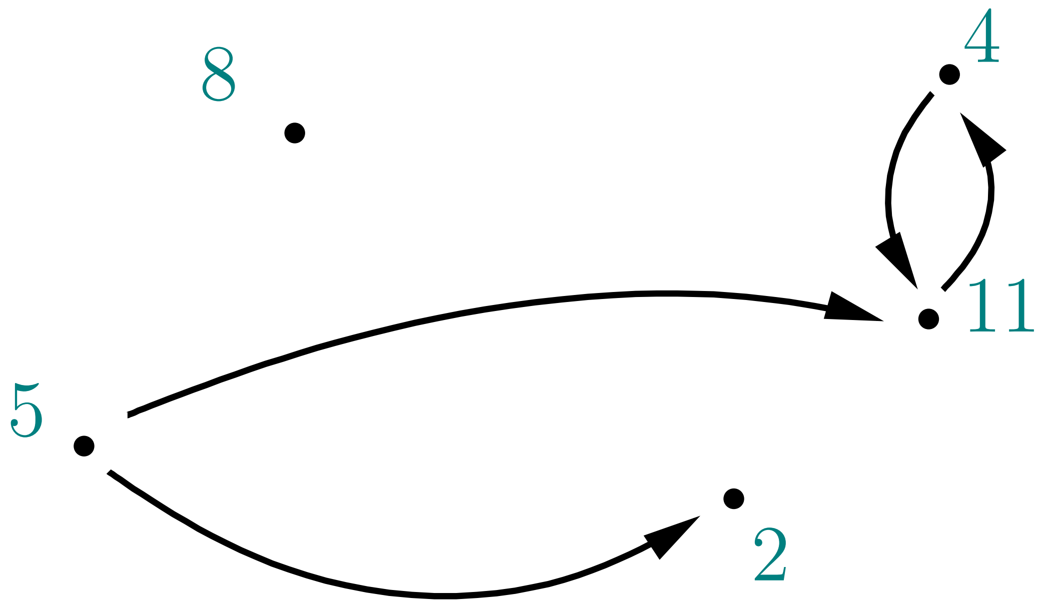
$R \overset{\text{def}}{=} \{(a_1, b_2), (a_1, b_3), (a_2, b_3), (a_3, b_4)\}$

$S \overset{\text{def}}{=} \{(b_1, c_1), (b_2, c_2), (b_3, c_2), (b_4, c_3), (b_4, c_4)\}$

