Working with ProofWeb

1. Go to the web page

   `http://prover.cs.ru.nl/`

   (ProofWeb does not work in all browsers. Firefox should be okay.)

2. Click on the name of your course near the bottom of the page.

3. Log in using your username and password. (Ask your teacher for this information if you do not know it yet.)

4. Click on the button of the problem that you want to work on.

5. (Possibly select from the `Debug` menu the item

   `Toggle Electric Terminator`

   This will make the commands execute automatically every time you type a period character (‘`.`’). Else you will need to click on the down arrow or type *control-down* to execute commands.)

6. Click on the down arrow or type *control-down* until you have executed the

   `Proof.`

   line after the `Theorem` line. Now you will be able to work on a proof. When you are doing this, the incomplete version of the proof will be shown on the lower right.

7. Replace the

   `(*! prop_proof *)`

   comment with a sequence of tactics as described on the next pages. If you did not toggle the electric terminator, execute these tactics by clicking the down arrow repeatedly (or by typing *control-down* repeatedly.)

8. You can also enter tactics by selecting items from the `Template`, `Backward` and `Forward` menus. Replace the question marks (‘`?`’) by labels, formulas and terms. After you did so click the down arrow or type *control-down*.

9. You can edit the part of the text that has not yet been executed (the part that has been executed will be green). To undo steps in the proof click the up arrow or type *control-up*.

10. If you have problems with ProofWeb, or want to exchange experiences, use the ProofWeb form on the discussion board (‘Blackboard’) of your course.

Example

Formula

$$(A \wedge B) \rightarrow A$$

Exercise

```
(* Exercise 1 *)

Require Import ProofWeb.

Variables A B : Prop.

Theorem prop_001 : (A /\ B) -> A.
Proof.
(*! prop_proof *)

Qed.
```

Solution

```
(* Exercise 1 *)

Require Import ProofWeb.

Variables A B : Prop.

Theorem prop_001 : (A /\ B) -> A.
Proof.
imp_i H.
f_con_e1 H.
Qed.
```

Proof

1	$A \wedge B$	assumption
2	A	$\wedge e_1$ 1
3	$A \wedge B \rightarrow A$	$\rightarrow$i 1—2

Rendering in ProofWeb

1	H:	A ∧ B	assumption
2		A	$\wedge e_1$ 1
3		A ∧ B → A	→i 1-2

Formula syntax

$\bot$	`False`
$\top$	`True`
$\neg A$	`~` A
$A \land B$	A `/\` B
$A \lor B$	A `\/` B
$A \to B$	A `->` B
$A \leftrightarrow B$	A `<->` B
$\forall x\, A$	`all` x`,` A
$\exists x\, A$	`exi` x`,` A

When formulas that are not single identifiers are used as arguments of tactics, they need to be put in brackets.

The '`all`' and '`exi`' quantifiers bind more strongly than the built-in Coq quantifiers '`forall`' and '`exists`'.

The '`all`' and '`exi`' internally use a function '`_K`'. If through a bug in one of the tactics this function ever appears in a goal, one should use the tactic '`remove_K`' to get rid of it (and report the bug).

Tactics

The green 'H:' labels that occur in these descriptions are not part of the way proofs are written in Huth & Ryan, but are necessary for working in ProofWeb. They are the symbolic equivalents (which stay the same throughout the proof process) of the line numbers (which change all the time).

Structural tactics

`ass H`							
m	H:	A		m	H:	A	
		...					
n		A	$\longrightarrow$	n		A	copy m
`insert H` B							
						...	
				n	H:	B	
		...				...	
n		A	$\longrightarrow$	$n+1$		A	

Backward tactics

The tactic names may be prefixed with `b_...` to contrast them to the corresponding forward tactics.

Rules that are not intuitionistically valid are marked with a star. Rules that according to Huth & Ryan are derived rules are marked with a dagger.

conjunction introduction					
`con_i`					
				...	
			n	A	
				...	
	...		$n+1$	B	
n	$A \wedge B$	$\longrightarrow$	$n+2$	$A \wedge B$	$\wedge$i $n, (n+1)$
conjunction elimination left					
`con_e1` B					
				...	
	...		n	$A \wedge B$	
n	A	$\longrightarrow$	$n+1$	A	$\wedge e_1$ n

conjunction elimination right

`con_e2` A

					$\dots$	
	$\dots$		n		$A \wedge B$	
n	B	$\longrightarrow$	$n+1$		B	$\wedge e_2\ n$

disjunction introduction left

`dis_i1`

					$\dots$	
	$\dots$		n		A	
n	$A \vee B$	$\longrightarrow$	$n+1$		$A \vee B$	$\vee i_1\ n$

