Formal Logic

Lasse Nielsen & Morten Ib Nielsen
Department of Computer Science, University of Copenhagen

July 14, 2006

1 Initial remarks

Initially we planned to show a simple result of Group Theory namely the uniqueness of the neutral element. Our idea was to develop propositional logic and predicate calculus first. Based on these we planned to develop the axiomatic set theory ZFC and finally when we had sets we could define groups. Unfortunately it turned out that this was much more cumbersome than we thought both because we are newcomers to Logiweb^{TM} and also because core Logiweb^{TM} is very low level. Being newcomers to Logiweb^{TM} we have used a lot of time trying to find out how to use the system. This hasn't been easy due to the absence of a hands on users manual. Thus we wasted a lot of time early on trying to parse other peoples code from earlier years in order to understand how to use pyk (the language used to construct proofs ect. in Logiweb^{TM}). This was a very frustrating and non-trivial task since this years pyk syntax is different from earlier years! A lot of emailing back and forth with Klaus Grue helped us, but progress was slow. Very late in the course we had the opportunity to sit down with Klaus in a kind of assisted programming session, where Klaus helped us with our problems as they occurred - this was very rewarding. After that we revised our goals with respect to this project and we found that even though we were now able to prove things in Logiweb^{TM} our initial goal was out of range because of the assembler like nature of our predicate calculus. Instead we decided to take the first step towards a more high level interface to our predicate calculus.

2 Conclusion

This Logiweb^{TM} page is an exam project on the course *202 Logik* Spring 2006 at the Department of Computer Science, University of Copenhagen. The purpose with this project is to use Logiweb^{TM} *to publish a machine checkable proof for a theorem of our choise* - which we have done [1]. Since both of us have backgrounds in mathematics and formal methods in computer science we started out having

[1] This page has been verified and found correct by Logiweb^{TM} and is available at http://www.diku.dk/hjemmesider/ansatte/grue/logiweb/20060417/home/mortenib/finalversion/fixed/.

high ambitions. As hinted we had to revise our goals along the way for several reasons. Instead of doing a large and complicated proof we have decided to show how to use the Logiweb™ system to *define a theory* (a set of axioms) and how to *define and prove lemmas in a theory* (both using axioms as well as already proved lemmas). Along the way we have explained key parts of the Logiweb™ notation. It is our hope that this Logiweb™ page will serve as a helping hand to future students on this course providing the *getting started* manual that we have been missing so much.

We start out defining a theory called *pred calc* containing one possible set of axioms and inference rules for the predicate calculus. Since these axioms are very low level we define, inspired by Natural Deduction (see [LiCS]), a set of higher level (and more intuitive) proof rules which we prove. Finally we use these higher level rules to prove lemma 6.6.

3 Introduction

This Logiweb™ page is formally correct. This means that it has been verified and found correct by the Logiweb™ proof engine. Therefore we can say that it is correct modulo errors in the proof engine. Is the content correct then? Well yes, but only in the sense that the lemmas we have proved are consequences of the axioms and proof rules we have introduced. There is no guarantee of soundness of our axioms and proof rules.

We have structured this document as follows. In section 4 we define the axioms and proof rules of predicate calculus. We conclude the section with two simple proofs namely the very (intuitively) obvious lemma *trivia* 4.1 and the lemma *repeat* 4.2. Then in section 5 we state and prove lemmas inspired by Natural Deduction. Finally in section 6 we use these lemmas to prove lemma 6.6. The appendix includes various Logiweb™ dependent stuff like *Tex definitions* and *Priority table* but also our suggestions of how Logiweb™ could be improved.

4 First order predicate calculus

Based on Mathworld[2] and thus on Kleene (2002) we define first-order predicate calculus below. We note that the axioms 1 through 10 together with the inference rule modus ponens (pcmp) constitutes the propositional calculus. Our definitions are not exactly like those found on Mathworld for two reasons. First we have made $\Rightarrow$ right associative in order to get rid of unnecessary parenthesis. This means that $\mathcal{F} \Rightarrow \mathcal{G} \Rightarrow \mathcal{F}$ really means $\mathcal{F} \Rightarrow (\mathcal{G} \Rightarrow \mathcal{F})$ below. Second we had to express formulations such as *in which* x *occurs free* in a machine checkable way.

The [**Theory** pred calc] contains the following axioms:

[2] http://mathworld.wolfram.com/First-OrderLogic.html.

1. [pred calc **rule** pc1: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \Rightarrow \mathcal{G} \Rightarrow \mathcal{F}$]
2. [pred calc **rule** pc2: $\Pi\mathcal{F},\mathcal{G},\mathcal{H}\colon (\mathcal{F} \Rightarrow \mathcal{G}) \Rightarrow (\mathcal{F} \Rightarrow \mathcal{G} \Rightarrow \mathcal{H}) \Rightarrow \mathcal{F} \Rightarrow \mathcal{H}$]
3. [pred calc **rule** pc3: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \Rightarrow \mathcal{G} \Rightarrow \mathcal{F} \wedge \mathcal{G}$]
4. [pred calc **rule** pc4: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \Rightarrow \mathcal{F} \vee \mathcal{G}$]
5. [pred calc **rule** pc5: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \Rightarrow \mathcal{G} \vee \mathcal{F}$]
6. [pred calc **rule** pc6: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \wedge \mathcal{G} \Rightarrow \mathcal{F}$]
7. [pred calc **rule** pc7: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \wedge \mathcal{G} \Rightarrow \mathcal{G}$]
8. [pred calc **rule** pc8: $\Pi\mathcal{F},\mathcal{G},\mathcal{H}\colon (\mathcal{F} \Rightarrow \mathcal{G}) \Rightarrow (\mathcal{H} \Rightarrow \mathcal{G}) \Rightarrow \mathcal{F} \vee \mathcal{H} \Rightarrow \mathcal{G}$]
9. [pred calc **rule** pc9: $\Pi\mathcal{F},\mathcal{G}\colon (\mathcal{F} \Rightarrow \mathcal{G}) \Rightarrow (\mathcal{F} \Rightarrow \neg\mathcal{G}) \Rightarrow \neg\mathcal{F}$]
10. [pred calc **rule** pc10: $\Pi\mathcal{F}\colon \neg\neg\mathcal{F} \Rightarrow \mathcal{F}$]
11. [pred calc **rule** pc11: $\Pi\mathcal{X},\mathcal{R},\mathcal{G},\mathcal{F}\colon \mathcal{X}\#\mathcal{R} \Vdash \mathcal{X}\#\mathcal{G} \Vdash \langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle \Vdash \forall\mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{G}$]
12. [pred calc **rule** pc12: $\Pi\mathcal{X},\mathcal{R},\mathcal{G},\mathcal{F}\colon \mathcal{X}\#\mathcal{R} \Vdash \mathcal{X}\#\mathcal{G} \Vdash \langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle \Vdash \mathcal{G} \Rightarrow \exists\mathcal{X}.\,(\mathcal{F})$]

The proof rules in the [**Theory** pred calc] are:

- [pred calc **rule** pcmp: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \vdash \mathcal{F} \Rightarrow \mathcal{G} \vdash \mathcal{G}$]
- [pred calc **rule** pcia: $\Pi\mathcal{F},\mathcal{G},\mathcal{X}\colon \mathcal{X}\#\mathcal{G} \Vdash \mathcal{G} \Rightarrow \mathcal{F} \vdash \mathcal{G} \Rightarrow \forall\mathcal{X}.\,(\mathcal{F})$]
- [pred calc **rule** pcie: $\Pi\mathcal{F},\mathcal{G},\mathcal{X}\colon \mathcal{X}\#\mathcal{G} \Vdash \mathcal{F} \Rightarrow \mathcal{G} \vdash \exists\mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{G}$]
- [pred calc **rule** pcdeduction: $\Pi\mathcal{A},\mathcal{B}\colon \mathrm{Ded}(\mathcal{A},\mathcal{B}) \Vdash \mathcal{A} \vdash \mathcal{B}$]

4.1 A conservative extension

The rule *pcdeduction* is not really a part of Predicate Calculus according to Mathworld but it is a conservative extension in the sense than we cannot prove anything using this rule that we cannot prove without it. We include it in order to make proofs shorter and thus easier to understand for humans. As a curiosity we cannot prove that *pcdeduction* is a conservative extension formally [3].

[3] In a machine checkable way.

4.2 A troublesome extension

It seems fair to conclude $\mathcal{B}$ if both $\mathcal{A} \vdash \mathcal{B}$ and $\mathcal{A}$ are known. This is the proof rule *modus ponens*. Unfortunately Logiweb$^{\text{TM}}$ does not allow the use of modus ponens on anything other than lemmas e.g. modus ponens cannot be used on line numbers in proofs. One of the aims of this project is to introduce and prove natural deduction like proof rules enabling the use of pred calc on a higher level. Since we cannot use line numbers with modus ponens we cannot prove the implication introduction rule (see [LiCS, p. 27]). Therefore we need to add the following rule as a proof rule in [**Theory** pred calc].