$R; S = \{(a_1, c_2), (a_2, c_2), (a_3, c_3), (a_3, c_4)\}$

Relations as Directed Graphs

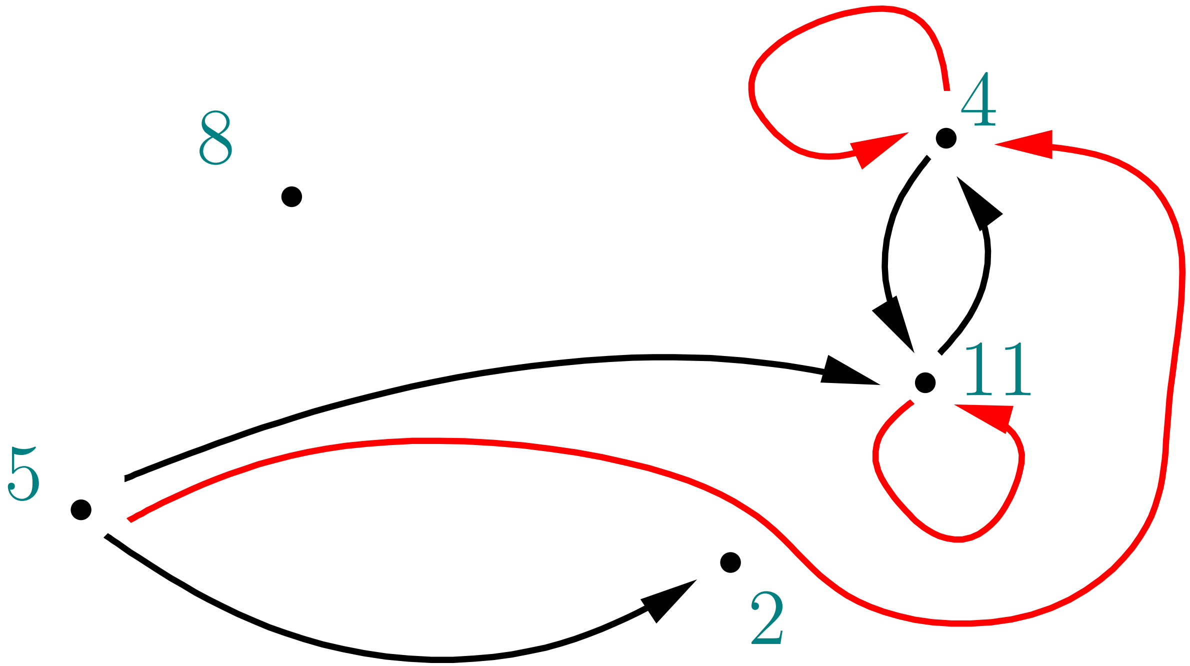
Relations from a set to itself



$$G \subseteq \mathbb{N} \times \mathbb{N}$$

$$G = \{(5,2),(5,11),(4,11),(11,4)\}$$

Transitivity



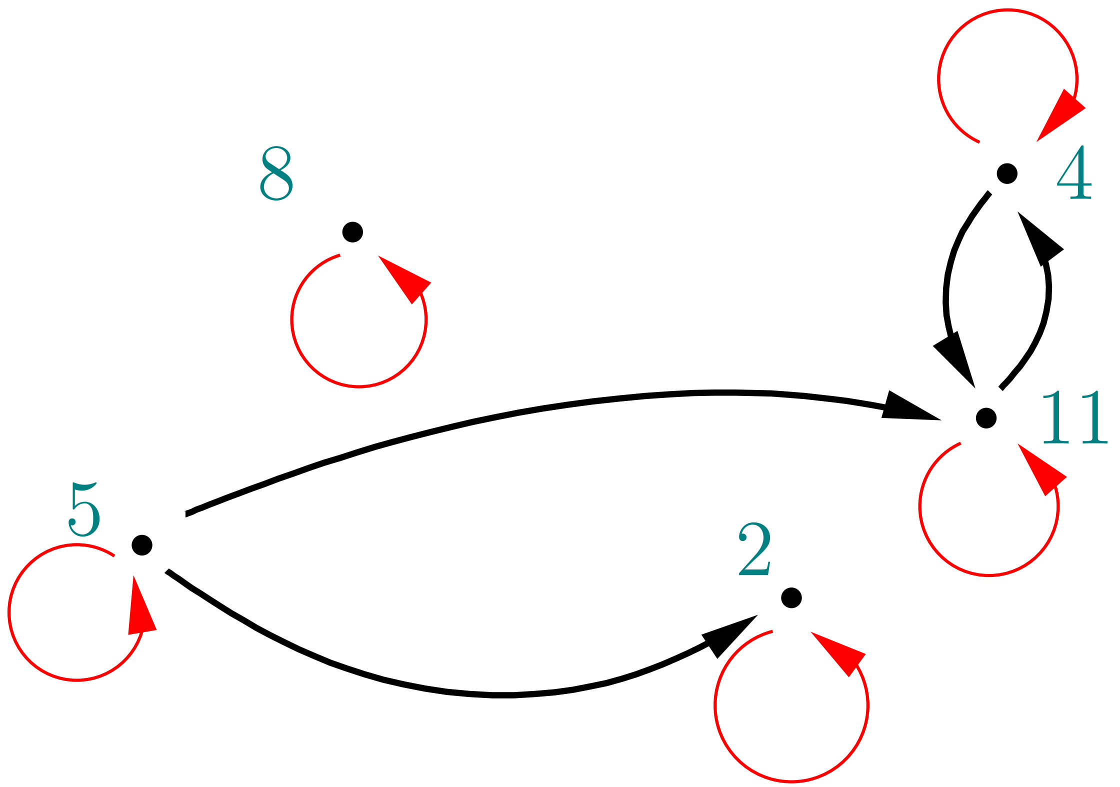
$R \subseteq A \times A$

$R^{+} \overset{\text{def}}{=} R \cup (R;R) \cup (R;R;R) \cup \ldots$

$G^{+} = \{(5,2),(5,11),(4,11),(11,4)\} \cup \{(5,4),(11,11),(4,4)\}$

R *is transitive* if $R = R^{+}$

Reflexivity

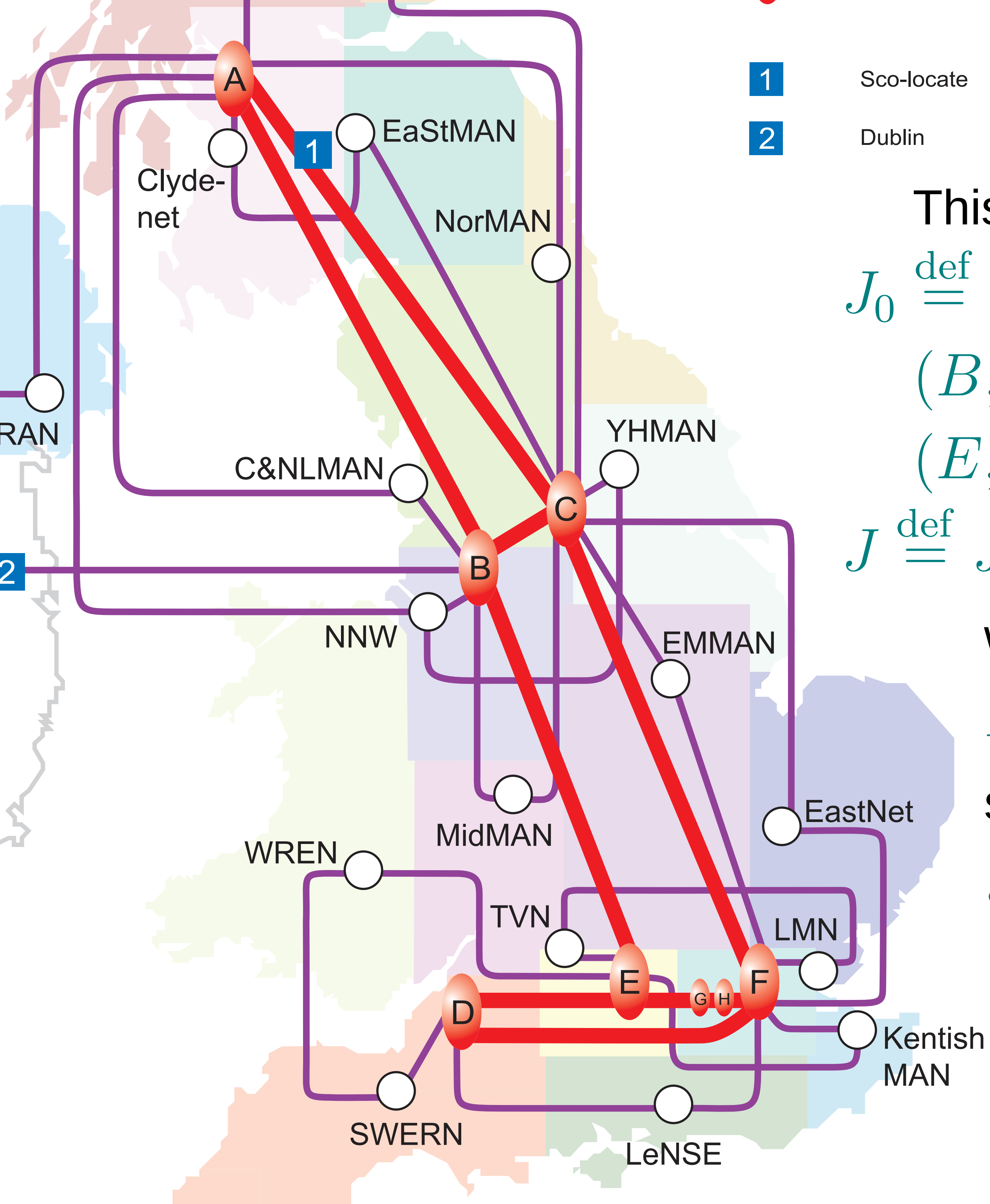


$A \overset{\text{def}}{=} \{2, 4, 5, 8, 11\}$

$G \cup 1_A =$

$\{(5,2), (5,11), (4,11), (11,4), (2,2), (4,4), (5,5), (8,8), (11,11)\}$

$R \subseteq A \times A$ *is reflexive* (over A) if $\forall\, a \in A.(a,a) \in R$



This is an *undirected* graph

$$J_0 \stackrel{\text{def}}{=} \{(A,B),(A,C),(B,C),(B,E),(C,F),(E,D),(D,F),(E,G),(G,H),(H,F)\}$$

$$J \stackrel{\text{def}}{=} J_0 \cup J_0^{-1}$$

where the inverse of R is

$$R^{-1} \stackrel{\text{def}}{=} \{(y,x) | (x,y) \in R\}$$

so J is *symmetric*, i.e.

$$J = J^{-1}$$

Directed Acyclic Graphs (DAGs)

$R \subseteq A \times A$ represents a *directed acyclic graph* if its transitive closure R^+ is *acyclic*, i.e.

$$\neg\exists\ a \in A.(a, a) \in R^+$$

Equivalence Relations

$R \subseteq A \times A$ is an *equivalence relation* (over A) if:

