

PART 1: TYPE THEORY · CHAPTER 2

Hilbert Proofs

Alexander V. Gheorghiu

COMP0009 Logic · University College London

CHAPTER 2

Hilbert Proofs

Why should anyone believe the last line?

A checker with four rules

- ▶ Programs are terms built from K, S and application.
- ▶ A typing is a list of judgements $\Gamma \vdash M : A$, each justified by (K), (S), (var) or (app).
- ▶ The checker accepts a program exactly when such a list exists.

WHAT IT CANNOT DO

Type a definition with a **parameter**.

There is no rule for $f\ x = M$.

Before we can fix that, one question: what is a proof?

OPENING

Turing's clerk

In 1936, to say what *computable* means, Turing did not describe a machine. He described a **person**:

- ▶ a clerk with a pencil and squared paper;
- ▶ a finite alphabet of symbols;
- ▶ a few *states of mind*.

At each moment: look at one square, then write or rub out a symbol, move one square, change state.

That is all the clerk may do.

The machine that bears Turing's name is the clerk with the person taken out. Post, the same year, described a worker marking boxes and arrived at the same place.

OPENING

Now do the same for proving

Imagine a clerk with a *language of statements*: letters standing for claims whose insides we do not look at, and a way to build “if this then that”.

One act is permitted: write down a new line, if it has one of a few fixed shapes, or follows from lines already written by one fixed rule.

WHAT THE CLERK MAY NOT DO

No insight.

No appeal to what the letters mean.

No step that cannot be checked by looking at the shapes on the page.

OPENING

Turing's question, turned around

Turing asked

What can this clerk *compute*?

We ask

Why should anyone believe the **last line**?

We shall find that the clerk has been in your compiler all along.

SECTION 2.1

Formulas

First, the language.

2.1 · FORMULAS

Definition 2.1: formulas

Fix a stock of letters $p, q, r, \dots$. The formulas are defined inductively:

1. Every letter is a formula.
2. If A and B are formulas, so is $A \rightarrow B$, read “if A then B ” or “ A implies B ”.
3. Nothing else is a formula.

IN ONE LINE

$$A, B ::= p \mid A \rightarrow B$$

The arrow groups to the right: $A \rightarrow B \rightarrow C$ is $A \rightarrow (B \rightarrow C)$.

IN HASKELL

```
data Formula = Letter String | Imp
  Formula Formula
```

2.1 · FORMULAS

Reading formulas: where do the brackets go?

Put in the brackets the grouping convention leaves out.

AS WRITTEN

WITH THE BRACKETS PUT IN

$p \rightarrow q \rightarrow p$	$\implies$	$p \rightarrow (q \rightarrow p)$
$p \rightarrow q \rightarrow r \rightarrow s$	$\implies$	$p \rightarrow (q \rightarrow (r \rightarrow s))$
$(p \rightarrow q) \rightarrow p$	$\implies$	<i>already bracketed: a different formula from $p \rightarrow q \rightarrow p$</i>
$(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$	$\implies$	$(p \rightarrow (q \rightarrow r)) \rightarrow ((p \rightarrow q) \rightarrow (p \rightarrow r))$

The last one is about to become very important. Read it until it looks like a sentence.

2.1 · FORMULAS

Your turn: formula or not?

Which of these are formulas? For those that are, fully bracket them.

1. $p \rightarrow q \rightarrow q \rightarrow p$
2. $p \rightarrow \rightarrow q$
3. $(p \rightarrow q) \rightarrow (r)$
4. $p q \rightarrow r$

ONE MINUTE

Use Definition 2.1 and nothing else.

2.1 · FORMULAS

Your turn: the answer

ITEM	CANDIDATE	VERDICT
1.	$p \rightarrow q \rightarrow q \rightarrow p$	formula: $p \rightarrow (q \rightarrow (q \rightarrow p))$
2.	$p \rightarrow \rightarrow q$	not a formula: an arrow needs a formula on each side
3.	$(p \rightarrow q) \rightarrow (r)$	formula, with a redundant bracket: $(p \rightarrow q) \rightarrow r$
4.	$p q \rightarrow r$	not a formula: juxtaposition is not in the grammar

Item 4 is a perfectly good term of Chapter 1. Formulas and terms are different languages.

2.1 · FORMULAS

Spelled alike, not the same

DEFINITION 1.9: TYPES

$$A, B ::= p \mid A \rightarrow B$$

$A \rightarrow B$ is a **contract**: give me an A , I return a B .

DEFINITION 2.1: FORMULAS

$$A, B ::= p \mid A \rightarrow B$$

$A \rightarrow B$ is a **statement**: B follows from A .

Same grammar, same brackets, same strings. For now treat them as different things that happen to be spelled alike.

There is more to logic than $\rightarrow$: and, or, not arrive in Chapter 3, and for-all in Chapter 4. One connective is enough to see the shape of a proof.

SECTION 2.2

Axioms, rules, and proofs

Two shapes and one rule.

What may the clerk write?

A proof is a list of lines. A line may be written only if

- ▶ it has one of a few fixed shapes, or
- ▶ it follows from earlier lines by one fixed rule.

Which shapes? Two will do.

A PROMISE

They were chosen so that one particular theorem would hold: the one your compiler relies on.

Section 2.7. Until then, take the shapes as given.

The first shape, in words

$$A \rightarrow B \rightarrow A$$

A claim is implied by anything: if A holds, then A holds whatever else does.

Read it as $A \rightarrow (B \rightarrow A)$. Once you have A , an extra hypothesis B changes nothing.

The second shape, in words

$$(A \rightarrow B \rightarrow C) \rightarrow (A \rightarrow B) \rightarrow A \rightarrow C$$

Modus ponens carried out under a hypothesis.

If, given A , you have $B \rightarrow C$, and, given A , you have B , then, given A , you have C .

The rule: modus ponens

From an implication and its antecedent, write the consequent.

$$\frac{A \rightarrow B \quad A}{B}$$

AN INSTANCE

If the ground is wet whenever it rains, and it rains, then the ground is wet.

Lines $A \rightarrow B$ and A on the page; write B .

Definition 2.2: axiom schemas and modus ponens

The axioms are all instances of two schemas, where A , B , C stand for arbitrary formulas:

$$(I) \quad A \rightarrow B \rightarrow A$$

$$(II) \quad (A \rightarrow B \rightarrow C) \rightarrow (A \rightarrow B) \rightarrow A \rightarrow C$$

The one rule of inference is modus ponens: from $A \rightarrow B$ and A , infer B .