- [pred calc **rule** pcunsound: $\Pi\mathcal{F},\mathcal{G}\colon (\mathcal{F} \vdash \mathcal{G}) \vdash \mathcal{F} \vdash \mathcal{G}$]

We have been told that this rule makes the system unsound, but we have not seen and cannot find an example of this. We have been told (by Klaus Grue via email) that it requires some special side conditions in order to be sound. Alas we have been unable to communicate with Klaus Grue in the week up to our deadline. Therefore we have been unable to ameliorate this deficiency but we are convinced that we only use the rule in sound ways. Given the right side conditions we expect that our proofs could be patched simply by adding these side conditions.

4.3 Notation

Some of the notation used above is most likely unfamiliar. So let's spend a few paragraphs explaining it before we continue. The meaning of the axioms and proof rules above are of course given by their definition but in order to understand the definitions we need to explain the basic syntax (or at least the part of it we use) in Logiweb$^{\text{TM}}$ definitions and proofs.

- **The symbol** Π is the for all meta quantifier.

- **The symbol** $\vdash$ is placed between two propositions. The meaning of $\mathrm{A} \vdash \mathrm{B}$ is that if A can be proved then so can B. $\vdash$ can be used sequentially in the way that $\mathrm{A} \vdash \mathrm{B} \vdash \mathrm{C}$ means if A can be proved, then if B can be proved, then so can C. That is if A and B can be proved then so can C.

- **The symbol** $\rhd$ is placed between a rule and a line number. It is the opposite of $\vdash$ in the sense that if R is a rule saying that $\mathrm{A} \vdash \mathrm{B}$ and L is a line concluding A, then $\mathrm{R} \rhd \mathrm{L}$ concludes B.

- **Explanation of** $\gg$. A line in a proof consists of the symbol $\gg$ with the use of a rule on the left, and the conclusion on the right. The example above would give the line $\mathrm{R} \rhd \mathrm{L} \gg \mathrm{B}$.

- **The meaning of** $\Vdash$ is almost the same as $\vdash$. If the condition on the left evaluates to true, then the proposition on the right can be proved. The condition on the left of a $\Vdash$ is a so called side-condition. The only side-conditions we use are expressed using substitution and $\sharp$ (explained below).

- **The** $\rhd\!\!\rhd$ symbol is used much like $\rhd$, but it is the opposite of $\Vdash$ instead of $\vdash$, thus the line number on the right of the symbol must conclude that the side-condition is fulfilled.
- **The side-condition** $\langle \mathrm{A} \equiv \mathrm{B} \mid \mathrm{C} := \mathrm{D} \rangle$ is fulfilled, if the proposition B where any occurrence of the meta-variable C is replaced by the proposition D is exactly equal to the proposition A. Another way to express this using a more common notation of substitution would be $\mathrm{A} \equiv \mathrm{B}[\mathrm{D}/\mathrm{C}]$. It is important to note that this substitution avoids variable capture. An example of why variable capture would be unsound is the propositional formula $\forall \mathrm{x} \exists \mathrm{y} : (\mathrm{x} \vee \mathrm{y}) \wedge (\neg \mathrm{x} \vee \neg \mathrm{y})$. We see that this is a tautology since the choice of $\mathrm{y} \equiv \neg \mathrm{x}$ satisfies the formula. But if we substituted y for x in this formula then we would get an absurdity.
- **The side-condition** $\mathrm{A} \sharp \mathrm{B}$ is fulfilled if the meta-variable A does not occur (free) in the proposition B.

4.4 Some small proofs

Having explained the syntax of definitions and proofs we continue with two simple but useful lemmas just to show how it is done. Lemma 4.1 is proved using only axioms and proof rules while the proof of lemma 4.2 uses the result of lemma 4.1 as well.

Lemma 4.1 [*pred calc* **lemma** *trivia*: $\forall \mathcal{F} : \mathcal{F} \Rightarrow \mathcal{F}$]

pred calc **proof of** trivia:

L01:	Arbitrary $\gg$	$\mathcal{F}$	;
L02:	pc2 $\gg$	$(\mathcal{F} \Rightarrow \mathcal{F} \Rightarrow \mathcal{F}) \Rightarrow (\mathcal{F} \Rightarrow (\mathcal{F} \Rightarrow \mathcal{F}) \Rightarrow \mathcal{F}) \Rightarrow \mathcal{F} \Rightarrow \mathcal{F}$	;
L03:	pc1 $\gg$	$\mathcal{F} \Rightarrow \mathcal{F} \Rightarrow \mathcal{F}$	;
L04:	pcmp $\rhd$ L03 $\rhd$ L02 $\gg$	$(\mathcal{F} \Rightarrow (\mathcal{F} \Rightarrow \mathcal{F}) \Rightarrow \mathcal{F}) \Rightarrow \mathcal{F} \Rightarrow \mathcal{F}$	;
L05:	pc1 $\gg$	$\mathcal{F} \Rightarrow (\mathcal{F} \Rightarrow \mathcal{F}) \Rightarrow \mathcal{F}$	;
L06:	pcmp $\rhd$ L05 $\rhd$ L04 $\gg$	$\mathcal{F} \Rightarrow \mathcal{F}$	$\square$

Lemma 4.2 (Repetition) [*pred calc* **lemma** *repeat*: $\Pi \mathcal{F} : \mathcal{F} \vdash \mathcal{F}$]

pred calc **proof of** repeat:

L01:	Arbitrary $\gg$	$\mathcal{F}$	;
L02:	Premise $\gg$	$\mathcal{F}$	;
L03:	trivia $\gg$	$\mathcal{F} \Rightarrow \mathcal{F}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{F}$	$\square$

5 Natural deduction

The axioms and proof rules in [**Theory** pred calc] constitutes a very low level proof system. In order to ameliorate this we introduce and prove some higher level (and more intuitive) proof rules. These proof rules are inspired by natural deduction as defined in [LiCS]. We conclude this section by justifying why we have replaced two rules from [LiCS] with a new rule.

Lemma 5.1 [*pred calc* **lemma** *andintro*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \vdash \mathcal{G} \vdash \mathcal{F} \wedge \mathcal{G}$]

pred calc **proof of** andintro:

L01:	Arbitrary $\gg$	$\mathcal{F},\mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F}$	;
L03:	Premise $\gg$	$\mathcal{G}$	;
L04:	pc3 $\gg$	$\mathcal{F} \Rightarrow \mathcal{G} \Rightarrow \mathcal{F} \wedge \mathcal{G}$	;
L05:	pcmp $\rhd$ L02 $\rhd$ L04 $\gg$	$\mathcal{G} \Rightarrow \mathcal{F} \wedge \mathcal{G}$	;
L06:	pcmp $\rhd$ L03 $\rhd$ L05 $\gg$	$\mathcal{F} \wedge \mathcal{G}$	$\square$

Lemma 5.2 [*pred calc* **lemma** *andelim1*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \wedge \mathcal{G} \vdash \mathcal{F}$]

pred calc **proof of** andelim1:

L01:	Arbitrary $\gg$	$\mathcal{F},\mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \wedge \mathcal{G}$	;
L03:	pc6 $\gg$	$\mathcal{F} \wedge \mathcal{G} \Rightarrow \mathcal{F}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{F}$	$\square$

Lemma 5.3 [*pred calc* **lemma** *andelim2*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \wedge \mathcal{G} \vdash \mathcal{G}$]

pred calc **proof of** andelim2:

L01:	Arbitrary $\gg$	$\mathcal{F},\mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \wedge \mathcal{G}$	;
L03:	pc7 $\gg$	$\mathcal{F} \wedge \mathcal{G} \Rightarrow \mathcal{G}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{G}$	$\square$

Lemma 5.4 [*pred calc* **lemma** *orintro1*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \vdash \mathcal{F} \vee \mathcal{G}$]

pred calc **proof of** orintro1:

L01:	Arbitrary $\gg$	$\mathcal{F},\mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F}$	;
L03:	pc4 $\gg$	$\mathcal{F} \Rightarrow \mathcal{F} \vee \mathcal{G}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{F} \vee \mathcal{G}$	$\square$

Lemma 5.5 [*pred calc* **lemma** *orintro2*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{G} \vdash \mathcal{F} \vee \mathcal{G}$]

pred calc **proof of** orintro2:

L01:	Arbitrary $\gg$	$\mathcal{F},\mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{G}$	;
L03:	pc5 $\gg$	$\mathcal{G} \Rightarrow \mathcal{F} \vee \mathcal{G}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{F} \vee \mathcal{G}$	$\square$

Lemma 5.6 [*pred calc* **lemma** *orelim*: $\Pi\mathcal{F},\mathcal{G},\mathcal{H}\colon \mathcal{F} \vee \mathcal{G} \vdash (\mathcal{F} \vdash \mathcal{H}) \vdash (\mathcal{G} \vdash \mathcal{H}) \vdash \mathcal{H}$]

pred calc **proof of** orelim:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}, \mathcal{H}$	;
L02:	Premise $\gg$	$\mathcal{F} \vee \mathcal{G}$	;
L03:	Premise $\gg$	$\mathcal{F} \vdash \mathcal{H}$	;
L04:	Premise $\gg$	$\mathcal{G} \vdash \mathcal{H}$	;
L05:	implyintro $\rhd$ L03 $\gg$	$\mathcal{F} \Rightarrow \mathcal{H}$	;
L06:	implyintro $\rhd$ L04 $\gg$	$\mathcal{G} \Rightarrow \mathcal{H}$	;
L07:	pc8 $\gg$	$(\mathcal{F} \Rightarrow \mathcal{H}) \Rightarrow (\mathcal{G} \Rightarrow \mathcal{H}) \Rightarrow \mathcal{F} \vee \mathcal{G} \Rightarrow \mathcal{H}$	;
L08:	pcmp $\rhd$ L05 $\rhd$ L07 $\gg$	$(\mathcal{G} \Rightarrow \mathcal{H}) \Rightarrow \mathcal{F} \vee \mathcal{G} \Rightarrow \mathcal{H}$	;
L09:	pcmp $\rhd$ L06 $\rhd$ L08 $\gg$	$\mathcal{F} \vee \mathcal{G} \Rightarrow \mathcal{H}$	;
L10:	pcmp $\rhd$ L02 $\rhd$ L09 $\gg$	$\mathcal{H}$	$\square$