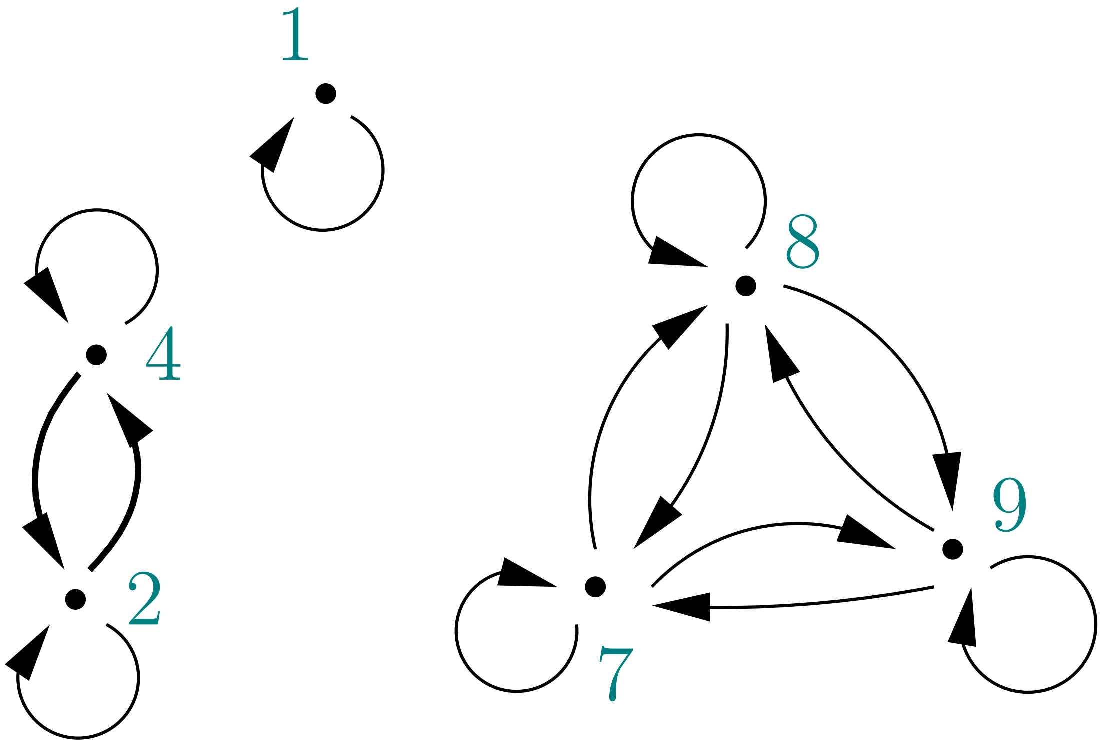
- R is reflexive, i.e. $\forall\ a \in A.(a, a) \in R$
- R is transitive,

  i.e. $\forall\ a_1, a_2, a_3 \in A.((a_1, a_2) \in R \wedge (a_2, a_3) \in R) \Rightarrow (a_1, a_3) \in R$
- R is symmetric, i.e. $\forall\ a_1, a_2 \in A.(a_1, a_2) \in R \Rightarrow (a_2, a_1) \in R$

e.g.$\{(m, n) \mid m \text{ mod } 3 = n \text{ mod } 3\}$ (over $\mathbb{N}$)

The *equivalence class* of $a \in A$ is all the things related to it, i.e. $\{a' \mid (a, a') \in R\}$

Equivalence Relations



An equivalence relation over $\{1, 2, 4, 7, 8, 9\}$

$\{(1,1),(2,2),(4,4),(2,4),(4,2),(7,7),(8,8),(9,9),(7,8),(8,7),(8,9),(9,8),(9,7),(7,9)\}$

with three equivalence classes: $\{1\}$, $\{2, 4\}$, and $\{7, 8, 9\}$

Pre-Orders

Reflexive transitive relations are known as *pre-orders* .

Suppose $\leq \subseteq A \times A$ is a pre-order over A.

By the definition, $a \leq a$, and if $a_1 \leq a_2 \leq a_3$ then $a_1 \leq a_3$.

But we can have $a_1 \leq a_2 \leq a_1$ for $a_1 \neq a_2$.

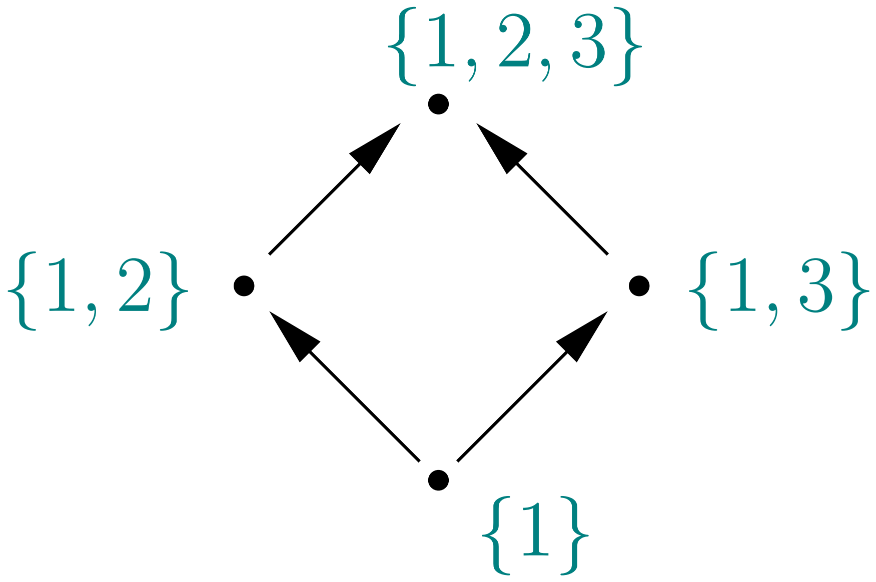
(Note that we drew pairs (a_1, a_2) as $a_1 \longrightarrow a_2$, but write $(a_1, a_2) \in \leq$ or $a_1 \leq a_2$)