Each schema stands for infinitely many axioms, all of one shape. Systems of this kind carry Hilbert's name; this design is Frege's.

Instances of (I)

An instance: substitute a formula for each of A , B and read off the result.

SUBSTITUTION

INSTANCE OF (I)

 $A := p, B := q$ $p \rightarrow q \rightarrow p$ $A := r \rightarrow s, B := p$ $(r \rightarrow s) \rightarrow p \rightarrow (r \rightarrow s)$ $A := p, B := p$ $p \rightarrow p \rightarrow p$ $A := p \rightarrow q, B := r$ $(p \rightarrow q) \rightarrow r \rightarrow (p \rightarrow q)$

Same formula for A at both ends. B is free.

Instances of (II)

Substitute for A, B, C in the schema $(A \rightarrow B \rightarrow C) \rightarrow (A \rightarrow B) \rightarrow A \rightarrow C$.

SUBSTITUTION

$A := p, B := q, C := r$

$A := p, B := p \rightarrow p, C := p$

$A := q, B := p, C := q$

INSTANCE OF (II)

$(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$

$(p \rightarrow (p \rightarrow p) \rightarrow p) \rightarrow (p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$

$(q \rightarrow p \rightarrow q) \rightarrow (q \rightarrow p) \rightarrow q \rightarrow q$

The second line is the one we will need in a few minutes. Check it against the schema.

Your turn: is it an axiom?

Is each an instance of (I), of (II), or neither?

1. $p \rightarrow q \rightarrow q$
2. $q \rightarrow (p \rightarrow p) \rightarrow q$
3. $(p \rightarrow q) \rightarrow (q \rightarrow r) \rightarrow p \rightarrow r$
4. $(p \rightarrow p \rightarrow p) \rightarrow (p \rightarrow p) \rightarrow p \rightarrow p$

ONE MINUTE

Find the substitution, or say why none exists.

Your turn: the answer

ITEM FORMULA

VERDICT

1. $p \rightarrow q \rightarrow q$

neither: (I) would need q at the end to match p at the start

2. $q \rightarrow (p \rightarrow p) \rightarrow q$

(I) with $A := q$, $B := p \rightarrow p$

3. $(p \rightarrow q) \rightarrow (q \rightarrow r) \rightarrow p \rightarrow r$

neither: it is the *conclusion* of hypothetical syllogism, not an instance of either schema

4. $(p \rightarrow p \rightarrow p) \rightarrow (p \rightarrow p) \rightarrow p \rightarrow p$

(II) with $A := p$, $B := p$, $C := p$

Checking a shape is a matter of comparing the pieces. No understanding required.

Checking is pattern matching

AN AXIOM

Does the line match (I) or (II)? A simple parser decides, symbol by symbol.

MODUS PONENS

Find the two cited lines. Confirm that one is exactly the other with an arrow and the new line attached.

Pattern matching. No understanding required.

Definition 2.3: proof from hypotheses; theorem

Let Γ be a set of formulas, the hypotheses. A proof of A from Γ is a finite sequence of formulas $A_1, A_2, \dots, A_n$ whose last line A_n is A , and in which each line A_k is justified in one of three ways:

1. (*axiom*) A_k is an instance of (I) or (II);
2. (*hypothesis*) A_k is a member of Γ ;
3. (*modus ponens*) there are earlier lines A_i, A_j with $i, j < k$ such that A_i is the formula $A_j \rightarrow A_k$.

NOTATION

If such a proof exists, write $\Gamma \vdash A$: “ A is provable from Γ ”.

When Γ is empty, write $\vdash A$ and call A a **theorem**.

No other kind of line is allowed.

Definition 2.3, in words

- ▶ Go down the list. Every line must be an axiom, a hypothesis, or the B of a modus ponens.
- ▶ The two premises of that modus ponens, $A \rightarrow B$ and A , must both appear **higher up**. That is the force of $i, j < k$.
- ▶ So checking is a single pass down the page.
- ▶ On the right of each line, write its justification. “MP i, j ” means: line i is the implication, line j is its antecedent.

The implication first, as the function came first in “(app) i, j ”.

Worked: a proof from hypotheses

Show $\{p \rightarrow q, p\} \vdash q$.

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

1.	$p \rightarrow q$	hypothesis
----	-------------------	------------

2.	p	hypothesis
----	-----	------------

3.	q	MP 1, 2
----	-----	---------

► Line 3 comes from MP: is line 1 the formula *line 2* $\rightarrow$ *line 3*? Yes: $p \rightarrow q$.

► Both cited lines are higher up.

Worked: two modus ponens in a row

Show $\{p \rightarrow q, q \rightarrow r, p\} \vdash r$.

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q$	hypothesis
2.	p	hypothesis
3.	q	MP 1, 2
4.	$q \rightarrow r$	hypothesis
5.	r	MP 4, 3

A proved line is used exactly like a hypothesis. Line 3 is cited by line 5.

The smallest theorem

With $\Gamma = \emptyset$ the hypothesis clause never applies. A proof is just axioms and modus ponens.

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

1.	$p \rightarrow q \rightarrow p$	(I) $[p, q]$
----	---------------------------------	--------------

One line. So $\vdash p \rightarrow q \rightarrow p$: a theorem, because the line is an axiom.

Every instance of a schema is a one-line theorem.

Hypotheses are what you have been told

A PROOF FROM Γ

Says what follows from what you have been told.

A THEOREM

The special case of nothing told at all.

A function does not know its inputs either: it is told what kind of thing each will be, and works for every input of that kind at once.

That was the context Γ of Definition 1.10. We write the same symbol $\vdash$ in both places, and that is not an accident.

Definition 1.11 beside Definition 2.3: the typing

DEFINITION 1.11: TYPING

A typing of $\Gamma \vdash M : A$ is a finite sequence of judgements, ending in that one, each line justified by:

1. (K) $\vdash K : B \rightarrow C \rightarrow B$;
2. (S) $\vdash S : (B \rightarrow C \rightarrow D) \rightarrow (B \rightarrow C) \rightarrow B \rightarrow D$;
3. (var) $x : B$ is in Γ , so $\Gamma \vdash x : B$;
4. (app) earlier lines $M : B \rightarrow C$ and $N : B$ give $M N : C$.

FOUR CASES

Two constants with a ready-made type, a variable from the context, an application of two earlier lines.