Lemma 5.7 [*pred calc* **lemma** *implyintro*: $\Pi \mathcal{F}, \mathcal{G} \colon (\mathcal{F} \vdash \mathcal{G}) \vdash \mathcal{F} \Rightarrow \mathcal{G}$]

pred calc **proof of** implyintro:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \vdash \mathcal{G}$	;
L03:	Block $\gg$	Begin	;
L04:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L05:	Premise $\gg$	$\mathcal{F}$	;
L06:	pcunsound $\rhd$ L02 $\rhd$ L05 $\gg$	$\mathcal{G}$	;
L07:	Block $\gg$	End	;
L08:	pcdeduction $\rhd$ L07 $\gg$	$\mathcal{F} \Rightarrow \mathcal{G}$	$\square$

Lemma 5.8 [*pred calc* **lemma** *notintro*: $\Pi \mathcal{F}, \mathcal{G} \colon (\mathcal{F} \vdash \mathcal{G}) \vdash (\mathcal{F} \vdash \neg \mathcal{G}) \vdash \neg \mathcal{F}$]

pred calc **proof of** notintro:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \vdash \mathcal{G}$	;
L03:	Premise $\gg$	$\mathcal{F} \vdash \neg \mathcal{G}$	;
L04:	implyintro $\rhd$ L02 $\gg$	$\mathcal{F} \Rightarrow \mathcal{G}$	;
L05:	implyintro $\rhd$ L03 $\gg$	$\mathcal{F} \Rightarrow \neg \mathcal{G}$	;
L06:	pc9 $\gg$	$(\mathcal{F} \Rightarrow \mathcal{G}) \Rightarrow (\mathcal{F} \Rightarrow \neg \mathcal{G}) \Rightarrow \neg \mathcal{F}$	;
L07:	pcmp $\rhd$ L04 $\rhd$ L06 $\gg$	$(\mathcal{F} \Rightarrow \neg \mathcal{G}) \Rightarrow \neg \mathcal{F}$	;
L08:	pcmp $\rhd$ L05 $\rhd$ L07 $\gg$	$\neg \mathcal{F}$	$\square$

Lemma 5.9 [*pred calc* **lemma** *notnotelim*: $\Pi \mathcal{F} \colon \neg\neg \mathcal{F} \vdash \mathcal{F}$]

pred calc **proof of** notnotelim:

L01:	Arbitrary $\gg$	$\mathcal{F}$	;
L02:	Premise $\gg$	$\neg\neg \mathcal{F}$	;
L03:	pc10 $\gg$	$\neg\neg \mathcal{F} \Rightarrow \mathcal{F}$	;
L04:	pcmp $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{F}$	$\square$

Lemma 5.10 [*pred calc* **lemma** *forallintro*: $\Pi \mathcal{F}, \mathcal{G}, \mathcal{X} \colon \mathcal{X} \# \mathcal{G} \vee \neg \mathcal{G} \Vdash \mathcal{F} \vdash \forall \mathcal{X}. (\mathcal{F})$]

pred calc **proof of** forallintro:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}, \mathcal{X}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G} \vee \neg\mathcal{G}$	;
L03:	Premise $\gg$	$\mathcal{F}$	;
L04:	lem $\gg$	$\mathcal{G} \vee \neg\mathcal{G}$	;
L05:	Block $\gg$	Begin	;
L06:	Arbitrary $\gg$	$\mathcal{G}, \mathcal{F}$	;
L07:	Premise $\gg$	$\mathcal{G} \vee \neg\mathcal{G}$	;
L08:	repeat $\rhd$ L03 $\gg$	$\mathcal{F}$	;
L09:	Block $\gg$	End	;
L10:	pcdeduction $\rhd$ L09 $\gg$	$\mathcal{G} \vee \neg\mathcal{G} \Rightarrow \mathcal{F}$	;
L11:	pcia $\unrhd$ L02 $\rhd$ L10 $\gg$	$\mathcal{G} \vee \neg\mathcal{G} \Rightarrow \forall\mathcal{X}.\,(\mathcal{F})$	;
L12:	pcmp $\rhd$ L04 $\rhd$ L11 $\gg$	$\forall\mathcal{X}.\,(\mathcal{F})$	$\Box$

Remark 5.11 In lemma 5.10 we use the side condition $\mathcal{X}\#\mathcal{G} \vee \neg\mathcal{G}$. We would like to use the equivalent $\mathcal{X}\#\mathcal{G}$ instead but we haven't been able to find a way to prove the former side condition from the latter. Another example of reasoning about side conditions is the following. If we have $\langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle$ and $\mathcal{X}\#\mathcal{R}$ then it follows that $\mathcal{X}\#\mathcal{G}$. Unfortunately we haven't heard of any ways to reason about side conditions therefore we are forced to assume side conditions in lemmas whenever we need them even though they might be deductable.

Lemma 5.12
[*pred calc* **lemma** *forallelim*: $\Pi\mathcal{X}, \mathcal{R}, \mathcal{G}, \mathcal{F}\colon \mathcal{X}\#\mathcal{R} \Vdash \mathcal{X}\#\mathcal{G} \Vdash \langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle \Vdash \forall\mathcal{X}.\,(\mathcal{F}) \vdash \mathcal{G}$]

pred calc **proof of** forallelim:

L01:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{R}, \mathcal{G}, \mathcal{F}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{R}$	;
L03:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G}$	;
L04:	Side-condition $\gg$	$\langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle$	;
L05:	pc11 $\unrhd$ L02 $\unrhd$ L03 $\unrhd$ L04 $\gg$	$\forall\mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{G}$	;
L06:	Block $\gg$	Begin	;
L07:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{G}, \mathcal{F}$	;
L08:	Premise $\gg$	$\forall\mathcal{X}.\,(\mathcal{F})$	;
L09:	pcmp $\rhd$ L08 $\rhd$ L05 $\gg$	$\mathcal{G}$	;
L10:	Block $\gg$	End	;
L11:	pcdeduction $\rhd$ L10 $\gg$	$\forall\mathcal{X}.\,(\mathcal{F}) \vdash \mathcal{G}$	$\Box$

Lemma 5.13
[*pred calc* **lemma** *existsintro*: $\Pi\mathcal{X}, \mathcal{R}, \mathcal{G}, \mathcal{F}\colon \mathcal{X}\#\mathcal{R} \Vdash \mathcal{X}\#\mathcal{G} \Vdash \langle\mathcal{G}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{R}\rangle \Vdash \mathcal{G} \vdash \exists\mathcal{X}.\,(\mathcal{F})$]

pred calc **proof of** existsintro:

L01:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{R}, \mathcal{G}, \mathcal{F}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{R}$	;

L03:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G}$	;
L04:	Side-condition $\gg$	$\langle\mathcal{G}{\equiv}\mathcal{F}\vert\mathcal{X}{:=}\mathcal{R}\rangle$	;
L05:	pc12 $\unrhd$ L02 $\unrhd$ L03 $\unrhd$ L04 $\gg$	$\mathcal{G} \Rightarrow \exists\mathcal{X}.\,(\mathcal{F})$	;
L06:	Block $\gg$	Begin	;
L07:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{G}, \mathcal{F}$	;
L08:	Premise $\gg$	$\mathcal{G}$	;
L09:	pcmp $\rhd$ L08 $\rhd$ L05 $\gg$	$\exists\mathcal{X}.\,(\mathcal{F})$	;
L10:	Block $\gg$	End	;
L11:	pcdeduction $\rhd$ L10 $\gg$	$\mathcal{G} \vdash \exists\mathcal{X}.\,(\mathcal{F})$	$\Box$

Lemma 5.14 [*pred calc* **lemma** *existselim*: $\Pi\mathcal{X},\mathcal{F},\mathcal{G}\colon \mathcal{X}\#\mathcal{G} \Vdash \exists\mathcal{X}.\,(\mathcal{F}) \vdash (\mathcal{F} \vdash \mathcal{G}) \vdash \mathcal{G}$]

pred calc **proof of** existselim:

L01:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{F}, \mathcal{G}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G}$	;
L03:	Premise $\gg$	$\exists\mathcal{X}.\,(\mathcal{F})$	;
L04:	Premise $\gg$	$\mathcal{F} \vdash \mathcal{G}$	;
L05:	implyintro $\rhd$ L04 $\gg$	$\mathcal{F} \Rightarrow \mathcal{G}$	;
L06:	pcie $\unrhd$ L02 $\rhd$ L05 $\gg$	$\exists\mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{G}$	;
L07:	pcmp $\rhd$ L03 $\rhd$ L06 $\gg$	$\mathcal{G}$	$\Box$

5.1 Derived lemmas

Below we apply the lemmas above together with the defining rules of [**Theory** pred calc] to prove some other fairly standard and very useful lemmas.