Partial Orders

A *partial order* $\leq$ over A is a reflexive transitive relation (so a pre-order) that is also *antisymmetric*,

$$\forall\ a_1, a_2 \in A. (a_1 \leq a_2 \wedge a_2 \leq a_1) \Rightarrow (a_1 = a_2)$$

For example, here's part of the $\subseteq$ relation over sets:



(when we draw a partial order, we usually omit the refl and tran edges — these are *Hasse diagrams*)

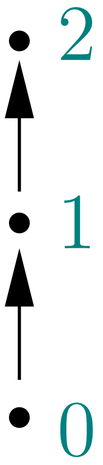
Total Orders

A *total order* (or *linear order*) $\leq$ over A is a reflexive, transitive, antisymmetric relation (so a partial order) that is also *total*,

$$\forall\ a_1, a_2 \in A.(a_1 \leq a_2 \vee a_2 \leq a_1)$$

(in fact the reflexivity condition is redundant)

For example, here's a Hasse diagram of part of the usual $\leq$ relation over $\mathbb{N}$:



Special Relations — Summary

A relation $R \subseteq A \times A$ is a directed graph. Properties:

- transitive $\forall\ a_1, a_2, a_3 \in A.(a_1\ R\ a_2 \wedge a_2\ R\ a_3) \Rightarrow a_1\ R\ a_3$
- reflexive $\forall\ a \in A.(a\ R\ a)$
- symmetric $\forall\ a_1, a_2 \in A.(a_1\ R\ a_2 \Rightarrow a_2\ R\ a_1)$
- acyclic $\forall\ a \in A.\neg(a\ R^+ a)$
- antisymmetric $\forall\ a_1, a_2 \in A.(a_1\ R\ a_2 \wedge a_2\ R\ a_1) \Rightarrow a_1 = a_2$
- total $\forall\ a_1, a_2 \in A.(a_1\ R\ a_2 \vee a_2\ R\ a_1)$

Combinations of properties: R is a ...

- directed acyclic graph if the transitive closure is acyclic
- undirected graph if symmetric
- equivalence relation if reflexive, transitive, and symmetric
- pre-order if reflexive and transitive,
- partial order if reflexive, transitive, and antisymmetric
- total order if reflexive, transitive, antisymmetric, and total