Definition 1.11 beside Definition 2.3: the proof

DEFINITION 2.3: PROOF

A proof of A from Γ is a finite sequence of formulas, ending in A , each line justified by:

1. (I) an instance of $B \rightarrow C \rightarrow B$;
2. (II) an instance of $(B \rightarrow C \rightarrow D) \rightarrow (B \rightarrow C) \rightarrow B \rightarrow D$;
3. (hypothesis) a member of Γ ;
4. (MP) earlier lines $B \rightarrow C$ and B give C .

THREE CLAUSES, FOUR SHAPES

Definition 2.3 lists three clauses; its first covers two schemas. Counting (I) and (II) separately, the four match the typing's four, in the same order.

The two definitions, matched

A TYPING HAS

(K) $K : B \rightarrow C \rightarrow B$ (S) $S : (B \rightarrow C \rightarrow D) \rightarrow (B \rightarrow C) \rightarrow B \rightarrow D$ (var) $x : B$ from Γ (app) i, j

A PROOF HAS

(I) $B \rightarrow C \rightarrow B$ (II) $(B \rightarrow C \rightarrow D) \rightarrow (B \rightarrow C) \rightarrow B \rightarrow D$ hypothesis $B \in \Gamma$ MP i, j

(I) is the type the checker gives K. (II) is the type it gives S. Checking a typing and constructing a proof are one activity.

SECTION 2.3

Typings are proofs

Rub out the programs.

Take a typing from Chapter 1

Example 1.12: SKK at $A \rightarrow A$. Write P for $A \rightarrow A$.

LINE	JUDGEMENT	JUSTIFICATION
1.	$\vdash S : (A \rightarrow P \rightarrow A) \rightarrow (A \rightarrow P) \rightarrow P$	(S) $[A, P, A]$
2.	$\vdash K : A \rightarrow P \rightarrow A$	(K) $[A, P]$
3.	$\vdash SK : (A \rightarrow P) \rightarrow P$	(app) 1, 2
4.	$\vdash K : A \rightarrow P$	(K) $[A, A]$
5.	$\vdash SKK : A \rightarrow A$	(app) 3, 4

2.3 · TYPINGS ARE PROOFS

Now rub out the programs

Delete everything to the left of each colon. Rename A to p . Replace each justification by its counterpart.

LINE	FORMULA	JUSTIFICATION
1.	$(p \rightarrow P \rightarrow p) \rightarrow (p \rightarrow P) \rightarrow P$	(II) $[p, P, p]$
2.	$p \rightarrow P \rightarrow p$	(I) $[p, P]$
3.	$(p \rightarrow P) \rightarrow P$	MP 1, 2
4.	$p \rightarrow P$	(I) $[p, p]$
5.	$p \rightarrow p$	MP 3, 4

Line 1 is an instance of (II). Line 3 is “MP 1, 2”. What is left is a five-line proof of $p \rightarrow p$ from no hypotheses.

The dictionary

IN A TYPING

IN A PROOF

the constant K $\longleftrightarrow$

an instance of (I)

the constant S $\longleftrightarrow$

an instance of (II)

a variable in the context

 $\longleftrightarrow$

a hypothesis

application $M N$ $\longleftrightarrow$

modus ponens

Two constants are two axiom schemas with programs attached. Application is modus ponens with programs attached. A variable is a hypothesis with a name.

Theorem 2.4: typings are proofs

For a context $\Gamma = x_1 : A_1, \dots, x_n : A_n$ write $|\Gamma| = \{A_1, \dots, A_n\}$: its types, now read as formulas.

THEOREM 2.4 · TYPINGS ARE PROOFS

- (i) (*Erasure*) Let a typing end in $\Gamma \vdash M : A$. Replace each line $\Gamma \vdash N : B$ by the formula B , and each justification by its counterpart: (K) by (I), (S) by (II), (var) by *hypothesis*, “(app) i, j ” by “MP i, j ”. The result is a proof of A from $|\Gamma|$.
- (ii) (*Decoration*) Conversely, let $A_1, \dots, A_n$ be a proof of A , and let Γ assign a distinct variable to each hypothesis the proof uses. Decorate each line: an instance of (I) with K, of (II) with S, a hypothesis with its variable, a line “MP i, j ” with $M_i M_j$. The result is a typing, ending in $\Gamma \vdash M_n : A$.

Why it is true: erasure, case by case

Go down the typing one line at a time and check that each justification survives.

1. A line justified by (K) has type $B \rightarrow C \rightarrow B$ for some B, C . Erased, that is an instance of (I).
2. A line justified by (S) has type $(B \rightarrow C \rightarrow D) \rightarrow (B \rightarrow C) \rightarrow B \rightarrow D$. Erased, an instance of (II).
3. A line justified by (var) has a type listed in Γ . Erased, a member of $|\Gamma|$: a hypothesis.
4. A line “(app) i, j ” has type C where line i has type $B \rightarrow C$ and line j has type B . Erased, these are $B \rightarrow C$ and B , so “MP i, j ” is correct.

Each line of the result is justified, and the last is A. Decoration runs the same four cases in reverse.

Worked: decorating a proof

Take the proof of q from $\{p \rightarrow q, p\}$. Give the hypotheses names: $f : p \rightarrow q, x : p$.

PROOF

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q$	hypothesis
2.	p	hypothesis
3.	q	MP 1, 2

TYPING

LINE	JUDGEMENT	JUSTIFICATION
1.	$f : p \rightarrow q$	(var)
2.	$x : p$	(var)
3.	$f x : q$	(app) 1, 2

MP decorates to application: f , then x , gives $f x$. The proof was a program all along.

Types are formulas

Types are formulas, and a typing is a proof.

Not “resembles”: is. Line for line, justification for justification.

Running a program removes detours (1)

A K step rewrites Kxy to x . Look at the proof of p behind it, with $x : p$ and $y : q$.

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q \rightarrow p$	(I) $[p, q]$ this is K
2.	p	hypothesis this is x
3.	$q \rightarrow p$	MP 1, 2
4.	q	hypothesis this is y
5.	p	MP 3, 4 this is Kxy

*Line 2 already is p . The proof went out through q and came back. That is a **detour**.*

Running a program removes detours (2)

BEFORE: Kxy

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q \rightarrow p$	(I)
2.	p	hyp
3.	$q \rightarrow p$	MP 1, 2
4.	q	hyp
5.	p	MP 3, 4

AFTER THE K STEP: x

LINE	FORMULA	JUSTIFICATION
2.	p	hyp

The step deletes the detour and keeps the proof of p that was there all along.