Lemma 5.15 (Modus Tollens) [*pred calc* **lemma** *mt*: $\Pi\mathcal{F},\mathcal{G}\colon \mathcal{F} \Rightarrow \mathcal{G} \vdash \neg\mathcal{G} \vdash \neg\mathcal{F}$]

pred calc **proof of** mt:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \Rightarrow \mathcal{G}$	;
L03:	Premise $\gg$	$\neg\mathcal{G}$	;
L04:	Block $\gg$	Begin	;
L05:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L06:	Premise $\gg$	$\mathcal{F}$	;
L07:	pcmp $\rhd$ L06 $\rhd$ L02 $\gg$	$\mathcal{G}$	;
L08:	Block $\gg$	End	;
L09:	pcdeduction $\rhd$ L08 $\gg$	$\mathcal{F} \vdash \mathcal{G}$	;
L10:	Block $\gg$	Begin	;
L11:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L12:	Premise $\gg$	$\mathcal{F}$	;
L13:	repeat $\rhd$ L03 $\gg$	$\neg\mathcal{G}$	;
L14:	Block $\gg$	End	;
L15:	pcdeduction $\rhd$ L14 $\gg$	$\mathcal{F} \vdash \neg\mathcal{G}$	;
L16:	notintro $\rhd$ L09 $\rhd$ L15 $\gg$	$\neg\mathcal{F}$	$\Box$

Lemma 5.16 [*pred calc* **lemma** *notnotintro*: $\Pi\mathcal{F}\colon \mathcal{F} \vdash \neg\neg\mathcal{F}$]

pred calc **proof of** notnotintro:

L01:	Arbitrary ≫	$\mathcal{F}$	;
L02:	Premise ≫	$\mathcal{F}$	;
L03:	Block ≫	Begin	;
L04:	Arbitrary ≫	$\quad\mathcal{F}$	;
L05:	Premise ≫	$\quad\mathcal{F}$	;
L06:	Premise ≫	$\quad\neg\mathcal{F}$	;
L07:	repeat ▷ L05 ≫	$\quad\mathcal{F}$	;
L06:	Block ≫	End	;
L08:	pcdeduction ▷ L06 ≫	$\mathcal{F} \Rightarrow \neg\mathcal{F} \Rightarrow \mathcal{F}$	;
L09:	pcmp ▷ L02 ▷ L08 ≫	$\neg\mathcal{F} \Rightarrow \mathcal{F}$	;
L10:	trivia ≫	$\neg\mathcal{F} \Rightarrow \neg\mathcal{F}$	;
L11:	pc9 ≫	$(\neg\mathcal{F} \Rightarrow \mathcal{F}) \Rightarrow (\neg\mathcal{F} \Rightarrow \neg\mathcal{F}) \Rightarrow \neg\neg\mathcal{F}$	;
L12:	pcmp ▷ L09 ▷ L11 ≫	$(\neg\mathcal{F} \Rightarrow \neg\mathcal{F}) \Rightarrow \neg\neg\mathcal{F}$	;
L13:	pcmp ▷ L10 ▷ L12 ≫	$\neg\neg\mathcal{F}$	□

Lemma 5.17 (Proof by Contradiction) [*pred calc* **lemma** *pbc*: $\Pi\mathcal{F},\mathcal{G}\colon (\neg\mathcal{F} \vdash \mathcal{G}) \vdash (\neg\mathcal{F} \vdash \neg\mathcal{G}) \vdash \mathcal{F}$]

pred calc **proof of** pbc:

L01:	Arbitrary ≫	$\mathcal{F},\mathcal{G}$	;
L02:	Premise ≫	$\neg\mathcal{F} \vdash \mathcal{G}$	;
L03:	Premise ≫	$\neg\mathcal{F} \vdash \neg\mathcal{G}$	;
L04:	notintro ▷ L02 ▷ L03 ≫	$\neg\neg\mathcal{F}$	;
L05:	notnotelim ▷ L04 ≫	$\mathcal{F}$	□

Lemma 5.18 (Law of the Excluded Middle)
[*pred calc* **lemma** *lem*: $\Pi\mathcal{F}\colon \mathcal{F} \vee \neg\mathcal{F}$]

pred calc **proof of** lem:

L01:	Arbitrary ≫	$\mathcal{F}$	;
L02:	Block ≫	Begin	;
L03:	Arbitrary ≫	$\quad\mathcal{F}$	;
L04:	Premise ≫	$\quad\neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L05:	Block ≫	Begin	;
L06:	Arbitrary ≫	$\quad\mathcal{F}$	;
L07:	Premise ≫	$\quad\mathcal{F}$	;
L08:	orintro1 ▷ L07 ≫	$\quad\mathcal{F} \vee \neg\mathcal{F}$	;
L09:	Block ≫	End	;
L10:	pcdeduction ▷ L09 ≫	$\quad\mathcal{F} \vdash \mathcal{F} \vee \neg\mathcal{F}$	;
L11:	Block ≫	Begin	;
L12:	Arbitrary ≫	$\quad\mathcal{F}$	;
L13:	Premise ≫	$\quad\mathcal{F}$	;
L14:	repeat ▷ L04 ≫	$\quad\neg(\mathcal{F} \vee \neg\mathcal{F})$	;

L15:	Block ≫	End	;
L16:	pcdeduction ▷ L15 ≫	$\mathcal{F} \vdash \neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L17:	notintro ▷ L10 ▷ L16 ≫	$\neg\mathcal{F}$	;
L18:	orintro2 ▷ L17 ≫	$\mathcal{F} \vee \neg\mathcal{F}$	;
L19:	Block ≫	End	;
L20:	pcdeduction ▷ L19 ≫	$\neg(\mathcal{F} \vee \neg\mathcal{F}) \vdash \mathcal{F} \vee \neg\mathcal{F}$	;
L21:	Block ≫	Begin	;
L22:	Arbitrary ≫	$\mathcal{F}$	;
L23:	Premise ≫	$\neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L24:	repeat ▷ L23 ≫	$\neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L25:	Block ≫	End	;
L26:	pcdeduction ▷ L25 ≫	$\neg(\mathcal{F} \vee \neg\mathcal{F}) \vdash \neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L27:	notintro ▷ L20 ▷ L26 ≫	$\neg\neg(\mathcal{F} \vee \neg\mathcal{F})$	;
L28:	notnotelim ▷ L27 ≫	$\mathcal{F} \vee \neg\mathcal{F}$	□

5.2 A word on ⊥

The proof rules of natural deduction in [LiCS] uses bottom. Bottom represents the concept of unsoundness, that is it should be impossible to prove bottom in a sound logic. The way to prove bottom would be to prove any absurdity, that is for any proposition A to prove both A and not A. In [LiCS] this is captured in the proof rule $\frac{\neg A \quad A}{\bot}$. In [LiCS] bottom is used in two ways. First if you under the assumption of a proposition A can prove bottom then you can conclude that A is false, that is not A is true. In [LiCS] this is captured by the proof rule $\frac{\begin{matrix} A \\ \vdots \\ \bot \end{matrix}}{\neg A}$ (notelim). This makes sense if we assume that the logical system is sound, because this means that it is free of absurdities, so if A was true it would be impossible to prove an absurdity thus A must be false. Second the assumption of bottom can be used to conclude anything. In [LiCS] this is captured in the proof rule $\frac{\bot}{A}$ (botelim).

Since the predicate logic from Mathworld, which we have used as a basis for [**Theory** pred calc], doesn't use or define the notion of bottom, we cannot adopt the rules of natural deduction directly. We have chosen to solve this problem by replacing the problematic proof rules above with a new proof rule called *notintro*. This way we can avoid the use of bottom alltogether while we preserve the rest of the system.

To justify our actions we hand proof the following metatheorem:

Theorem 5.19 *Let Nat' be the system of proof rules introduced in section 5 and let Nat be the same system without the rule notintro but with rules notelim and botelim added. Let* F *be fixed and define* $\bot \equiv \mathrm{F} \wedge \neg\mathrm{F}$[4]*. Then the following holds:*

1. *If* B *can be proved in Nat then* $\mathrm{B}[\mathrm{F} \wedge \neg\mathrm{F}/\bot]$ *can be proved in Nat'.*

2. *If* B *can be proved in Nat' then B can be proved in Nat.*

PROOF:

Both claims in this metatheorem are proved by induction on the derivation of the proof on the left hand side of the implication. To save space we only consider the interesting cases. Thus we skip all of the rules the two systems have in common.

Proof of 1:
In this proof $A' \equiv A[F \wedge \neg F/\bot]$.