disjunction introduction right

`dis_i2`

					$\dots$	
	$\dots$		n		B	
n	$A \vee B$	$\longrightarrow$	$n+1$		$A \vee B$	$\vee i_1\ n$

disjunction elimination

`dis_e (`A` \/ `B`) H1 H2`

					$\dots$	
			n		$A \vee B$	
			$n+1$	`H1:`	A	assumption
					$\dots$	
			$n+2$		C	
			$n+3$	`H2:`	B	assumption
					$\dots$	
	$\dots$		$n+4$		C	
n	C	$\longrightarrow$	$n+5$		C	$\vee e\ n,(n+1)—(n+2),(n+3)—(n+4)$

implication introduction

`imp_i H`

			n	`H:`	A	assumption
					$\dots$	
	$\dots$		$n+1$		B	
n	$A \rightarrow B$	$\longrightarrow$	$n+2$		$A \rightarrow B$	$\rightarrow i\ n—(n+1)$

implication elimination

`imp_e` A

				$\dots$	
			n	$A \to B$	
				$\dots$	
	$\dots$		$n+1$	A	
n	B	$\longrightarrow$	$n+2$	B	$\to$e $n, (n+1)$

negation introduction

`neg_i H`

			n	`H:`	A	assumption
					$\dots$	
	$\dots$		$n+1$		$\bot$	
n	$\neg A$	$\longrightarrow$	$n+2$		$\neg A$	$\neg$i n—$(n+1)$

negation elimination

`neg_e` A

				$\dots$	
			n	$\neg A$	
				$\dots$	
	$\dots$		$n+1$	A	
n	$\bot$	$\longrightarrow$	$n+2$	$\bot$	$\neg$e $n, (n+1)$

falsum elimination

`fls_e`

				$\dots$	
	$\dots$		n	$\bot$	
n	A	$\longrightarrow$	$n+1$	A	$\bot$e n

truth introduction

`tru_i`

	$\dots$				
n	$\top$	$\longrightarrow$	n	$\top$	$\top$i n

double negation introduction†

`negneg_i`

				$\dots$	
	$\dots$		n	A	
n	$\neg\neg A$	$\longrightarrow$	$n+1$	$\neg\neg A$	$\neg\neg$i n

*double negation elimination**

`negneg_e`

				...	
	...		n	$\neg\neg A$	
n	A	$\longrightarrow$	$n+1$	A	$\neg\neg$e n

law of excluded middle†*

`LEM`

	...				
n	$A \vee \neg A$	$\longrightarrow$	n	$A \vee \neg A$	LEM

proof by contradiction†*

`PBC H`

			n	`H:`	$\neg A$	assumption
					...	
	...		$n+1$		$\bot$	
n	A	$\longrightarrow$	$n+2$		A	PBC n—$(n+1)$

modus tollens†

`MT` B

				...	
			n	$A \rightarrow B$	
				...	
	...		$n+1$	$\neg B$	
n	$\neg A$	$\longrightarrow$	$n+2$	$\neg A$	MT $n, (n+1)$

universal introduction

`all_i` y

				y		
					...	
	...		n		$A[y/x]$	
n	$\forall x\, A$	$\longrightarrow$	$n+1$		$\forall x\, A$	$\forall x$ i n—n

universal elimination

`all_e (all x,` A`)`

				...	
	...		n	$\forall x\, A$	
n	$A[t/x]$	$\longrightarrow$	$n+1$	$A[t/x]$	$\forall x$ e n

existential introduction

`exi_i` t

				$\dots$	
	$\dots$		n	$A[t/x]$	
n	$\exists x\, A$	$\longrightarrow$	$n+1$	$\exists x\, A$	$\exists x$ i n

existential elimination

`exi_e (exi x,` A`)` y `H`

					$\dots$	
			n		$\exists x\, A$	
				y		
			$n+1$	`H:`	$A[y/x]$	
					$\dots$	
	$\dots$		$n+2$		B	
n	B	$\longrightarrow$	$n+3$		B	$\exists x$ e n,$(n+1)$—$(n+2)$

equality introduction

`equ_i`

	$\dots$				
n	$t = t$	$\longrightarrow$	n	$t = t$	=i

equality elimination, simple version

`equ_e (`t_1 `=` t_2`)`

				$\dots$	
			n	$t_1 = t_2$	
				$\dots$	
	$\dots$		$n+1$	$A[t_1/x]$	
n	$A[t_2/x]$	$\longrightarrow$	$n+2$	$A[t_2/x]$	=e $n, (n+1)$

equality elimination, general version (t_2 *may occur in* A)