An S step is a detour of the same kind through (II).

Subject reduction, with the programs rubbed out

THEOREM 1.16, AS PROGRAMS

If $M : A$ and $M \rightsquigarrow N$, then $N : A$.

AS PROOFS

A proof with a detour removed is still a proof of the same formula.

Running a program is removing detours from a proof.

Everything the checker does with its four cases is something a logician would recognise. The next sections write proofs, and every proof written there is also a program typed.

SECTION 2.4

A first proof

Five lines to say that p implies p .

The claim

$$\vdash p \rightarrow p$$

“ p implies p ” is a theorem: provable from no hypotheses.

WHY IT IS NOT TRIVIAL

$p \rightarrow p$ is neither (I) nor (II). It must come from modus ponens, from lines we have to find.

2.4 · A FIRST PROOF

The proof

The square brackets give the formulas substituted for A , B , C , in that order.

LINE	FORMULA	JUSTIFICATION
1.	$(p \rightarrow (p \rightarrow p) \rightarrow p) \rightarrow (p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$	(II) $[p, p \rightarrow p, p]$
2.	$p \rightarrow (p \rightarrow p) \rightarrow p$	(I) $[p, p \rightarrow p]$
3.	$(p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$	MP 1, 2
4.	$p \rightarrow (p \rightarrow p)$	(I) $[p, p]$
5.	$p \rightarrow p$	MP 3, 4

2.4 · A FIRST PROOF

Checking it, as the clerk would

LINE JUSTIFICATION CLAIMED

THE CLERK'S CHECK

- | | |
|---|---|
| 1. (II) with $A, B, C := p, p \rightarrow p, p$ | fill in the schema: it matches |
| 2. (I) with $A, B := p, p \rightarrow p$ | fill in the schema: it matches |
| 3. MP 1, 2 | line 1 is line 2 $\rightarrow$ line 3? yes |
| 4. (I) with $A, B := p, p$ | fill in the schema: it matches |
| 5. MP 3, 4 | line 3 is line 4 $\rightarrow$ line 5? yes |

*Lines 1, 2, 4 are matched against two shapes. Lines 3 and 5 against the two lines they cite.
Each is trivial to check.*

Finding line 1 (1): work backwards

- ▶ The goal $p \rightarrow p$ is not an axiom, so it comes by modus ponens: from some C and $C \rightarrow (p \rightarrow p)$.
- ▶ Could $C \rightarrow (p \rightarrow p)$ be an instance of (I)? Then C would be p , and we would need a proof of p on its own.
- ▶ There is none: p is a bare letter, and nothing in the system proves a bare letter. (Chapter 3 confirms.)

so

Try (II).

Finding line 1 (2): try (II)

- ▶ The last part of (II), $A \rightarrow C$, must be $p \rightarrow p$. That fixes $A = C = p$ and leaves B free.
- ▶ The two premises are then $p \rightarrow (B \rightarrow p)$ and $p \rightarrow B$.
- ▶ Choose $B := p \rightarrow p$. The first is an instance of (I) with $A, B := p, p \rightarrow p$. The second is an instance of (I) with p, p .

That is line 1. The search took a paragraph; the check takes none.

2.4 · A FIRST PROOF

Beside Example 1.12

TYPING OF SKK

LINE JUSTIFICATION

1. (S) $[A, A \rightarrow A, A]$
2. (K) $[A, A \rightarrow A]$
3. (app) 1, 2
4. (K) $[A, A]$
5. (app) 3, 4

PROOF OF $p \rightarrow p$

LINE JUSTIFICATION

1. (II) $[p, p \rightarrow p, p]$
2. (I) $[p, p \rightarrow p]$
3. MP 1, 2
4. (I) $[p, p]$
5. MP 3, 4

With p for A and the programs rubbed out, the two lists are the same list. Theorem 2.4, on a five-line example.

Your turn: spot the bad line

Someone claims $\{p \rightarrow q\} \vdash q$ and offers:

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q$	hypothesis
2.	p	hypothesis
3.	q	MP 1, 2

Check it against Definition 2.3.

ONE MINUTE

Which line, if any, breaks the rules?
Why does it matter?

Your turn: the answer

- ▶ Line 2 is justified as a hypothesis, but p is *not* a member of $\Gamma = \{p \rightarrow q\}$. The proof is invalid.
- ▶ The same list *is* a proof of q from $\{p \rightarrow q, p\}$: that was the worked example earlier.

A hypothesis is what you were told. You cannot tell yourself more.

In program terms: line 2 uses a variable that is not in the context. The checker rejects it for exactly the same reason.

SECTION 2.5

Structural facts

Scope, the way a programmer takes it for granted.

A short proof from a hypothesis

Show $\{p\} \vdash q \rightarrow p$. The hypothesis is p ; we must show $q \rightarrow p$ follows from it.

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

- | | | |
|----|---------------------------------|--------------|
| 1. | $p \rightarrow q \rightarrow p$ | (I) $[p, q]$ |
| 2. | p | hypothesis |
| 3. | $q \rightarrow p$ | MP 1, 2 |

Weakening: from p , conclude that p holds whatever q does.

Hypothetical syllogism

Chain two implications: from $p \rightarrow q$ and $q \rightarrow r$, conclude $p \rightarrow r$. Show $\{p \rightarrow q, q \rightarrow r\} \vdash p \rightarrow r$.

LINE	FORMULA	JUSTIFICATION
1.	$q \rightarrow r$	hypothesis
2.	$(q \rightarrow r) \rightarrow p \rightarrow (q \rightarrow r)$	(I) $[q \rightarrow r, p]$
3.	$p \rightarrow (q \rightarrow r)$	MP 2, 1
4.	$(p \rightarrow (q \rightarrow r)) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$	(II) $[p, q, r]$
5.	$(p \rightarrow q) \rightarrow p \rightarrow r$	MP 4, 3
6.	$p \rightarrow q$	hypothesis
7.	$p \rightarrow r$	MP 5, 6

Read the blocks, not the symbols

- ▶ **Lines 1–3.** Take the hypothesis $q \rightarrow r$ and weaken it, by (I), to $p \rightarrow (q \rightarrow r)$: it says less, promising $q \rightarrow r$ only once p is given.
- ▶ **Lines 4–5.** Put that through (II), which turns “given p , q implies r ” into “if p gives q , then p gives r ”.
- ▶ **Line 7.** Fire the result on the other hypothesis.