The rule $\frac{A \quad \neg A}{\bot}$.
Given proofs of A and $\neg A$ we need to prove $F \wedge \neg F$. Using the induction hypothesis on the proofs of A and $\neg A$, we get proofs of A' and $\neg A'$ using our system of lemmas. Now we have proofs of A' and $\neg A'$, which means that we can also prove A' and $\neg A'$ using F or $\neg F$ as assumptions. Now we can construct the proof of $F \wedge \neg F$ like this:

$$\frac{\dfrac{\dfrac{\begin{matrix}\neg F\\ \vdots\\ A'\end{matrix} \qquad \begin{matrix}\neg F\\ \vdots\\ \neg A'\end{matrix}}{\neg\neg F}}{F} \qquad \dfrac{\begin{matrix}F\\ \vdots\\ A'\end{matrix} \qquad \begin{matrix}F\\ \vdots\\ \neg A'\end{matrix}}{\neg F}}{F \wedge \neg F}$$

The rule $\frac{\bot}{A}$.
Given proof of $F \wedge \neg F$ we need to prove A'. Using the induction hypothesis we get a proof of $F \wedge \neg F$. Using andelim1 and andelim2 we obtain proofs of F and $\neg F$. Finally (using weakening) we can construct proofs of F and $\neg F$ under the assumption of $\neg A'$. Now we can construct the proof of A' as follows:

$$\frac{\dfrac{\dfrac{\begin{matrix}\neg A'\\ \vdots\\ F \wedge \neg F\end{matrix}}{F} \qquad \dfrac{\begin{matrix}\neg A'\\ \vdots\\ F \wedge \neg F\end{matrix}}{\neg F}}{\neg\neg A'}}{A'}$$

The rule $\frac{A \vdash \bot}{\neg A}$. Using the induction hypothesis on the proof of $A \vdash \bot$ we obtain a proof of $A' \vdash F \wedge \neg F$. Using the andelim1 and andelim2 lemmas, we get proofs of $A' \vdash F$ and $A' \vdash \neg F$. Now we can construct the proof of $\neg A'$ like this:

$$\frac{\dfrac{\begin{matrix}A'\\ \vdots\\ F \wedge \neg F\end{matrix}}{F} \qquad \dfrac{\begin{matrix}A'\\ \vdots\\ F \wedge \neg F\end{matrix}}{\neg F}}{\neg A'}$$

[4]F must be fixed for all occurrences of bottom e.g. $\bot \vee \bot$ must be translated to $(F \wedge \neg F) \vee (F \wedge \neg F)$ and can't be translated to $(F \wedge \neg F) \vee (G \wedge \neg G)$.

That concludes all the interesting rules. The other rules follow by using the induction hypothesis on the given proofs, and using the same rule to conclude the desired proposition.

Proof of 2:
There is only one interesting rule, and that is notintro. Given the proofs of A ⊢ B and A ⊢ ¬B we wish to prove ¬A. We construct the proof like this.

$$\dfrac{\dfrac{\begin{matrix} A & A \\ \vdots & \vdots \\ B & \neg B \end{matrix}}{\bot}}{\neg A}$$

That concludes all the interesting rules. The other rules follow by using the induction hypothesis on the given proofs, and using the same rule to conclude the desired proposition.

■

5.2.1 A practical lemma

Above we saw that ⊥ could be replaced with F ∧ ¬F and that the proof rule botelim allows us to conclude anything once we have ⊥. This gives rise to the following very usefull lemmas.

Lemma 5.20 [*pred calc* **lemma** *bottomelim*: $\Pi\mathcal{F},\mathcal{G}: \mathcal{F} \wedge \neg\mathcal{F} \vdash \mathcal{G}$]

pred calc **proof of** bottomelim:

L01:	Arbitrary ≫	$\mathcal{F},\mathcal{G}$	;
L02:	Premise ≫	$\mathcal{F} \wedge \neg\mathcal{F}$	;
L03:	Block ≫	Begin	;
L04:	Arbitrary ≫	$\mathcal{F},\mathcal{G}$	;
L05:	Premise ≫	$\neg\mathcal{G}$	;
L06:	andelim1 ▷ L02 ≫	$\mathcal{F}$	;
L07:	Block ≫	End	;
L08:	pcdeduction ▷ L07 ≫	$\neg\mathcal{G} \vdash \mathcal{F}$	;
L09:	Block ≫	Begin	;
L10:	Arbitrary ≫	$\mathcal{F},\mathcal{G}$	;
L11:	Premise ≫	$\neg\mathcal{G}$	;
L12:	andelim2 ▷ L02 ≫	$\neg\mathcal{F}$	;
L13:	Block ≫	End	;
L14:	pcdeduction ▷ L13 ≫	$\neg\mathcal{G} \vdash \neg\mathcal{F}$	;
L15:	notintro ▷ L08 ▷ L14 ≫	$\neg\neg\mathcal{G}$	;
L16:	notnotelim ▷ L15 ≫	$\mathcal{G}$	□

Lemma 5.21 [*pred calc* **lemma** *lemnotintro*: $\Pi\mathcal{F},\mathcal{G}: (\mathcal{F} \Rightarrow \mathcal{G} \wedge \neg\mathcal{G}) \vdash \neg\mathcal{F}$]

pred calc **proof of** lemnotintro:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L02:	Premise $\gg$	$\mathcal{F} \Rightarrow \mathcal{G} \wedge \neg\mathcal{G}$	;
L03:	Block $\gg$	Begin	;
L04:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L05:	Premise $\gg$	$\mathcal{F}$	;
L06:	pcmp $\rhd$ L05 $\rhd$ L02 $\gg$	$\mathcal{G} \wedge \neg\mathcal{G}$	;
L07:	andelim1 $\rhd$ L06 $\gg$	$\mathcal{G}$	;
L08:	Block $\gg$	End	;
L09:	pcdeduction $\rhd$ L08 $\gg$	$\mathcal{F} \vdash \mathcal{G}$	;
L10:	Block $\gg$	Begin	;
L11:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}$	;
L12:	Premise $\gg$	$\mathcal{F}$	;
L13:	pcmp $\rhd$ L12 $\rhd$ L02 $\gg$	$\mathcal{G} \wedge \neg\mathcal{G}$	;
L14:	andelim2 $\rhd$ L13 $\gg$	$\neg\mathcal{G}$	;
L15:	Block $\gg$	End	;
L16:	pcdeduction $\rhd$ L15 $\gg$	$\mathcal{F} \vdash \neg\mathcal{G}$	;
L17:	notintro $\rhd$ L09 $\rhd$ L16 $\gg$	$\neg\mathcal{F}$	$\square$

6 Application of natural deduction lemmas

In this section we set out to prove lemma 6.6 which is the main result of this project. The content of the lemma is simple, but the formal proof is quite long. Therefore we have split it into six lemmas. Finally we prove an easy result containing quantifiers in lemma 6.8. We have also tried to prove $\Pi \mathcal{F}, \mathcal{G}, \mathcal{X}, \mathcal{H}, \mathcal{P} \colon \mathcal{X} \# \mathcal{G} \Vdash \mathcal{X} \# \mathcal{H} \Vdash \langle \mathcal{H} {\equiv} \mathcal{F} | \mathcal{X} {:=} \mathcal{G} \rangle \Vdash \mathcal{P} \Rightarrow \forall \mathcal{X}.\, (\mathcal{F}) \vdash \neg\mathcal{P} \Rightarrow \exists \mathcal{X}.\, (\mathcal{F}) \vdash \exists \mathcal{X}.\, (\mathcal{F})$

but we ran into problems when we tried to used side conditions within blocks. Either we got a false side condition or a metageneralization error. Therefore these attempts have been removed from our final report.

Lemma 6.1 [*pred calc* **lemma** *hlplem1*: $\Pi \mathcal{P}, \mathcal{Q} \colon (\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q} \vdash \mathcal{Q} \Rightarrow \mathcal{P} \vdash \mathcal{P} \Rightarrow \mathcal{Q} \vdash \mathcal{P}$]

pred calc **proof of** hlplem1:

L01:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L02:	Premise $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}$	;
L03:	Premise $\gg$	$\mathcal{Q} \Rightarrow \mathcal{P}$	;
L04:	Premise $\gg$	$\mathcal{P} \Rightarrow \mathcal{Q}$	;
L05:	pcmp $\rhd$ L04 $\rhd$ L02 $\gg$	$\mathcal{Q}$	;
L06:	pcmp $\rhd$ L05 $\rhd$ L03 $\gg$	$\mathcal{P}$	$\square$

Lemma 6.2 [*pred calc* **lemma** *hlplem2*: $\Pi \mathcal{P}, \mathcal{Q} \colon \mathcal{P} \vdash \neg\mathcal{P} \vdash \mathcal{Q}$]

pred calc **proof of** hlplem2:

L01:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;

L02:	Premise $\gg$	$\mathcal{P}$	;
L03:	Premise $\gg$	$\neg\mathcal{P}$	;
L04:	andintro $\rhd$ L02 $\rhd$ L03 $\gg$	$\mathcal{P} \wedge \neg\mathcal{P}$	;
L05:	bottomelim $\rhd$ L04 $\gg$	$\mathcal{Q}$	$\square$

Lemma 6.3 [*pred calc* **lemma** *hlplem3*: $\Pi\mathcal{P},\mathcal{Q}\colon \neg(\mathcal{P} \Rightarrow \mathcal{Q}) \vdash \neg\mathcal{P} \vdash (\mathcal{P} \Rightarrow \mathcal{Q}) \wedge \neg(\mathcal{P} \Rightarrow \mathcal{Q})$]

pred calc **proof of** hlplem3:

