COMP0009 Logic — Problem Sheet 2

Part A — Charlie's certificates

The story so far. On Sheet 1, Alice sent Bob ten coins over a network where nobody trusts anybody. The coins were locked behind a small program, a *script*, and every computer on the network, Charlie's among them, ran the script on Bob's evidence: if the run ended in *true*, the coins moved to Bob. Mallory attacked twice, once with a script that stopped only after building an enormous answer, and once with a script that never stopped at all. Charlie's defence was a *typing*: a short numbered list of lines, each justified by (K), (S), (var), (given) or (app), which he checked before running anything.

Now the network changes its rules. A payment no longer arrives with a script to run. It arrives with a *certificate*: a short numbered list of lines that shows, one step at a time, why the coins may move.

Charlie's computer does not need to understand a certificate. It checks each line against a few fixed shapes, the way a spell-checker checks words against a dictionary, and it accepts the payment only if every line passes. Checking is quick, and needs no cleverness. Writing a certificate is another matter.

In the four scenes below you will check certificates, find out that every certificate is a script in disguise, see why some certificates are very hard to write, and meet the machine that writes them for you.

Marks are in square brackets; the whole of Part A is worth 25.

A1. Charlie checks a certificate.

A certificate is made of *statements*. Each letter stands for a yes-or-no claim about a payment:

y:	Alice has said yes	w:	the password is right
k:	the key is right	m:	the coins may move

Longer statements are built with one connective, $A \to B$, read "if A then B". The arrow groups to the right, so $w \to k \to m$ means $w \to (k \to m)$: "if the password is right, then, if the key is right, the coins may move". (These are the types of Sheet 1, with letters in place of Int and $String$.)

The *rulebook* lists the statements that everyone on the network has agreed to. Bob can collect Alice's coins in two ways, so the rulebook has two rules:

$$y \to m \qquad\qquad w \to k \to m.$$

The *evidence* is what Bob tells the network: for example, w and k.

A certificate is a numbered list of statements. Each line carries, on its right, one of three justifications:

- **hypothesis**: the line is a rule of the rulebook, or a piece of evidence.
- **MP** i**,** j: line i is $A \to B$, line j is A, both are earlier, and this line is B. (MP stands for *modus ponens*: from "if A then B" and "A", write "B".)
- **(I)** or **(II)**: the line has one of two fixed shapes, with any statements at all in place of A, B and C:

  (I) $\quad A \to B \to A$ $\qquad\qquad$ (II) $\quad (A \to B \to C) \to (A \to B) \to A \to C$

  In square brackets after (I) or (II), write what was put in place of A, B (and C), in that order.

(I) says: if A holds, it still holds whatever else does. (II) says: if, given A, B implies C, and, given A, B holds, then, given A, C holds.

The last line is what the certificate proves. Logicians call a certificate a *proof*, and the rulebook and the evidence its *hypotheses*; they write $\Gamma \vdash_{\mathrm{HJ}} A$ when there is a proof of A from the hypotheses Γ.

Example. Bob's certificate that the coins may move, from the rulebook and the evidence w, k:

1. $w \to k \to m$	hypothesis
2. w	hypothesis
3. $k \to m$	MP 1, 2
4. k	hypothesis
5. m	MP 3, 4

To check line 3, Charlie looks at the lines it cites. Line 1 is $w \to (k \to m)$, which is "line 2 $\to$ line 3". So MP 1, 2 is correct. Line 5 is checked the same way.

And a shape: $(k \to m) \to y \to (k \to m)$ is an instance of (I) $[k \to m,\ y]$, because it is $A \to B \to A$ with $k \to m$ for A and y for B.

(a) Each line below is given only by its statement. Fill in its reason: (I) or (II) with what was put in place of the letters, or "neither". [2]

(i) $m \to k \to m$ _____

(ii) $m \to m \to k$ _____

(iii) $(w \to k) \to y \to (w \to k)$ _____

(iv) $(y \to w \to k) \to (y \to w) \to y \to k$ _____

(v) $(y \to w \to k) \to (y \to k) \to y \to w$ _____

Hint: cut the statement at the arrows that are not inside brackets. For (I), the first piece is A, the second is B, and everything after them must be A again. For (II), the first three pieces are $A \to B \to C$, $A \to B$ and A, and everything after them must be C.