Every Hilbert proof has this character: a few ideas, spelled out at a length the checker can follow.

2.5 · STRUCTURAL FACTS

Example 2.5: the two erasures

Example 1.12 erased to the proof of $p \rightarrow p$ (and decorated back gives $S K K$): the identity function *is* the proof that p implies p . Exercise 1.3(a), in the context $f : q \rightarrow r$, $g : p \rightarrow q$, erases to the proof of hypothetical syllogism; here it is with its programs written back:

LINE	FORMULA	JUSTIFICATION	PROGRAM
1.	$q \rightarrow r$	hypothesis	f
2.	$(q \rightarrow r) \rightarrow p \rightarrow (q \rightarrow r)$	(I)	K
3.	$p \rightarrow (q \rightarrow r)$	MP 2, 1	$K f$
4.	$(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$	(II)	S
5.	$(p \rightarrow q) \rightarrow p \rightarrow r$	MP 4, 3	$S (K f)$
6.	$p \rightarrow q$	hypothesis	g
7.	$p \rightarrow r$	MP 5, 6	$S (K f) g$

Composition, applied to two functions, is the proof that chains two implications.

Proposition 2.6: structural facts

PROPOSITION 2.6 · STRUCTURAL FACTS

For all sets of formulas Γ, Δ and all formulas A, B :

1. (*reflexivity*) if $A \in \Gamma$ then $\Gamma \vdash A$;
2. (*monotonicity*) if $\Gamma \subseteq \Delta$ and $\Gamma \vdash A$ then $\Delta \vdash A$;
3. (*transitivity*) if $\Gamma \vdash A$ and $\Delta \cup \{A\} \vdash B$ then $\Gamma \cup \Delta \vdash B$;
4. (*finiteness*) if $\Gamma \vdash A$ then $\Gamma_0 \vdash A$ for some finite $\Gamma_0 \subseteq \Gamma$.

Why: reflexivity and monotonicity

- (1) The one-line sequence A is a proof of A from Γ : its only line is justified as a hypothesis, because $A \in \Gamma$.
- (2) A proof of A from Γ is already a proof of A from Δ : every line justified as a member of Γ is a member of Δ too, and the other two kinds of line do not mention Γ .

In neither case is anything said about what the formulas mean, only about lists.

Why: transitivity is splicing

- ▶ Write out the proof of A from Γ .
- ▶ After it, write the proof of B from $\Delta \cup \{A\}$, with each line justified as “hypothesis A ” deleted.
- ▶ Each modus ponens that cited a deleted line now cites the last line of the first proof, which is A .
- ▶ Every remaining hypothesis is in Γ or Δ : a proof of B from $\Gamma \cup \Delta$.

Only this item rewrites a proof. The others use the list we were given as it stands.

2.5 · STRUCTURAL FACTS

Transitivity at work (1): the two proofs

Take $\Gamma = \{p\}$ and $A = q \rightarrow p$, and $\Delta = \{q\}$ with $B = p$.

$\Gamma \vdash A: \{p\} \vdash q \rightarrow p$

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

- | | | |
|----|---------------------------------|---------|
| 1. | $p \rightarrow q \rightarrow p$ | (I) |
| 2. | p | hyp |
| 3. | $q \rightarrow p$ | MP 1, 2 |

$\Delta \cup \{A\} \vdash B: \{q, q \rightarrow p\} \vdash p$

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

- | | | |
|----|-------------------|--------------------|
| 1. | $q \rightarrow p$ | hyp (this is A) |
| 2. | q | hyp |
| 3. | p | MP 1, 2 |

Transitivity at work (2): spliced

Delete line 1 of the second proof (the hypothesis A). Its use is replaced by the last line of the first proof.

LINE	FORMULA	JUSTIFICATION
1.	$p \rightarrow q \rightarrow p$	(I) first proof
2.	p	hypothesis
3.	$q \rightarrow p$	MP 1, 2
4.	q	hypothesis second proof, with its line 1 gone
5.	p	MP 3, 4

A proof of p from $\{p, q\}$. A theorem once proved is a line in any later proof, just as a function once written may be called.

Why: finiteness

- ▶ A proof is a finite list, so only finitely many of its lines are justified as hypotheses.
- ▶ Let Γ_0 be the set of formulas on those lines: a finite subset of Γ .
- ▶ The very same list is a proof of A from Γ_0 .

Looks trivial. It will matter in Chapter 9.

In the language of programs

FACT ABOUT $\vdash$

WHAT A PROGRAMMER TAKES FOR GRANTED

(1) reflexivity

a variable in scope may be used

(2) monotonicity

extra names in scope do no harm

(3) transitivity

a helper function may be inlined at its call sites

(4) finiteness

a program is a finite text

Logic has to earn them. Item (3) is the one to remember: proofs compose.

What we cannot yet do

What the checker cannot do is type a definition with a parameter. What we cannot yet see is what that has to do with logic.

The next section shows what it costs a proof to do without such a rule. The one after shows that the rule was there all the time.

SECTION 2.6

The cost of an implication

What an implication costs.

2.6 · THE COST OF AN IMPLICATION

The goal: hypothetical syllogism as a theorem

Before, we proved it *from* two implications written down as hypotheses. Now we want it with no hypotheses at all:

$$(q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$$

“If q implies r , then, if p implies q , p implies r .”

THE CONSTRAINT

Schemas and modus ponens only.
There is no “assume”.

Abbreviations, so it fits on a slide

$$P := q \rightarrow r, \quad Q := p \rightarrow q \underset{r}{\rightarrow} r, \quad R := (p \rightarrow q) \rightarrow p \rightarrow$$

The goal is then $P \rightarrow R$.

P is the first hypothesis of the old proof. $Q \rightarrow R$ is the instance of (II) that was its line 4.