`equ_e' (`t_1 `=` t_2`) (fun x =>` A`)`

				$\dots$	
	$\dots$		n	$t_1 = t_2$	
				$\dots$	
	$\dots$		$n+1$	$A[t_1/x]$	
n	$A[t_2/x]$	$\longrightarrow$	$n+2$	$A[t_2/x]$	=e $n, (n+1)$

Forward tactics

conjunction introduction

`f_con_i H1 H2`

m_1	H1:	A		m_1	H1:	A	
m_2	H2:	B		m_2	H2:	B	
		$\dots$					
n		$A \wedge B$	$\longrightarrow$	n		$A \wedge B$	$\wedge$i m_1, m_2

conjunction elimination left

`f_con_e1 H`

m	H:	$A \wedge B$		m	H:	$A \wedge B$	
		$\dots$					
n		A	$\longrightarrow$	n		A	$\wedge$e$_1$ m

conjunction elimination right

`f_con_e2 H`

m	H:	$A \wedge B$		m	H:	$A \wedge B$	
		$\dots$					
n		B	$\longrightarrow$	n		B	$\wedge$e$_2$ m

disjunction introduction left

`f_dis_i1 H`

m	H:	A		m	H:	A	
		$\dots$					
n		$A \vee B$	$\longrightarrow$	n		$A \vee B$	$\vee$i$_1$ m

disjunction introduction right

`f_dis_i2 H`

m	H:	B		m	H:	B	
		$\dots$					
n		$A \vee B$	$\longrightarrow$	n		$A \vee B$	$\vee$i$_2$ m

implication elimination							
`f_imp_e H1 H2`							
m_1	`H1:`	$A \to B$		m_1	`H1:`	$A \to B$	
m_2	`H2:`	A		m_2	`H2:`	A	
		$\dots$					
n		B	$\longrightarrow$	n		B	$\to$e m_1, m_2
negation elimination							
`f_neg_e H1 H2`							
m_1	`H1:`	$\neg A$		m_1	`H1:`	$\neg A$	
m_2	`H2:`	A		m_2	`H2:`	A	
		$\dots$					
n		$\bot$	$\longrightarrow$	n		$\bot$	$\neg$e m_1, m_2
falsum elimination							
`f_fls_e H`							
m	`H:`	$\bot$		m	`H:`	$\bot$	
		$\dots$					
n		A	$\longrightarrow$	n		A	$\bot$e m
truth introduction							
`f_tru_i`							
		$\dots$					
n		$\top$	$\longrightarrow$	n		$\top$	$\top$i n
double negation introduction†							
`f_negneg_i H`							
m	`H:`	A		m	`H:`	A	
		$\dots$					
n		$\neg\neg A$	$\longrightarrow$	n		$\neg\neg A$	$\neg\neg$i m
*double negation elimination**							
`f_negneg_e H`							
m	`H:`	$\neg\neg A$		m	`H:`	$\neg\neg A$	
		$\dots$					
n		A	$\longrightarrow$	n		A	$\neg\neg$e m

*law of excluded middle**†							
`f_LEM`							
		$\dots$					
n		$A \vee \neg A$	$\longrightarrow$	n		$A \vee \neg A$	LEM
modus tollens†							
`f_MT H1 H2`							
m_1	`H1:`	$A \to B$		m_1	`H1:`	$A \to B$	
m_2	`H2:`	$\neg B$		m_2	`H2:`	$\neg B$	
		$\dots$					
n		$\neg A$	$\longrightarrow$	n		$\neg A$	MT m_1, m_2
universal elimination							
`f_all_e H`							
m	`H:`	$\forall x\, A$		m	`H:`	$\forall x\, A$	
		$\dots$					
n		$A[t/x]$	$\longrightarrow$	n		$A[t/x]$	$\forall x$ e m
existential introduction							
`f_exi_i H`							
m	`H:`	$A[t/x]$		m	`H:`	$A[t/x]$	
		$\dots$					
n		$\exists x\, A$	$\longrightarrow$	n		$\exists x\, A$	$\exists x$ i m
equality introduction							
`f_equ_i`							
		$\dots$					
n		$t = t$	$\longrightarrow$	n		$t = t$	=i
equality elimination							
`f_equ_e H1 H2`							
m_1	`H1:`	$t_1 = t_2$		m_1	`H1:`	$t_1 = t_2$	
m_2	`H2:`	$A[t_1/x]$		m_2	`H2:`	$A[t_1/x]$	
		$\dots$					
n		$A[t_2/x]$	$\longrightarrow$	n		$A[t_2/x]$	=e m_1, m_2