(b) Now the other way round. Each line below is given only by its reason. Fill in its statement. [1]

(i) _____ (I) $[w,\ k]$

(ii) _____ (I) $[m,\ y \to k]$

Then fill in the gaps in this certificate, from the hypothesis $y \to k \to m$:

1. _____	(II) $[y,\ k,\ m]$
2. $y \to k \to m$	hypothesis
3. _____	MP 1, 2

Hint: copy the shape, and put in place of each letter what the brackets say. Anything with an arrow in it goes in brackets.

(c) Mallory has only the key, so her evidence is k. She sends this certificate:

1. $y \to m$	hypothesis
2. k	hypothesis
3. $k \to y \to k$	(I) $[k,\ y]$
4. $y \to k$	MP 3, 2
5. y	MP 4, 2
6. m	MP 1, 5

Charlie rejects it. Find the first line that fails, and say why. Then say, in plain English, what the argument of that line is, and why nobody should believe it. [2]

Hint: check every line, including the ones that look odd.

(d) Write two certificates for $k \to m$, each from the rulebook and some evidence. [2]

(i) The evidence is w. What does $k \to m$ now say about Bob?

(ii) The evidence is y. What does $k \to m$ say now?

Hint: in (ii), first get m. Then (a)(i) is useful.

A2. Every certificate is a script.

Charlie has seen lists like these before. On Sheet 1, a *typing* was a numbered list of lines, each justified in one of five ways. Leave out (given), the ready-made facts about numbers and words, and put the other four beside a certificate:

a line of a typing	its reason	a line of a certificate	its reason
$\mathsf{K} : A \to B \to A$	(K)	$A \to B \to A$	(I)
S: the statement on the right	(S)	$(A \to B \to C) \to (A \to B) \to A \to C$	(II)
$x : A$	(var)	A	hypothesis
$M\,N : B$	(app) i, j	B	MP i, j

They are the same list. Rub the scripts out of a typing, and what is left is a certificate. Write scripts onto a certificate, and what you get is a typing. To write them on, which is called *decorating*:

- give each hypothesis a name, and put that name on its lines;
- put K on every (I) line and S on every (II) line;
- on a line MP i, j, put the script of line i applied to the script of line j.

Example. The identity script $\mathsf{S\,K\,K}$ of Sheet 1 (it runs $\mathsf{S\,K\,K}\,x \rightsquigarrow \mathsf{K}\,x\,(\mathsf{K}\,x) \rightsquigarrow x$) has this typing. It is a decorated certificate with no hypotheses:

1. $\mathsf{S} : (p \to (p \to p) \to p) \to (p \to (p \to p)) \to p \to p$	(II) $[p,\ p \to p,\ p]$
2. $\mathsf{K} : p \to (p \to p) \to p$	(I) $[p,\ p \to p]$
3. $\mathsf{S\,K} : (p \to (p \to p)) \to p \to p$	MP 1, 2
4. $\mathsf{K} : p \to (p \to p)$	(I) $[p,\ p]$
5. $\mathsf{S\,K\,K} : p \to p$	MP 3, 4

Cover the scripts, and you see a certificate that p implies p. Cover the statements, and you see how $\mathsf{S\,K\,K}$ is built. The script that hands back its input *is* the certificate that p implies p.

(a) Decorate Bob's certificate from the example in A1. Call the hypotheses $rule2 : w \to k \to m$, $pw : w$ and $key : k$. Write the script on every line. Which script is on the last line, and what does it say, in words? [2]

Hint: line 3 gets the script $rule2\ pw$.