2.6 · THE COST OF AN IMPLICATION

The proof from the schemas alone

LINE FORMULA

JUSTIFICATION

- | | |
|---|----------------------------|
| 1. $(P \rightarrow Q \rightarrow R) \rightarrow (P \rightarrow Q) \rightarrow P \rightarrow R$ | (II) $[P, Q, R]$ |
| 2. $Q \rightarrow R$, i.e. $(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$ | (II) $[p, q, r]$ |
| 3. $(Q \rightarrow R) \rightarrow P \rightarrow (Q \rightarrow R)$ | (I) $[Q \rightarrow R, P]$ |
| 4. $P \rightarrow Q \rightarrow R$ | MP 3, 2 |
| 5. $(P \rightarrow Q) \rightarrow P \rightarrow R$ | MP 1, 4 |
| 6. $P \rightarrow Q$, i.e. $(q \rightarrow r) \rightarrow p \rightarrow (q \rightarrow r)$ | (I) $[q \rightarrow r, p]$ |
| 7. $P \rightarrow R$ | MP 5, 6 |

2.6 · THE COST OF AN IMPLICATION

Look at line 1

Seven lines again, and each checks in a moment. But with the abbreviations undone, line 1 is an instance of (II) with **twenty letters and nineteen arrows**:

$$\begin{aligned} & ((q \rightarrow r) \rightarrow (p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r) \\ & \rightarrow ((q \rightarrow r) \rightarrow (p \rightarrow q \rightarrow r)) \rightarrow (q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r \end{aligned}$$

Nobody finds that by thinking about what implication means.

2.6 · THE COST OF AN IMPLICATION

The closed proof contains the open one

FROM HYPOTHESES (SECTION 2.5)

 $q \rightarrow r$ and $p \rightarrow q$ written down as lines 1 and 6

line 4: the instance of (II)

line 5: modus ponens, in the open

AS A THEOREM (THIS SECTION)

line 6 is $q \rightarrow r$ weakened by (I), exactly as line 2 was there

line 2: the same instance of (II)

line 1: an instance of (II) whose only job is to carry out, under the hypothesis P , the modus ponens that line 5 performed in the open

The closed proof contains the open one, rebuilt one level up.

2.6 · THE COST OF AN IMPLICATION

That is the cost

To prove $A \rightarrow B$ outright you cannot assume A and derive B : “assume” is not one of the three kinds of line.

You must build the proof of B with A threaded through every step, by instances of (II) that grow with the formulas they carry.

FOR SCALE

One arrow: a nuisance.

Three arrows: unmanageable (try flip).

Every function you have written has an arrow for each parameter.

And yet

The typing of Exercise 1.3(b) is this proof, line for line: (S) for (II), (K) for (I).

A compiler produces that typing in a microsecond.

Something finds these lines. The next section says what.

SECTION 2.7

The Deduction Theorem

The clerk may not assume. It does not need to.

Theorem 2.7: Deduction Theorem

THEOREM 2.7 · DEDUCTION THEOREM

For every set of formulas Γ and all formulas A, B :

$$\Gamma \cup \{A\} \vdash B \iff \Gamma \vdash A \rightarrow B$$

Whenever B has a proof from Γ together with A , the implication $A \rightarrow B$ has a proof from Γ alone.

HOW IT IS PROVED

By a **transformation of proofs**:
rewrite a proof of B that uses A ,
line by line, into one that does not.

2.7 · THE DEDUCTION THEOREM

Right to left: one use of modus ponens

Suppose $\Gamma \vdash A \rightarrow B$. Show $\Gamma \cup \{A\} \vdash B$.

1. Take a proof of $A \rightarrow B$ from Γ . It is also a proof from $\Gamma \cup \{A\}$, by monotonicity.
2. Add the line A , justified as a hypothesis: A is a member of $\Gamma \cup \{A\}$.
3. Add the line B , by modus ponens on $A \rightarrow B$ and A .

The easy direction: use the implication.

Left to right: the plan

Let $A_1, \dots, A_n = B$ be a proof of B from $\Gamma \cup \{A\}$.

Go down it one line at a time. For each line A_k , write out a proof of $A \rightarrow A_k$ from Γ *alone*, using the proofs already written for the earlier lines.

At the last line, $A_n = B$, we have a proof of $A \rightarrow B$.

INDUCTION ON k

Hypothesis: for every $j < k$, a proof of $A \rightarrow A_j$ from Γ has been written.

To show: a proof of $A \rightarrow A_k$ from Γ , built on top of those.

2.7 · THE DEDUCTION THEOREM

How A_k is justified: three cases A_k IS ...an axiom, or a member of Γ A itselfby modus ponens from A_j and $A_i = A_j \rightarrow A_k$

WHAT THE THEOREM WRITES

three lines

the five-line proof of $A \rightarrow A$

three lines, on top of two proofs already written

A hypothesis is a member of Γ or A itself. An axiom and a member of Γ are handled alike.

Case 1: an axiom or a member of Γ

A_k is a legitimate line of a proof from Γ . Write

LINE	FORMULA	JUSTIFICATION
a.	A_k	as before
b.	$A_k \rightarrow A \rightarrow A_k$	(I) [A_k , A]
c.	$A \rightarrow A_k$	MP b, a

IN WORDS

A line we did not need A for stays true when we assume A as well. (I) says exactly that.

Case 2: A_k is A itself

Then $A \rightarrow A_k$ is $A \rightarrow A$. The five-line proof of Section 2.4, with A for p , proves it from no hypotheses at all, so in particular from Γ .

THE FIVE LINES

(II) $[A, A \rightarrow A, A]$

(I) $[A, A \rightarrow A]$

MP

(I) $[A, A]$

MP

This is where the proof of $p \rightarrow p$ was waiting to be used.

2.7 · THE DEDUCTION THEOREM

Case 3: modus ponens

There are earlier lines A_j and $A_i = A_j \rightarrow A_k$. By the induction hypothesis we already hold proofs of $A \rightarrow A_j$ and of $A \rightarrow A_j \rightarrow A_k$.

LINE FORMULA

JUSTIFICATION

- | | | |
|----|---|---|
| a. | $(A \rightarrow A_j \rightarrow A_k) \rightarrow (A \rightarrow A_j) \rightarrow A \rightarrow A_k$ | (II) $[A, A_j, A_k]$ |
| b. | $(A \rightarrow A_j) \rightarrow A \rightarrow A_k$ | MP a, $(A \rightarrow A_j \rightarrow A_k)$ |
| c. | $A \rightarrow A_k$ | MP b, $(A \rightarrow A_j)$ |

Three lines, on top of the two proofs already written. Only this case uses the induction hypothesis.

Why it ends

- ▶ Concatenate the pieces in order: a proof from Γ whose last line is $A \rightarrow A_n$, that is, $A \rightarrow B$.
- ▶ Only two schemas were needed: (I) in the first case, (II) in the third, and both inside the five-line proof the second case borrows.

That is why (I) and (II) are the axioms: they are exactly what this construction uses.