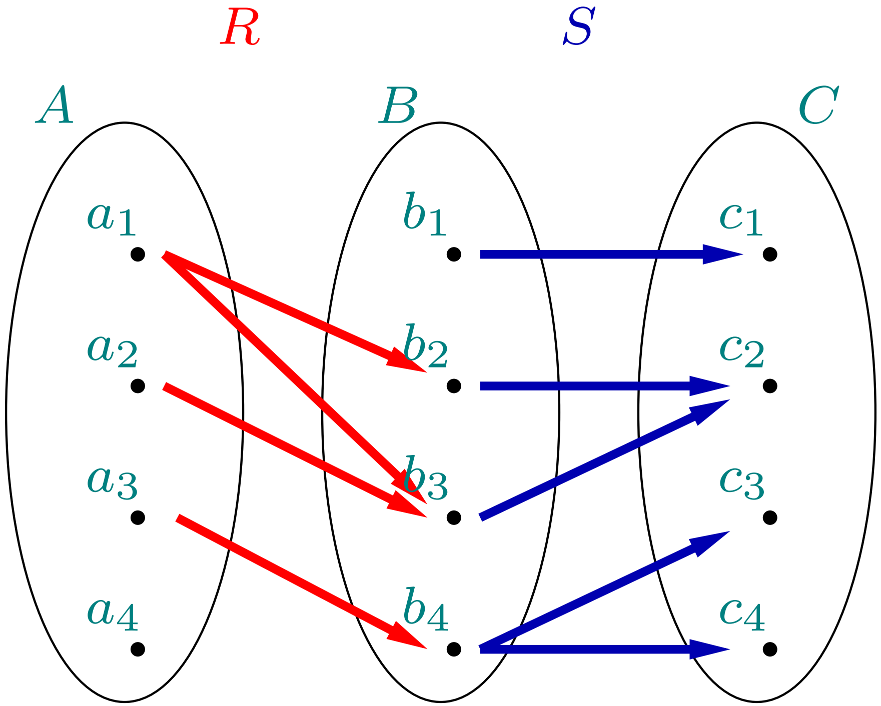
Functions

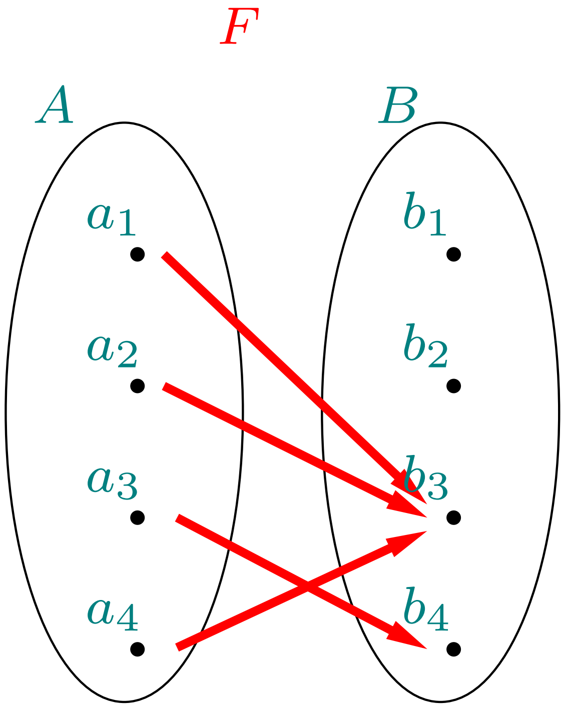
A *function* from A to B is just a relation which identifies exactly one element of B for each element of A.

$R \subseteq A \times \mathrm{B}$ is *functional* iff

$\forall a \in A. \exists b \in \mathrm{B}. (a, b) \in R$ and

$\forall a \in A. \forall b, b' \in \mathrm{B}. ((a, b) \in R \wedge (a, b') \in R) \Rightarrow b = b'$





Application — Relaxed Memory: One Intel/AMD Example

Initial shared memory values: $x = 0 \qquad y = 0$

Per-processor registers: $r_A \qquad r_B$

Processor A	Processor B
`store` $x := 1$	`store` $y := 1$
`load` $r_A := y$	`load` $r_B := x$

Processor A	Processor B
MOV [x]←$1	MOV [y]←$1
MOV EAX←[y]	MOV EBX←[x]

Final register values: $r_A =?$ $\qquad r_B =?$

Application — Relaxed Memory: One Intel/AMD Example

Initial shared memory values: $x = 0 \qquad y = 0$

Per-processor registers: $r_A \qquad r_B$

Processor A	Processor B
$\texttt{store}\ x := 1$	$\texttt{store}\ y := 1$
$\texttt{load}\ r_A := y$	$\texttt{load}\ r_B := x$

Processor A	Processor B
MOV [x]←$1	MOV [y]←$1
MOV EAX←[y]	MOV EBX←[x]

Final register values: $r_A =?$ $\qquad r_B =?$

Each processor can do its own store action before the store of the other processor.

Makes it hard to understand what your programs are doing!

Already a real problem for OS, compiler, and library authors.

Application — Relaxed Memory: part of the formalisation

$$
\begin{array}{l}
\mathrm{preserved\_program\_order}\ \ \mathrm{E} = \\
\quad \{(e_1, e_2) \mid (e_1, e_2) \in (\mathrm{po\_strict}\ \ \mathrm{E}) \wedge \\
\qquad ((\exists p\ r.(\mathrm{loc}\ \ e_1 = \mathrm{loc}\ \ e_2) \wedge \\
\qquad\qquad\qquad (\mathrm{loc}\ \ e_1 = \mathsf{Some}\ (\mathsf{Location\_reg}\ p\ \ r))) \vee \\
\qquad (\mathrm{mem\_load}\ \ e_1 \wedge \mathrm{mem\_load}\ \ e_2) \vee \\
\qquad (\mathrm{mem\_store}\ \ e_1 \wedge \mathrm{mem\_store}\ \ e_2) \vee \\
\qquad (\mathrm{mem\_load}\ \ e_1 \wedge \mathrm{mem\_store}\ \ e_2) \vee \\
\qquad (\mathrm{mem\_store}\ \ e_1 \wedge \mathrm{mem\_load}\ \ e_2 \wedge (\mathrm{loc}\ \ e_1 = \mathrm{loc}\ \ e_2)) \vee \\
\qquad ((\mathrm{mem\_load}\ \ e_1 \vee \mathrm{mem\_store}\ \ e_1) \wedge \mathrm{locked}\ \ \mathrm{E}\ \ e_2) \vee \\
\qquad (\mathrm{locked}\ \ \mathrm{E}\ \ e_1 \wedge (\mathrm{mem\_load}\ \ e_2 \vee \mathrm{mem\_store}\ \ e_2)))\}
\end{array}
$$

Induction

Example

Theorem $\sum_{i=1}^{n} i = n * (n+1)/2$

Proof By induction on n.

Base case (0): $\sum_{i=1}^{0} i = 0 = 0 * 1/2$

Inductive case ($n+1$): Assume $\sum_{i=1}^{n} i = n * (n+1)/2$ as the inductive hypothesis, then we have to prove $\sum_{i=1}^{n+1} i = (n+1) * ((n+1)+1)/2$.

But $\sum_{i=1}^{n+1} i = \sum_{i=1}^{n} i + (n+1) = n * (n+1)/2 + (n+1) = (n+1) * (n+1+1)/2$ □

What's really going on?