(b) The rulebook gains two rules, with names:

$sign : y \to s$ (if Alice says yes, the payment is signed)
$pay : s \to m$ (signed payments may move)

Here is a certificate for $y \to m$ from these two hypotheses:

1. $s \to m$ — hypothesis
2. $(s \to m) \to y \to (s \to m)$ — (I) $[s \to m,\ y]$
3. $y \to (s \to m)$ — MP 2, 1
4. $(y \to s \to m) \to (y \to s) \to y \to m$ — (II) $[y,\ s,\ m]$
5. $(y \to s) \to y \to m$ — MP 4, 3
6. $y \to s$ — hypothesis
7. $y \to m$ — MP 5, 6

(i) Decorate it. Which script is on line 7? [2]

(ii) Run that script on an argument x, one step per line, using $\mathsf{K}\,M\,N \rightsquigarrow M$ and $\mathsf{S}\,M\,N\,P \rightsquigarrow M\,P\,(N\,P)$. What does the script do to x? [1]

Hint: an (I) or (II) line gets K *or* S*, never a name.*

(c) Mallory now runs one of the computers too, and she pads the certificates she passes on, to waste Charlie's time. Here is one of Bob's payments as she passes it on: a certificate for m from the rule $y \to m$ and the evidence y and k.

1. $y \to m$ — hypothesis
2. y — hypothesis
3. m — MP 1, 2
4. $m \to k \to m$ — (I) $[m,\ k]$
5. $k \to m$ — MP 4, 3
6. k — hypothesis
7. m — MP 5, 6

Decorate it, calling the hypotheses $rule1 : y \to m$, $ok : y$ and $key : k$. Which script is on line 7? Run it for one step. Which lines of Mallory's certificate does the shorter script still need? [2]

A3. Certificates with no evidence.

Some statements need no rulebook and no evidence: they hold for every payment, whatever anyone says. A certificate with no hypotheses at all proves a *theorem*, and we write $\vdash_{\mathrm{HJ}} A$. You have met one already: $\vdash_{\mathrm{HJ}} p \to p$, in A2.

Alice wants the network to know, once and for all, that two rules always chain:

$$(s \to m) \to (y \to s) \to y \to m.$$

"If signed payments may move, then, if Alice's yes means signed, Alice's yes means the coins may move." Here is a certificate with no hypotheses. To keep it short, P stands for $s \to m$, Q for $y \to s \to m$, and R for $(y \to s) \to y \to m$, so the goal is $P \to R$.

1. $(P \to Q \to R) \to (P \to Q) \to P \to R$ — (II) $[P,\ Q,\ R]$
2. $Q \to R$, that is, $(y \to s \to m) \to (y \to s) \to y \to m$ — (II) $[y,\ s,\ m]$
3. $(Q \to R) \to P \to (Q \to R)$ — (I) $[Q \to R,\ P]$
4. $P \to Q \to R$ — MP 3, 2
5. $(P \to Q) \to P \to R$ — MP 1, 4
6. $P \to Q$, that is, $(s \to m) \to y \to (s \to m)$ — (I) $[s \to m,\ y]$
7. $P \to R$ — MP 5, 6

Every line checks. But here is line 1 written out in full:

$$((s \to m) \to (y \to s \to m) \to (y \to s) \to y \to m)$$
$$\to ((s \to m) \to y \to s \to m) \to (s \to m) \to (y \to s) \to y \to m$$

Twenty letters and nineteen arrows. Nobody finds that by thinking about what "if" means.

The way out is a theorem about certificates, the *Deduction Theorem*:

if Γ *together with* A *gives a certificate for* B*, then* Γ *alone gives a certificate for* $A \to B$*.*

It is a recipe. To get a certificate for $A \to B$: *assume* A (add it to the hypotheses), write a certificate for B, and then *discharge* A (take it out of the hypotheses, and put it in front of an arrow).

Example. Alice's theorem by the recipe. Assume $s \to m$, $y \to s$ and y:

1. $y \to s$	hypothesis
2. y	hypothesis
3. s	MP 1, 2
4. $s \to m$	hypothesis
5. m	MP 4, 3

So $s \to m,\ y \to s,\ y \vdash_{\mathrm{HJ}} m$. Now discharge, the last assumption first:

discharge y:	$s \to m,\ y \to s \vdash_{\mathrm{HJ}} y \to m$
discharge $y \to s$:	$s \to m \vdash_{\mathrm{HJ}} (y \to s) \to y \to m$
discharge $s \to m$:	$\vdash_{\mathrm{HJ}} (s \to m) \to (y \to s) \to y \to m$

Five obvious lines and three discharges, in place of the twenty-letter line.