2.7 · THE DEDUCTION THEOREM

Run it (1): the input

Take $\Gamma = \{p \rightarrow q\}$ and $A = p$. A proof of q from $\Gamma \cup \{p\}$:

LINE	FORMULA	JUSTIFICATION
A_1	$p \rightarrow q$	member of Γ
A_2	p	A itself
A_3	q	MP A_1, A_2

Goal: a proof of $p \rightarrow q$ from Γ alone. Three lines, three cases.

2.7 · THE DEDUCTION THEOREM

Run it (2): lines A_1 and A_2

LINE FORMULA

JUSTIFICATION

 $A_1 = p \rightarrow q$: **case 1**1. $p \rightarrow q$

hypothesis

2. $(p \rightarrow q) \rightarrow p \rightarrow (p \rightarrow q)$ (I) $[p \rightarrow q, p]$ 3. $p \rightarrow (p \rightarrow q)$ MP 2, 1 a proof of $A \rightarrow A_1$ $A_2 = p$: **case 2**4. $(p \rightarrow (p \rightarrow p) \rightarrow p) \rightarrow (p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$ (II) $[p, p \rightarrow p, p]$ 5. $p \rightarrow (p \rightarrow p) \rightarrow p$ (I) $[p, p \rightarrow p]$ 6. $(p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$

MP 4, 5

7. $p \rightarrow (p \rightarrow p)$ (I) $[p, p]$ 8. $p \rightarrow p$ MP 6, 7 a proof of $A \rightarrow A_2$

2.7 · THE DEDUCTION THEOREM

Run it (3): line A_3

$A_3 = q$ is modus ponens on $A_1 = A_2 \rightarrow A_3$ and A_2 . We hold $p \rightarrow p$ (line 8) and $p \rightarrow (p \rightarrow q)$ (line 3).

LINE	FORMULA	JUSTIFICATION
9.	$(p \rightarrow p \rightarrow q) \rightarrow (p \rightarrow p) \rightarrow p \rightarrow q$	(II) $[p, p, q]$
10.	$(p \rightarrow p) \rightarrow p \rightarrow q$	MP 9, 3
11.	$p \rightarrow q$	MP 10, 8

Eleven lines: a proof of $p \rightarrow q$ from $\{p \rightarrow q\}$, with no use of the hypothesis p .

2.7 · THE DEDUCTION THEOREM

Run it (4): the whole output

LINE FORMULA

JUSTIFICATION

- | | |
|--|--------------------------------|
| 1. $p \rightarrow q$ | hypothesis |
| 2. $(p \rightarrow q) \rightarrow p \rightarrow (p \rightarrow q)$ | (I) $[p \rightarrow q, p]$ |
| 3. $p \rightarrow (p \rightarrow q)$ | MP 2, 1 |
| 4. $(p \rightarrow (p \rightarrow p) \rightarrow p) \rightarrow (p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$ | (II) $[p, p \rightarrow p, p]$ |
| 5. $p \rightarrow (p \rightarrow p) \rightarrow p$ | (I) $[p, p \rightarrow p]$ |
| 6. $(p \rightarrow (p \rightarrow p)) \rightarrow p \rightarrow p$ | MP 4, 5 |
| 7. $p \rightarrow (p \rightarrow p)$ | (I) $[p, p]$ |
| 8. $p \rightarrow p$ | MP 6, 7 |
| 9. $(p \rightarrow p \rightarrow q) \rightarrow (p \rightarrow p) \rightarrow p \rightarrow q$ | (II) $[p, p, q]$ |
| 10. $(p \rightarrow p) \rightarrow p \rightarrow q$ | MP 9, 3 |
| 11. $p \rightarrow q$ | MP 10, 8 |

2.7 · THE DEDUCTION THEOREM

Run it with the programs on

Context $f : p \rightarrow q$. Write each line's program beside it.

LINES PROGRAM

1–3 case 1: $f, K, K f$

4–8 case 2: SKK

9 case 3: S

10 $S(K f)$

11 $S(K f)(SKK)$

THE OUTPUT

$S(K f)(SKK) : p \rightarrow q$

The parameter x of $f x$ has gone. No variable is left but f .

A compiler, found by accident

THE DEFINITION

$$g\ x := f\ x$$

WHAT THE THEOREM COMPILES IT TO

$$g := S\ (K\ f)\ I$$

with $I = S\ K\ K$.

- ▶ Case 1 writes a K (the line did not depend on x).
- ▶ Case 2 writes an I (the line *is* x).
- ▶ Case 3 writes an S (an application whose parts both depend on x).

2.7 · THE DEDUCTION THEOREM

The compiler's missing rule is a theorem

To type a definition $f\ x = M$, a checker wants to say: suppose x has type A ; if on that supposition M has type B , then f has type $A \rightarrow B$.

$$\Gamma, x:A \vdash M:B \implies \Gamma \vdash f:A \rightarrow B$$

PROGRAMS RUBBED OUT

If $\Gamma \cup \{A\} \vdash B$ then $\Gamma \vdash A \rightarrow B$.

That *is* the Deduction Theorem, left to right.

The rule the checker lacked is not a rule but a theorem of the system it already has. And its proof is a construction.

This is the compiler promised in Theorem 1.8

- ▶ Run the construction with the programs left on. It compiles the parameter away.
- ▶ What comes out of $f\ x_1 \cdots x_n := M$ is a term with *no variables at all*: it runs the same way on every argument and has the type the definition should have.
- ▶ That is why (I) and (II) are the axioms: the types of the two programs that survive when every variable has been compiled away. Chosen, long before anyone compiled anything, so that this theorem would hold.

Chapter 3 sets the two languages side by side and finds them the same language.

2.7 · THE DEDUCTION THEOREM

You may assume the premise

Read the theorem the other way round. To prove $A \rightarrow B$, *assume* A and derive B .

Every such proof can be rewritten, mechanically, into a proof of the implication with no assumption.

The clerk may write “suppose A ” — not because the rules allow it, but because a theorem says the supposition can always be compiled away.

Worked: hypothetical syllogism by assuming

Assume $q \rightarrow r$, $p \rightarrow q$ and p . Derive r .

LINE	FORMULA	JUSTIFICATION
------	---------	---------------

1.	$q \rightarrow r$	hypothesis
----	-------------------	------------

2.	$p \rightarrow q$	hypothesis
----	-------------------	------------

3.	p	hypothesis
----	-----	------------

4.	q	MP 2, 3
----	-----	---------

5.	r	MP 1, 4
----	-----	---------

► Discharge p : $\{q \rightarrow r, p \rightarrow q\} \vdash p \rightarrow r$.