L01:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L02:	Premise $\gg$	$\neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L03:	Premise $\gg$	$\neg\mathcal{P}$	;
L04:	Block $\gg$	Begin	;
L05:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L06:	Premise $\gg$	$\mathcal{P}$	;
L07:	hlplem2 $\rhd$ L06 $\rhd$ L03 $\gg$	$\mathcal{Q}$	;
L08:	Block $\gg$	End	;
L09:	pcdeduction $\rhd$ L08 $\gg$	$\mathcal{P} \Rightarrow \mathcal{Q}$	;
L10:	andintro $\rhd$ L09 $\rhd$ L02 $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \wedge \neg(\mathcal{P} \Rightarrow \mathcal{Q})$	$\square$

Lemma 6.4 [*pred calc* **lemma** *hlplem4*: $\Pi\mathcal{P},\mathcal{Q}\colon \neg(\mathcal{P} \Rightarrow \mathcal{Q}) \vdash \mathcal{P}$]

pred calc **proof of** hlplem4:

L01:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L02:	Premise $\gg$	$\neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L03:	Block $\gg$	Begin	;
L04:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L05:	Premise $\gg$	$\neg\mathcal{P}$	;
L06:	hlplem3 $\rhd$ L02 $\rhd$ L05 $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \wedge \neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L07:	Block $\gg$	End	;
L08:	pcdeduction $\rhd$ L07 $\gg$	$\neg\mathcal{P} \Rightarrow (\mathcal{P} \Rightarrow \mathcal{Q}) \wedge \neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L09:	lemnotintro $\rhd$ L08 $\gg$	$\neg\neg\mathcal{P}$	;
L10:	notnotelim $\rhd$ L09 $\gg$	$\mathcal{P}$	$\square$

Lemma 6.5 [*pred calc* **lemma** *hlplem5*: $\Pi\mathcal{P},\mathcal{Q}\colon (\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q} \vdash \mathcal{Q} \Rightarrow \mathcal{P} \vdash \mathcal{P}$]

pred calc **proof of** hlplem5:

L01:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L02:	Premise $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}$	;
L03:	Premise $\gg$	$\mathcal{Q} \Rightarrow \mathcal{P}$	;
L04:	Block $\gg$	Begin	;
L05:	Arbitrary $\gg$	$\mathcal{P},\mathcal{Q}$	;
L06:	Premise $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}$	;
L07:	Premise $\gg$	$\mathcal{Q} \Rightarrow \mathcal{P}$	;
L08:	Premise $\gg$	$\mathcal{P} \Rightarrow \mathcal{Q}$	;
L09:	hlplem1 $\rhd$ L06 $\rhd$ L07 $\rhd$ L08 $\gg$	$\mathcal{P}$	;

L10:	Block $\gg$	End	;
L11:	pcdeduction $\rhd$ L10 $\gg$	$((\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}) \Rightarrow (\mathcal{Q} \Rightarrow \mathcal{P}) \Rightarrow (\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{P}$	;
L12:	pcmp $\rhd$ L02 $\rhd$ L11 $\gg$	$(\mathcal{Q} \Rightarrow \mathcal{P}) \Rightarrow (\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{P}$	;
L13:	pcmp $\rhd$ L03 $\rhd$ L12 $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{P}$	;
L14:	Block $\gg$	Begin	;
L15:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L16:	Premise $\gg$	$\mathcal{P} \Rightarrow \mathcal{Q}$	;
L17:	pcmp $\rhd$ L16 $\rhd$ L13 $\gg$	$\mathcal{P}$	;
L18:	Block $\gg$	End	;
L19:	pcdeduction $\rhd$ L18 $\gg$	$\mathcal{P} \Rightarrow \mathcal{Q} \vdash \mathcal{P}$	;
L20:	Block $\gg$	Begin	;
L21:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L22:	Premise $\gg$	$\neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L23:	hlplem4 $\rhd$ L22 $\gg$	$\mathcal{P}$	;
L24:	Block $\gg$	End	;
L25:	pcdeduction $\rhd$ L24 $\gg$	$\neg(\mathcal{P} \Rightarrow \mathcal{Q}) \vdash \mathcal{P}$	;
L26:	lem $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \vee \neg(\mathcal{P} \Rightarrow \mathcal{Q})$	;
L27:	orelim $\rhd$ L26 $\rhd$ L19 $\rhd$ L25 $\gg$	$\mathcal{P}$	□

Lemma 6.6 (Main result) [*pred calc* **lemma** *goal1*: $\Pi\mathcal{P}, \mathcal{Q}$: $((\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}) \Rightarrow ((\mathcal{Q} \Rightarrow \mathcal{P}) \Rightarrow \mathcal{P})$]

pred calc **proof of** goal1:

L01:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L02:	Block $\gg$	Begin	;
L03:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L04:	Premise $\gg$	$(\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}$	;
L05:	Block $\gg$	Begin	;
L06:	Arbitrary $\gg$	$\mathcal{P}, \mathcal{Q}$	;
L07:	Premise $\gg$	$\mathcal{Q} \Rightarrow \mathcal{P}$	;
L08:	hlplem5 $\rhd$ L04 $\rhd$ L07 $\gg$	$\mathcal{P}$	;
L09:	Block $\gg$	End	;
L10:	pcdeduction $\rhd$ L09 $\gg$	$(\mathcal{Q} \Rightarrow \mathcal{P}) \Rightarrow \mathcal{P}$	;
L11:	Block $\gg$	End	;
L12:	pcdeduction $\rhd$ L11 $\gg$	$((\mathcal{P} \Rightarrow \mathcal{Q}) \Rightarrow \mathcal{Q}) \Rightarrow ((\mathcal{Q} \Rightarrow \mathcal{P}) \Rightarrow \mathcal{P})$	□

Lemma 6.7
[*pred calc* **lemma** *hlplem6*: $\Pi\mathcal{X}, \mathcal{F}, \mathcal{G}, \mathcal{H}$: $\mathcal{X}\#\mathcal{G} \Vdash \mathcal{X}\#\mathcal{H} \Vdash \langle\mathcal{H}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{G}\rangle \Vdash \forall\mathcal{X}.\,(\mathcal{F}) \vdash \exists\mathcal{X}.\,(\mathcal{F})$]

pred calc **proof of** hlplem6:

L01:	Arbitrary $\gg$	$\mathcal{X}, \mathcal{F}, \mathcal{G}, \mathcal{H}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G}$	;
L03:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{H}$	;
L04:	Side-condition $\gg$	$\langle\mathcal{H}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{G}\rangle$	;

L05:	Premise $\gg$	$\forall \mathcal{X}.\,(\mathcal{F})$	;
L06:	forallelim $\unrhd$ L02 $\unrhd$ L03 $\unrhd$ L04 $\rhd$ L05 $\gg$	$\mathcal{H}$	;
L07:	existsintro $\unrhd$ L02 $\unrhd$ L03 $\unrhd$ L04 $\rhd$ L06 $\gg$	$\exists \mathcal{X}.\,(\mathcal{F})$	$\Box$

Lemma 6.8
[*pred calc* **lemma** *goal2*: $\Pi \mathcal{F}, \mathcal{G}, \mathcal{X}, \mathcal{H}, \mathcal{K}\colon \mathcal{X}\#\mathcal{G} \Vdash \mathcal{X}\#\mathcal{H} \Vdash \langle \mathcal{H}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{G}\rangle \Vdash \forall \mathcal{X}.\,(\mathcal{F}) \vdash \neg\exists \mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{K}$]

pred calc **proof of** goal2:

L01:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{G}, \mathcal{X}, \mathcal{H}, \mathcal{K}$	;
L02:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{G}$	;
L03:	Side-condition $\gg$	$\mathcal{X}\#\mathcal{H}$	;
L04:	Side-condition $\gg$	$\langle \mathcal{H}{\equiv}\mathcal{F}|\mathcal{X}{:=}\mathcal{G}\rangle$	;
L05:	Premise $\gg$	$\forall \mathcal{X}.\,(\mathcal{F})$	;
L06:	hlplem6 $\unrhd$ L02 $\unrhd$ L03 $\unrhd$ L04 $\rhd$ L05 $\gg$	$\exists \mathcal{X}.\,(\mathcal{F})$	;
L07:	Block $\gg$	Begin	;
L08:	Arbitrary $\gg$	$\mathcal{F}, \mathcal{X}, \mathcal{K}$	;
L09:	Premise $\gg$	$\exists \mathcal{X}.\,(\mathcal{F})$	;
L10:	Premise $\gg$	$\neg\exists \mathcal{X}.\,(\mathcal{F})$	;
L11:	andintro $\rhd$ L09 $\rhd$ L10 $\gg$	$\exists \mathcal{X}.\,(\mathcal{F}) \wedge \neg\exists \mathcal{X}.\,(\mathcal{F})$	;
L12:	bottomelim $\rhd$ L11 $\gg$	$\mathcal{K}$	;
L13:	Block $\gg$	End	;
L14:	pcdeduction $\rhd$ L13 $\gg$	$\exists \mathcal{X}.\,(\mathcal{F}) \Rightarrow \neg\exists \mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{K}$	;
L15:	pcmp $\rhd$ L06 $\rhd$ L14 $\gg$	$\neg\exists \mathcal{X}.\,(\mathcal{F}) \Rightarrow \mathcal{K}$	$\Box$

A Ways Logiweb™ could be improved

Working with Logiweb™ have given us some ideas of how the system could be improved and how it could be made more userfriendly. Our ideas are summarized below:

- A getting started manual explaining the basic concepts and quirks of the system including simplified examples. Man pages together with the base and check pages leaves a lot to be desired.
- Sample pages / proofs for the current version of the pyk compiler (we have used a lot of time searching other peoples Logiweb™ pages to gain knowledge but it has often been in vain because the syntax had changed).
- Better error messages and / or an explanation of how to read them. Also it would be nice if error messages included the line number of the error.
- It would be nice if LaTeX errors and warnings could be displayed. Just knowing that there is an error leaves a lot of detective work to be done.