(a) Bob wants the network to accept the key before the password. Use the recipe to show

$$\vdash_{\mathrm{HJ}} (w \to k \to m) \to k \to w \to m.$$

Write your assumptions, the certificate from them, and the three discharges in order. [2]

(b) Take your certificate from (a), but discharge in the order $w \to k \to m$, then k, then w. Which theorem do you get? Say in words what it means. [1]

(c) The recipe does not break the rules: a certificate may still use only hypotheses, (I), (II) and MP. The theorem is proved by a *machine* that rewrites a certificate which uses A as a hypothesis into one that does not. It goes down the certificate, and for each line C it writes a few lines ending in $A \to C$:

- C **is a hypothesis other than A, or an (I) or (II) line.** Write C; then $C \to A \to C$, by (I) $[C,\ A]$; then $A \to C$, by MP. *Three lines.*
- C **is A itself.** Write the five lines of A2's example, with A in place of p. The last of them is $A \to A$. *Five lines.*
- C **came by MP from $D \to C$ and D.** The machine has already written $A \to D \to C$ and $A \to D$. Write $(A \to D \to C) \to (A \to D) \to A \to C$, by (II) $[A,\ D,\ C]$; then $(A \to D) \to A \to C$, by MP; then $A \to C$, by MP. *Three lines.*

Run the machine on this certificate for m from $y \to m$ and y, discharging y:

1.	$y \to m$	hypothesis
2.	y	hypothesis
3.	m	MP 1, 2

(i) For each of the three lines, say which of the machine's cases it falls under. For line 3, say what D and C are.

(ii) Here is what the machine writes. Fill in the gaps: the reasons come from (i), and writing a statement from its reason is what you did in A1(b).

1.	$y \to m$	hypothesis
2.	_____	(I) $[y \to m,\ y]$
3.	_____	MP 2, 1
4–8.	the five lines of A2's example, with y for p; line 8 is $y \to y$	
9.	_____	(II) [____, ____, ____]
10.	_____	MP 9, 3
11.	$y \to m$	MP 10, 8

(iii) The original certificate had 3 lines, and the machine's has 11. Your certificate in (a) had about 5 lines, and goes through the machine three times, once for each discharge. Roughly how long is the result? [2]

A4. The machine is a compiler.

On Sheet 1, Charlie's checker had a line for K, for S, for a variable, for a ready-made fact and for an application, but none for a *definition*, such as

$$release\ x := rule1\ x,$$

whose placeholder x stands for any evidence that Alice said yes. Sheet 1 also claimed that every definition on it could be written with K and S alone, without placeholders. The machine of A3 is the reason. In its first case it writes an (I) line, which decorates to K; in its third case it writes a (II) line, which decorates to S.

Example. Decorate the certificate that the machine was given in A3(c), calling its hypotheses $rule1 : y \to m$ and $x : y$:

1.	$rule1 : y \to m$	hypothesis
2.	$x : y$	hypothesis
3.	$rule1\ x : m$	MP 1, 2

The last line is the body of *release*. Discharging y is removing the placeholder x.

(a) Now decorate the machine's output, the eleven lines of A3(c), calling the hypothesis $rule1 : y \to m$. (The hypothesis y has been discharged, so x appears nowhere.) Which script is on line 11? [2]

Hint: lines 4–8 are A2's example, so line 8 gets $\mathsf{S}\,\mathsf{K}\,\mathsf{K}$.

(b) Run your script from (a) on x, one step per line. Which definition has the machine compiled? [2]

(c) On Sheet 1 the library had $twice\ f\ x \coloneqq f\,(f\,x)$. With $f : p \to p$ and $x : p$, its body has this decorated certificate:

1.	$f : p \to p$	hypothesis
2.	$x : p$	hypothesis
3.	$f\,x : p$	MP 1, 2
4.	$f\,(f\,x) : p$	MP 1, 3

Run the machine on the scripts alone, discharging x, using these three rules:

- a hypothesis line other than x, with script M, becomes $\mathsf{K}\,M$;
- the line x itself becomes $\mathsf{S}\,\mathsf{K}\,\mathsf{K}$;
- a line $M\,N$ by MP becomes $\mathsf{S}\,M'\,N'$, where M' and N' are what the lines of M and N became.