► Discharge $p \rightarrow q$: $\{q \rightarrow r\} \vdash (p \rightarrow q) \rightarrow p \rightarrow r$.

► Discharge $q \rightarrow r$: $\vdash (q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$.

The twenty-letter line, written for you

Three lines of thought and three discharges. The twenty-letter line of Section 2.6 is what the third discharge writes on your behalf.

Compare: seven lines found by staring, against five lines of ordinary thinking.

The price, and what it buys

THE PRICE

The rewritten proof is about three times as long: three lines for each original one. That is what every implication proved in the bare system costs, and why nobody writes such proofs by hand.

WHAT IT BUYS

$\vdash p \rightarrow p$ took a paragraph of searching. With the theorem it is immediate from $\{p\} \vdash p$, a one-line proof.

The theorem's own proof uses the five-line proof once, in the case $A_k = A$. What the theorem saves you is finding it again.

Exercise 2.1: $p \rightarrow p$ by the Deduction Theorem

Prove $\vdash p \rightarrow p$ by the Deduction Theorem, in one line of thought.

Then say which five lines the theorem writes on your behalf.

ONE MINUTE

Which case of the proof does it use?

Exercise 2.1: the answer

- ▶ $\{p\} \vdash p$ by a one-line proof: the hypothesis p .
- ▶ Discharge p . The last line of the proof is $A_k = A$ itself, so **case 2** applies.
- ▶ Case 2 writes the five-line proof of Section 2.4, with A for p : (II), (I), MP, (I), MP.

One line of thought, and the theorem does the rest.

Exercise 2.2: hypothetical syllogism, flipped

Prove $\vdash (p \rightarrow q) \rightarrow (q \rightarrow r) \rightarrow p \rightarrow r$ by the Deduction Theorem.

Then start to prove it from the schemas alone, following the transformation, and stop when you can see how long it will be.

TWO MINUTES

Hypotheses: $p \rightarrow q, q \rightarrow r, p$.

Exercise 2.2: the answer

- ▶ From $p \rightarrow q$, $q \rightarrow r$ and p : MP gives q , MP gives r . Five lines.
- ▶ Discharge p , then $q \rightarrow r$, then $p \rightarrow q$.
- ▶ Each discharge roughly triples the proof it is applied to, so the bare proof runs to several dozen lines, its first line an instance of (II) longer than the one in Section 2.6.

Exercise 2.3: flip

Flip swaps the order of two arguments. Its type is $(p \rightarrow q \rightarrow r) \rightarrow q \rightarrow p \rightarrow r$.

1. Prove $\{p \rightarrow q \rightarrow r\} \vdash q \rightarrow p \rightarrow r$ from the schemas and modus ponens, without the Deduction Theorem. (About nine lines. The instance of (II) you need has q as its first letter.)
2. Prove the closed formula by the Deduction Theorem. Roughly how long would (1), put through the construction, be?

FIVE MINUTES

Part (2) first. Then see why (1) is hard.

2.7 · THE DEDUCTION THEOREM

Exercise 2.3: the answer to (1)

LINE FORMULA

1. $p \rightarrow q \rightarrow r$
2. $(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$
3. $(p \rightarrow q) \rightarrow p \rightarrow r$
4. $((p \rightarrow q) \rightarrow p \rightarrow r) \rightarrow q \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$
5. $q \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$
6. $(q \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r) \rightarrow (q \rightarrow p \rightarrow q) \rightarrow q \rightarrow p \rightarrow r$
7. $(q \rightarrow p \rightarrow q) \rightarrow q \rightarrow p \rightarrow r$
8. $q \rightarrow p \rightarrow q$
9. $q \rightarrow p \rightarrow r$

JUSTIFICATION

hypothesis
(II) $[p, q, r]$
MP 2, 1
(I) $[(p \rightarrow q) \rightarrow p \rightarrow r, q]$
MP 4, 3
(II) $[q, p \rightarrow q, p \rightarrow r]$
MP 6, 5
(I) $[q, p]$
MP 7, 8

Exercise 2.3: the answer to (2)

- ▶ Assume $p \rightarrow q \rightarrow r$, q and p . MP gives $q \rightarrow r$; MP gives r .
- ▶ Discharge p , then q , then $p \rightarrow q \rightarrow r$: that proves $\vdash (p \rightarrow q \rightarrow r) \rightarrow q \rightarrow p \rightarrow r$.
- ▶ Each discharge roughly triples the proof beneath it, so the nine lines of (1), put through the construction, become some hundreds.

Five lines of thought against some hundreds of lines on the page. That is the Deduction Theorem's whole value.

Exercise 2.4: write a checker

Write a program that takes a list of formulas, each annotated (I), (II), hypothesis or MP i, j , and a set of hypotheses, and returns whether it is a proof in the sense of Definition 2.3.

What is its running time in the length of the proof?

THEN

Which lines of your checker are rule (var) and which are rule (app) of a type checker? What would you add to check typings instead?

2.7 · THE DEDUCTION THEOREM

In one slide

- ▶ A **proof** is a finite list of formulas, each an axiom, a hypothesis, or modus ponens from earlier lines. Checking one is pattern matching.
- ▶ A **typing is a proof** with the programs written beside the lines: K, S are the two schemas, application is modus ponens, a variable is a hypothesis. Running a program removes detours.
- ▶ The rule the compiler needed for a parameter is the **Deduction Theorem**: $\Gamma \cup \{A\} \vdash B$ iff $\Gamma \vdash A \rightarrow B$. Its proof is the algorithm that compiles the parameter away.

How did the compiler know? *It checked a proof.*

Where this leaves us

We have been compiling definitions away because that is all the four rules could type. But the Deduction Theorem says the case for parameters is admissible: adding it as a fifth case would let nothing new be typed.

So add it, and let the parameter stay.

CHAPTER 3

The λ put back. Three rules, two ours and one a theorem of ours — and a mathematician has been using them since 1934.

READING

Sources

- ▶ Open Logic Project, *Axiomatic derivations*. Free to use. Its remark that the schemas were chosen for the Deduction Theorem is the one this chapter turns into a compiler.
- ▶ Curry and Feys, *Combinatory Logic*. 1958. Where K, S meet (I), (II) in print.
- ▶ Turner (1979). The Deduction Theorem's construction as a working compiler.
- ▶ Wadler, *Propositions as types*. CACM 2015. The whole story.
- ▶ Turing (1936); Post (1936). The clerk and the worker.

Thank you.