- A pre-parser substituting `(` with `parenthesis`, `)` with `end parenthesis`, `,` with `comma` ect. would be nice.

References

[LiCS] Logic in Computer Science - Modelling and Reasoning about Systems, Second Edition - 2004.
By Michael Huth & Mark Ryan
ISBN: 0-521-54310-X

B Pyk definitions

([pred calc $\stackrel{\text{pyk}}{\rightarrow}$ "pred calc"]
[pc1 $\stackrel{\text{pyk}}{\rightarrow}$ "pc1"]
[pc2 $\stackrel{\text{pyk}}{\rightarrow}$ "pc2"]
[pc3 $\stackrel{\text{pyk}}{\rightarrow}$ "pc3"]
[pc4 $\stackrel{\text{pyk}}{\rightarrow}$ "pc4"]
[pc5 $\stackrel{\text{pyk}}{\rightarrow}$ "pc5"]
[pc6 $\stackrel{\text{pyk}}{\rightarrow}$ "pc6"]
[pc7 $\stackrel{\text{pyk}}{\rightarrow}$ "pc7"]
[pc8 $\stackrel{\text{pyk}}{\rightarrow}$ "pc8"]
[pc9 $\stackrel{\text{pyk}}{\rightarrow}$ "pc9"]
[pc10 $\stackrel{\text{pyk}}{\rightarrow}$ "pc10"]
[pc11 $\stackrel{\text{pyk}}{\rightarrow}$ "pc11"]
[pc12 $\stackrel{\text{pyk}}{\rightarrow}$ "pc12"]
[pcmp $\stackrel{\text{pyk}}{\rightarrow}$ "pcmp"]
[pcunsound $\stackrel{\text{pyk}}{\rightarrow}$ "pcunsound"]
[pcia $\stackrel{\text{pyk}}{\rightarrow}$ "pcia"]
[pcie $\stackrel{\text{pyk}}{\rightarrow}$ "pcie"]
[pcdeduction $\stackrel{\text{pyk}}{\rightarrow}$ "pcdeduction"]
[trivia $\stackrel{\text{pyk}}{\rightarrow}$ "trivia"]
[repeat $\stackrel{\text{pyk}}{\rightarrow}$ "repeat"]
[andintro $\stackrel{\text{pyk}}{\rightarrow}$ "andintro"]
[andelim1 $\stackrel{\text{pyk}}{\rightarrow}$ "andelim1"]
[andelim2 $\stackrel{\text{pyk}}{\rightarrow}$ "andelim2"]
[orintro1 $\stackrel{\text{pyk}}{\rightarrow}$ "orintro1"]
[orintro2 $\stackrel{\text{pyk}}{\rightarrow}$ "orintro2"]

[orelim $\stackrel{\text{pyk}}{\to}$ "orelim"]
[notintro $\stackrel{\text{pyk}}{\to}$ "notintro"]
[implyintro $\stackrel{\text{pyk}}{\to}$ "implyintro"]
[notnotintro $\stackrel{\text{pyk}}{\to}$ "notnotintro"]
[notnotelim $\stackrel{\text{pyk}}{\to}$ "notnotelim"]
[mt $\stackrel{\text{pyk}}{\to}$ "mt"]
[pbc $\stackrel{\text{pyk}}{\to}$ "pbc"]
[lem $\stackrel{\text{pyk}}{\to}$ "lem"]
[forallintro $\stackrel{\text{pyk}}{\to}$ "forallintro"]
[forallelim $\stackrel{\text{pyk}}{\to}$ "forallelim"]
[existsintro $\stackrel{\text{pyk}}{\to}$ "existsintro"]
[existselim $\stackrel{\text{pyk}}{\to}$ "existselim"]
[bottomelim $\stackrel{\text{pyk}}{\to}$ "bottomelim"]
[lemnotintro $\stackrel{\text{pyk}}{\to}$ "lemnotintro"]
[hlplem1 $\stackrel{\text{pyk}}{\to}$ "hlplem1"]
[hlplem2 $\stackrel{\text{pyk}}{\to}$ "hlplem2"]
[hlplem3 $\stackrel{\text{pyk}}{\to}$ "hlplem3"]
[hlplem4 $\stackrel{\text{pyk}}{\to}$ "hlplem4"]
[hlplem5 $\stackrel{\text{pyk}}{\to}$ "hlplem5"]
[goal1 $\stackrel{\text{pyk}}{\to}$ "goal1"]
[hlplem6 $\stackrel{\text{pyk}}{\to}$ "hlplem6"]
[goal2 $\stackrel{\text{pyk}}{\to}$ "goal2"]
[$* \equiv *$ $\stackrel{\text{pyk}}{\to}$ "" setequiv ""]
[$* = *$ $\stackrel{\text{pyk}}{\to}$ "" setequals ""]
[$\neg *$ $\stackrel{\text{pyk}}{\to}$ "lnot ""]
[$* \wedge *$ $\stackrel{\text{pyk}}{\to}$ "" land ""]
[$* \vee *$ $\stackrel{\text{pyk}}{\to}$ "" lor ""]
[$\forall * . (*)$ $\stackrel{\text{pyk}}{\to}$ "forall " dot " end forall"]
[$\exists * . (*)$ $\stackrel{\text{pyk}}{\to}$ "exists " dot " end exists"]
[$* \in *$ $\stackrel{\text{pyk}}{\to}$ "" setin ""]
[test $\stackrel{\text{pyk}}{\to}$ "test"]
)$^{\mathbf{P}}$

C Tex definitions

- [$\neg \mathsf{x}$ $\stackrel{\text{tex}}{=}$ "\neg #1."]

- [$\mathsf{x} \wedge \mathsf{y} \stackrel{\text{tex}}{=}$ “#1. \wedge #2.”]
- [$\mathsf{x} \vee \mathsf{y} \stackrel{\text{tex}}{=}$ “#1. \vee #2.”]
- [$\mathsf{x} \Rightarrow \mathsf{y} \stackrel{\text{tex}}{=}$ “#1. \Rightarrow #2.”]
- [$\forall \mathsf{y}.\, (\mathsf{b}) \stackrel{\text{tex}}{=}$ “\forall #1. . \left(#2.\right)”]
- [$\exists \mathsf{y}.\, (\mathsf{b}) \stackrel{\text{tex}}{=}$ “\exists #1. . \left(#2.\right)”]
- [$\mathsf{y} \in \mathsf{b} \stackrel{\text{tex}}{=}$ “#1. \in #2.”]
- [$\mathsf{y} \equiv \mathsf{b} \stackrel{\text{tex}}{=}$ “#1. \equiv #2.”]
- [$\mathsf{y} = \mathsf{b} \stackrel{\text{tex}}{=}$ “#1. = #2.”]