What does line 4 become? Check that it runs, on x, to $f\,(f\,x)$. [2]

Hint: you may use your answer to (b), with f *in place of* $rule1$.

Charlie's certificates are real. In 1996 George Necula and Peter Lee proposed *proof-carrying code*: a program sent to your computer arrives with a proof that it is safe to run, and your computer checks the proof, quickly and without trusting the sender, before it runs the program. Mathematics is now checked the same way. In 2024 an AI system from Google DeepMind solved problems from the International Mathematical Olympiad by writing proofs in a language called Lean, and nobody had to trust the AI's reasoning, because Lean's small checker went through every line. Finding a certificate can take great cleverness; checking one takes none. And your compiler has been doing this all along: each time it accepts a function with a parameter, the theorem of A3 is the reason it may.

Part B — Practice

Practice, for when you want more of the same. The last two, B13 and B14, go beyond this week: try them only if you are curious. The rules you need:

- Statements: letters, and $A \to B$. The arrow groups to the right: $A \to B \to C$ means $A \to (B \to C)$.
- A certificate (proof) of A from hypotheses Γ is a numbered list ending in A, each line a hypothesis, an instance of (I) $A \to B \to A$ or (II) $(A \to B \to C) \to (A \to B) \to A \to C$, or MP i, j from earlier lines i (which is $D \to E$) and j (which is D). We write $\Gamma \vdash_{\mathrm{HJ}} A$; with no hypotheses, $\vdash_{\mathrm{HJ}} A$.
- Decorating: a hypothesis gets its name, (I) gets K, (II) gets S, MP i, j gets (script i) (script j). $\mathsf{K}\,M\,N \rightsquigarrow M$, $\mathsf{S}\,M\,N\,P \rightsquigarrow M\,P\,(N\,P)$.
- The Deduction Theorem: Γ with A proves B exactly when Γ proves $A \to B$; the machine that turns the first into the second is in A3(c).

In the drills p, q, r, s are letters.

B1. Insert the brackets that right-grouping hides.

(i) $p \to q \to r \to s$ _____

(ii) $(p \to q) \to r \to s$ _____

(iii) $p \to (q \to r) \to s$ _____

Then say which two of these are the same statement:

(iv) $p \to (q \to r)$

(v) $(p \to q) \to r$

(vi) $p \to q \to r$

B2. Fill in the reason for each line: (I) or (II) with what was put in place of the letters, or "neither".

(i) $q \to p \to q$ _____

(ii) $(p \to q) \to (p \to q) \to (p \to q)$ _____

(iii) $p \to (q \to r) \to p$ _____

(iv) $(p \to q) \to p \to q$ _____

(v) $(q \to q \to q) \to (q \to q) \to q \to q$ _____

(vi) $(p \to q \to r) \to (p \to r) \to p \to r$ _____

B3. Fill in the statement for each reason.

(i) _____ (I) $[p,\ p]$

(ii) _____ (I) $[q \to p,\ r]$

(iii) _____ (II) $[p,\ p \to q,\ r]$

Then fill in the gaps in this certificate, from the hypothesis $p \to q \to r$:

1. $p \to q \to r$ hypothesis
2. _____ (II) $[p,\ q,\ r]$
3. _____ MP 2, 1
4. _____ (I) $[(p \to q) \to p \to r,\ q]$
5. _____ MP 4, 3

B4. Is each list a certificate for its last line from the hypotheses given? If not, find the first line that fails, and say why.

(i) Hypotheses $p \to q$, $q \to r$ and p:

1.	p	hypothesis
2.	$p \to q$	hypothesis
3.	q	MP 2, 1
4.	$q \to r$	hypothesis
5.	r	MP 4, 3

(ii) Hypotheses $p \to q$ and q:

1.	$p \to q$	hypothesis
2.	q	hypothesis
3.	p	MP 1, 2

(iii) Hypothesis q:

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

(iv) Hypotheses $p \to q$ and p:

1.	$p \to q$	hypothesis
2.	q	MP 1, 3
3.	p	hypothesis

B5. Write a certificate for each.