Using a fact about $\mathbb{N}$, the *induction principle*

$$(P(0) \wedge (\forall\ n.P(n) \Rightarrow P(n+1))) \Rightarrow \forall\ n.P(n)$$

(really a schema — that's true for any predicate P)

We think of an induction hypothesis, here taking

$$P(n) \stackrel{\text{def}}{=} \sum_{i=1}^{n} i = n * (n+1)/2$$

and instantiate the schema with it:

$$\begin{array}{ll}
(& (\sum_{i=1}^{0} i = 0 * (0+1)/2) \wedge \\
 & (\forall\ n.(\sum_{i=1}^{n} i = n * (n+1)/2) \\
 & \qquad \Rightarrow \\
 & \qquad (\sum_{i=1}^{n+1} i = (n+1) * ((n+1)+1)/2))) \\
\Rightarrow & \\
\forall\ n. \sum_{i=1}^{n} i = n * (n+1)/2 &
\end{array}$$

$$
\begin{array}{l}
(\quad (\sum_{i=1}^{0} i = 0 * (0+1)/2) \wedge \\
\quad\ (\forall\ n. (\sum_{i=1}^{n} i = n * (n+1)/2) \\
\qquad\qquad \Rightarrow \\
\qquad\qquad (\sum_{i=1}^{n+1} i = (n+1) * ((n+1)+1)/2))) \\
\Rightarrow \\
\forall\ n. \sum_{i=1}^{n} i = n * (n+1)/2
\end{array}
$$

Then we prove the antecedents of the top-level implication (with our normal proof techniques), and use modus ponens to conclude the consequent.

Induction on lists

An ML function to append two lists:

```
fun app ([], ys)     = ys
  | app (x::xs, ys) = x :: app(xs,ys)
```

This is *terminating* and *pure* (no mutable state, no IO, no exceptions). So we can regard it as a mathematical function app.

It operates on *lists*. Suppose they are lists of elements of a set A.

Is app associative?

Induction on lists

Theorem

$$\forall\ xs, ys, zs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$$

Proof We use the *induction schema for lists*

$$(P([]) \wedge (\forall\ xs.P(xs) \Rightarrow \forall\ x.P(x :: xs))) \Rightarrow \forall\ xs.P(xs)$$

with the induction hypothesis

$$P(xs) \overset{\text{def}}{=} \ \forall\ ys, zs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$$

Base case: we have to prove $P([])$,

i.e. $\forall\ ys, zs.\mathsf{app}(\mathsf{app}([], ys), zs) = \mathsf{app}([], \mathsf{app}(ys, zs))$

a. Consider arbitrary ys and zs.

b. $\mathsf{app}(\mathsf{app}([], ys), zs) = \mathsf{app}(ys, zs)$ by the first clause of the defn of app

c. ... $= \mathsf{app}([], \mathsf{app}(ys, zs))$ by the first clause of the defn of app (backwards)

Inductive step: we have to prove $(\forall\ xs.P(xs) \Rightarrow \forall\ x.P(x :: xs)))$

1. Consider an arbitrary xs.

2. Assume $P(xs)$

3. $\forall\ ys, zs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$ from 2, unfolding defn of P

4. Consider an arbitrary x

(now we have to prove $P(x :: xs)$, i.e.

$\forall\ ys, zs.\mathsf{app}(\mathsf{app}(x :: xs, ys), zs) = \mathsf{app}(x :: xs, \mathsf{app}(ys, zs))$)

5. Consider arbitrary ys and zs

6. $\mathsf{app}(\mathsf{app}(x :: xs, ys), zs) = \mathsf{app}(x :: \mathsf{app}(xs, ys), zs)$ by the second clause of app

7. ... $= x :: \mathsf{app}(\mathsf{app}(xs, ys), zs)$ by the second clause of app

8. ... $= x :: \mathsf{app}(xs, \mathsf{app}(ys, zs))$ instantiating 3 with $ys = ys$,$zs = zs$ under $x :: \ldots$

9. ... $= \mathsf{app}(x :: xs, \mathsf{app}(ys, zs))$ by the second clause of app (backwards)

10. $P(x :: xs)$ from 5–9, by $\forall$-introduction and folding the defn of P

11. $\forall\ x.P(x :: xs)$ from 4–10 by $\forall$-introduction

12. $P(xs) \Rightarrow \forall\ x.P(x :: xs)$ from 2–11 by $\Rightarrow$-introduction

13. $\forall\ xs.P(xs) \Rightarrow \forall\ x.P(x :: xs)$ from 1–12 by $\forall$-introduction

Now from the induction scheme, (c), and (13), we have $\forall xs.P(xs)$, which (unfolding the defn of P) is exactly the theorem statement.