(i) $p \to q \to r,\ p,\ q \vdash_{\mathrm{HJ}} r$

(ii) $p \to q,\ q \to r,\ p \vdash_{\mathrm{HJ}} r$

(iii) $q \vdash_{\mathrm{HJ}} p \to q$

(iv) $p \to q \to r,\ q \vdash_{\mathrm{HJ}} p \to r$ (seven lines; you need one (I) and one (II) line)

B6. Gluing certificates. A2(b)'s certificate, with p, q, r for y, s, m, shows $p \to q,\ q \to r \vdash_{\mathrm{HJ}} p \to r$ in seven lines; with r, s for q, r it shows $p \to r,\ r \to s \vdash_{\mathrm{HJ}} p \to s$.

(i) Explain how to glue the two into a certificate for $p \to q,\ q \to r,\ r \to s \vdash_{\mathrm{HJ}} p \to s$, and say how many lines it has.

(ii) Why is a certificate from Γ also a certificate from Γ with one more hypothesis added?

(iii) Can a certificate use infinitely many hypotheses? Why not?

B7. Rub out the scripts. On Sheet 1, B11 typed $\mathsf{K}\,\mathsf{K}$ like this:

1.	$\vdash_{\mathrm{KS}} \mathsf{K} : (A \to B \to A) \to C \to A \to B \to A$	(K)
2.	$\vdash_{\mathrm{KS}} \mathsf{K} : A \to B \to A$	(K)
3.	$\vdash_{\mathrm{KS}} \mathsf{K}\,\mathsf{K} : C \to A \to B \to A$	(app) 1, 2

Erase it, with p, q, r for A, B, C, and give the justification of each line. Then run $\mathsf{K}\,\mathsf{K}\,x\,y\,z$ and say what the script does.

B8. Flip, without the recipe. Here is a certificate for $p \to q \to r \vdash_{\mathrm{HJ}} q \to p \to r$:

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

Check lines 4 and 6. Decorate the certificate, calling the hypothesis f, and run the script on line 9 applied to b and then a. What does it do?

B9. Two detours. Decorate this certificate for q from $p \to q$, p, r, s, calling the hypotheses f, x, y, z, and run the script on its last line to the end. How many lines does the certificate for the final script have?

1.	$p \to q$	hypothesis
2.	p	hypothesis
3.	q	MP 1, 2
4.	$q \to r \to q$	(I) $[q,\ r]$
5.	$r \to q$	MP 4, 3
6.	r	hypothesis
7.	q	MP 5, 6
8.	$q \to s \to q$	(I) $[q,\ s]$
9.	$s \to q$	MP 8, 7
10.	s	hypothesis
11.	q	MP 9, 10

B10. Use the recipe of A3.

(i) $\vdash_{\mathrm{HJ}} p \to p$. Which lines does the machine write for you?
(ii) $\vdash_{\mathrm{HJ}} (p \to q) \to (q \to r) \to p \to r$
(iii) $\vdash_{\mathrm{HJ}} p \to (p \to q) \to q$
(iv) $p \to q \to r \vdash_{\mathrm{HJ}} q \to p \to r$ (compare B8)

B11. Run the machine of A3(c) on this certificate for $p \vdash_{\mathrm{HJ}} q \to p$, discharging p:

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

How many lines does the machine write, and what is the last line? Is there a shorter certificate for that? Decorate the machine's output and run the script on x and then y.

B12. In A3, line 1 of Alice's certificate was (II) $[P,\ Q,\ R]$ with P for $s \to m$, Q for $y \to s \to m$ and R for $(y \to s) \to y \to m$. Write it out with P, Q, R replaced, and confirm that it has twenty letters and nineteen arrows. Line 1 redoes, one level up and under the hypothesis P, one of the MP lines of A2(b)'s certificate. Which one?

B13. *(For the curious.)* A shortcut for the machine. Suppose a line C was obtained without using A anywhere above it. Show that, if the machine keeps the lines that obtained C, two more lines give $A \to C$, whatever kind of line C is. What does this shortcut do to scripts? Why does it make the machine's output shorter?

B14. *(For the curious.)* Write a checker: a program (in Python, Haskell, or careful pseudocode) that takes a list of statements, each with its justification, and a list of hypotheses, and says whether the list is a certificate. How long does it take, in the length of the list? Which part of your program corresponds to rule (var) of a type checker, and which to (app)?