Simpler proof structure: first rearrange the quantifiers

$$\forall\ xs, ys, zs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$$

iff

$$\forall\ ys, zs.\forall\ xs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$$

Then consider arbitrary ys and zs, and inside that do induction on lists, with induction hypothesis

$$P(xs) \overset{\mathrm{def}}{=} \mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$$

(instead of $P(xs) \overset{\mathrm{def}}{=} \forall\ ys, zs.\mathsf{app}(\mathsf{app}(xs, ys), zs) = \mathsf{app}(xs, \mathsf{app}(ys, zs))$)

OK, as we don’t need to instantiate P at different ys and zs

Generalizing an Induction Hypothesis

ML functions for the length of a list:

```
fun nlength []        = 0
  | nlength (x::xs) = 1 + nlength xs
fun addlen (k,[])      = k
  | addlen (k,x::xs) = addlen(k+1,xs)
```

(compiler optimization?) Both are terminating and pure.

Theorem ? $\mathtt{addlen}(0, \ell) = \mathtt{nlength}(\ell)$

Induction on ℓ — but which induction hypothesis?

$P''(\ell) \stackrel{\text{def}}{=} \mathtt{addlen}(0, \ell) = \mathtt{nlength}(\ell)$ too weak

$P'(\ell) \stackrel{\text{def}}{=} \mathtt{addlen}(k, \ell) = k + \mathtt{nlength}(\ell)$ too rigid: need to vary k

$P(\ell) \stackrel{\text{def}}{=} \forall\, k.\mathtt{addlen}(k, \ell) = k + \mathtt{nlength}(\ell)$ just right

Base case: we need to show $P([\,])$, i.e. $\forall\ k.\mathtt{addlen}(k,[\,]) = k + \mathtt{nlength}([\,])$

1. Consider an arbitrary k.

2. $\mathtt{addlen}(k,[\,]) = k = k + 0 = k + \mathtt{nlength}(0)$ by the defn $\mathtt{addlen}$ and $\mathtt{nlength}$

Inductive step: we need to show $(\forall\ \ell.P(\ell) \Rightarrow \forall\ x.P(x :: \ell)))$

3. Consider an arbitrary ℓ

4. Assume the induction hypothesis $P(\ell)$, i.e. $\forall\ k.\mathtt{addlen}(k,\ell) = k + \mathtt{nlength}(\ell)$

5. Consider an arbitrary x

   (now we have to show $P(x :: \ell)$, i.e. $\forall\ k.\mathtt{addlen}(k, x :: \ell) = k + \mathtt{nlength}(x :: \ell)$)

6. Consider an arbitrary k

7. $\mathtt{addlen}(k, x :: \ell) = \mathtt{addlen}(k+1, \ell)$ by defn $\mathtt{addlen}$

8. ... $= (k+1) + \mathtt{nlength}(\ell)$ by 4, instantiating k with $k+1$

9. ... $= k + \mathtt{nlength}(x :: \ell)$ by defn $\mathtt{nlength}$

11. $\forall\ k.\mathtt{addlen}(k, x :: \ell) = k + \mathtt{nlength}(x :: \ell)$ from 6–9 by $\forall$-introduction

12. $P(x :: \ell)$ from 11 by folding defn P

13. $\forall\ x.P(x :: \ell)$ from 5–12 by $\forall$-introduction

14. $P(\ell) \Rightarrow \forall\ x.P(x :: \ell)$ from 4–13 by $\Rightarrow$-introduction

15. $\forall\ \ell.P(\ell) \Rightarrow \forall\ x.P(x :: \ell)$ from 3–14 by $\forall$-introduction

The theorem follows by instantiating P with $k = 0$ □

...rewriting that semi-structured proof more idiomatically:

Theorem $\texttt{addlen}(0, \ell) = \texttt{nlength}(\ell)$

Proof Induction on ℓ, with I.H. $P(\ell) \overset{\text{def}}{=} \forall\, k.\texttt{addlen}(k, \ell) = k + \texttt{nlength}(\ell)$

in induction schema $(P([]) \wedge (\forall\, xs.P(xs) \Rightarrow \forall\, x.P(x :: xs))) \Rightarrow \forall\, xs.P(xs)$

Base case: we need to show $P([])$

Consider an arbitrary k, then $\texttt{addlen}(k, []) = k = k + 0 = k + \texttt{nlength}(0)$ by defn $\texttt{addlen}$ and $\texttt{nlength}$

Inductive step: consider an arbitrary ℓ, assume $P(\ell)$, and consider an arbitrary x. We have to show $P(x :: \ell)$.

Consider an arbitrary k.

$\texttt{addlen}(k, x :: \ell) = \texttt{addlen}(k + 1, \ell)$ by defn $\texttt{addlen}$

... $= (k + 1) + \texttt{nlength}(\ell)$ by $P(\ell)$, instantiating k with $k + 1$

... $= k + \texttt{nlength}(x :: \ell)$ by defn $\texttt{nlength}$

Conclusion

We’ve introduced a good part of the language of discrete mathematics

(vocabulary, grammar, pragmatics...)

Fluency comes with use; you’ll see that this is a remarkably flexible tool for formulating and analysing computational problems.

The End