D Priority table

Priority table
Preassociative
[test], [base], [bracket $*$ end bracket], [big bracket $*$ end bracket], [\$ $*$ \$], [**flush left** [$*$]], [x], [y], [z], [[$* \bowtie *$]], [[$* \xrightarrow{*} *$]], [pyk], [tex], [name], [prio], [$*$], [T], [if($*, *, *$)], [[$* \stackrel{*}{\Rightarrow} *$]], [val], [claim], [$\bot$], [f($*$)], [$(*)^{\mathrm{I}}$], [F], [$\underline{0}$], [$\underline{1}$], [$\underline{2}$], [$\underline{3}$], [$\underline{4}$], [$\underline{5}$], [$\underline{6}$], [$\underline{7}$], [$\underline{8}$], [$\underline{9}$], [0], [1], [2], [3], [4], [5], [6], [7], [8], [9], [a], [b], [c], [d], [e], [f], [g], [h], [i], [j], [k], [l], [m], [n], [o], [p], [q], [r], [s], [t], [u], [v], [w], [$(*)^{\mathrm{M}}$], [If($*, *, *$)], [array{$*$} $*$ end array], [l], [c], [r], [empty], [$\langle * \mid * := * \rangle$], [$\mathcal{M}(*)$], [$\tilde{\mathcal{U}}(*)$], [$\mathcal{U}(*)$], [$\mathcal{U}^{\mathrm{M}}(*)$], [**apply**($*, *$)], [$\mathbf{apply}_1(*, *)$], [identifier($*$)], [$\mathrm{identifier}_1(*, *)$], [array-plus($*, *$)], [array-remove($*, *, *$)], [array-put($*, *, *, *$)], [array-add($*, *, *, *, *$)], [bit($*, *$)], [$\mathrm{bit}_1(*, *)$], [rack], ["vector"], ["bibliography"], ["dictionary"], ["body"], ["codex"], ["expansion"], ["code"], ["cache"], ["diagnose"], ["pyk"], ["tex"], ["texname"], ["value"], ["message"], ["macro"], ["definition"], ["unpack"], ["claim"], ["priority"], ["lambda"], ["apply"], ["true"], ["if"], ["quote"], ["proclaim"], ["define"], ["introduce"], ["hide"], ["pre"], ["post"], [$\mathcal{E}(*, *, *)$], [$\mathcal{E}_2(*, *, *, *, *)$], [$\mathcal{E}_3(*, *, *, *)$], [$\mathcal{E}_4(*, *, *, *)$], [**lookup**($*, *, *$)], [**abstract**($*, *, *, *$)], [[$\lceil * \rceil$]], [$\mathcal{M}(*, *, *)$], [$\mathcal{M}_2(*, *, *, *)$], [$\mathcal{M}^*(*, *, *)$], [macro], [$\mathrm{s}_0$], [**zip**($*, *$)], [$\mathbf{assoc}_1(*, *, *)$], [$(*)^{\mathbf{p}}$], [self], [[$* \stackrel{..}{=} *$]], [[$* \doteq *$]], [[$* \stackrel{\prime}{=} *$]], [[$* \stackrel{\text{pyk}}{=} *$]], [[$* \stackrel{\text{tex}}{=} *$]], [[$* \stackrel{\text{name}}{=} *$]], [**Priority table**[$*$]], [$\tilde{\mathcal{M}}_1$], [$\tilde{\mathcal{M}}_2(*)$], [$\tilde{\mathcal{M}}_3(*)$], [$\tilde{\mathcal{M}}_4(*, *, *, *)$], [$\tilde{\mathcal{M}}(*, *, *)$], [$\tilde{\mathcal{Q}}(*, *, *)$], [$\tilde{\mathcal{Q}}_2(*, *, *)$], [$\tilde{\mathcal{Q}}_3(*, *, *, *)$], [$\tilde{\mathcal{Q}}^*(*, *, *)$], [($*$)], [($*$)], [display($*$)], [statement($*$)], [[$*]^{\cdot}$], [[$*]^{-}$], [**aspect**($*, *$)], [**aspect**($*, *, *$)], [$\langle * \rangle$], [$\mathbf{tuple}_1(*)$], [$\mathbf{tuple}_2(*)$], [$\mathrm{let}_2(*, *)$], [$\mathrm{let}_1(*, *)$], [[$* \stackrel{\text{claim}}{=} *$]], [checker], [**check**($*, *$)], [$\mathbf{check}_2(*, *, *)$], [$\mathbf{check}_3(*, *, *)$], [$\mathbf{check}^*(*, *)$], [$\mathbf{check}^*_2(*, *, *)$], [[$*]^{\cdot}$], [[$*]^{-}$], [[$*]^{\circ}$], [msg], [[$* \stackrel{\text{msg}}{=} *$]], [<stmt>], [stmt], [[$* \stackrel{\text{stmt}}{=} *$]], [HeadNil′], [HeadPair′], [Transitivity′], [$\bot\!\!\bot$], [Contra′], [T'_{E}], [L_1], [$\underline{*}$], [$\mathcal{A}$], [$\mathcal{B}$], [$\mathcal{C}$], [$\mathcal{D}$], [$\mathcal{E}$], [$\mathcal{F}$], [$\mathcal{G}$], [$\mathcal{H}$], [$\mathcal{I}$], [$\mathcal{J}$], [$\mathcal{K}$], [$\mathcal{L}$], [$\mathcal{M}$], [$\mathcal{N}$], [$\mathcal{O}$], [$\mathcal{P}$], [$\mathcal{Q}$], [$\mathcal{R}$], [$\mathcal{S}$], [$\mathcal{T}$], [$\mathcal{U}$], [$\mathcal{V}$], [$\mathcal{W}$], [$\mathcal{X}$], [$\mathcal{Y}$], [$\mathcal{Z}$], [$\langle * \mid * := * \rangle$], [$\langle^* * \mid * := * \rangle$], [$\emptyset$], [Remainder],

$[(*)^{\mathbf{v}}]$, [intro(*, *, *, *)], [intro(*, *, *)], [error(*, *)], [error$_2$(*, *)], [proof(*, *, *)], [proof$_2$(*, *)], $[\mathcal{S}(*, *)]$, $[\mathcal{S}^{\mathrm{I}}(*, *)]$, $[\mathcal{S}^{\rhd}(*, *)]$, $[\mathcal{S}_1^{\rhd}(*, *, *)]$, $[\mathcal{S}^{\mathrm{E}}(*, *)]$, $[\mathcal{S}_1^{\mathrm{E}}(*, *, *)]$, $[\mathcal{S}^{+}(*, *)]$, $[\mathcal{S}_1^{+}(*, *, *)]$, $[\mathcal{S}^{-}(*, *)]$, $[\mathcal{S}_1^{-}(*, *, *)]$, $[\mathcal{S}^{*}(*, *)]$, $[\mathcal{S}_1^{*}(*, *, *)]$, $[\mathcal{S}_2^{*}(*, *, *, *)]$, $[\mathcal{S}^{@}(*, *)]$, $[\mathcal{S}_1^{@}(*, *, *)]$, $[\mathcal{S}^{\vdash}(*, *)]$, $[\mathcal{S}_1^{\vdash}(*, *, *, *)]$, $[\mathcal{S}^{\Vdash}(*, *)]$, $[\mathcal{S}_1^{\Vdash}(*, *, *, *)]$, $[\mathcal{S}^{\mathrm{i.e.}}(*, *)]$, $[\mathcal{S}_1^{\mathrm{i.e.}}(*, *, *, *)]$, $[\mathcal{S}_2^{\mathrm{i.e.}}(*, *, *, *, *)]$, $[\mathcal{S}^{\forall}(*, *)]$, $[\mathcal{S}_1^{\forall}(*, *, *, *)]$, $[\mathcal{S}^{;}(*, *)]$, $[\mathcal{S}_1^{;}(*, *, *)]$, $[\mathcal{S}_2^{;}(*, *, *, *)]$, $[\mathcal{T}(*)]$, [claims(*, *, *)], [claims$_2$(*, *, *)], [<proof>], [proof], [[**Lemma** *: *]], [[**Proof of** *: *]], [[* **lemma** *: *]], [[* **antilemma** *: *]], [[* **rule** *: *]], [[* **antirule** *: *]], [verifier], $[\mathcal{V}_1(*)]$, $[\mathcal{V}_2(*, *)]$, $[\mathcal{V}_3(*, *, *, *)]$, $[\mathcal{V}_4(*, *)]$, $[\mathcal{V}_5(*, *, *, *)]$, $[\mathcal{V}_6(*, *, *, *)]$, $[\mathcal{V}_7(*, *, *, *)]$, [Cut(*, *)], [Head$_\oplus$(*)], [Tail$_\oplus$(*)], [rule$_1$(*, *)], [rule(*, *)], [Rule tactic], [Plus(*, *)], [[**Theory** *]], [theory$_2$(*, *)], [theory$_3$(*, *)], [theory$_4$(*, *, *)], [HeadNil″], [HeadPair″], [Transitivity″], [Contra″], [HeadNil], [HeadPair], [Transitivity], [Contra], [T$_{\mathrm{E}}$], [ragged right], [ragged right expansion], [parm(*, *, *)], [parm*(*, *, *)], [inst(*, *)], [inst*(*, *)], [occur(*, *, *)], [occur*(*, *, *)], [unify(* = *, *)], [unify*(* = *, *)], [unify$_2$(* = *, *)], [L$_{\mathrm{a}}$], [L$_{\mathrm{b}}$], [L$_{\mathrm{c}}$], [L$_{\mathrm{d}}$], [L$_{\mathrm{e}}$], [L$_{\mathrm{f}}$], [L$_{\mathrm{g}}$], [L$_{\mathrm{h}}$], [L$_{\mathrm{i}}$], [L$_{\mathrm{j}}$], [L$_{\mathrm{k}}$], [L$_{\mathrm{l}}$], [L$_{\mathrm{m}}$], [L$_{\mathrm{n}}$], [L$_{\mathrm{o}}$], [L$_{\mathrm{p}}$], [L$_{\mathrm{q}}$], [L$_{\mathrm{r}}$], [L$_{\mathrm{s}}$], [L$_{\mathrm{t}}$], [L$_{\mathrm{u}}$], [L$_{\mathrm{v}}$], [L$_{\mathrm{w}}$], [L$_{\mathrm{x}}$], [L$_{\mathrm{y}}$], [L$_{\mathrm{z}}$], [L$_{\mathrm{A}}$], [L$_{\mathrm{B}}$], [L$_{\mathrm{C}}$], [L$_{\mathrm{D}}$], [L$_{\mathrm{E}}$], [L$_{\mathrm{F}}$], [L$_{\mathrm{G}}$], [L$_{\mathrm{H}}$], [L$_{\mathrm{I}}$], [L$_{\mathrm{J}}$], [L$_{\mathrm{K}}$], [L$_{\mathrm{L}}$], [L$_{\mathrm{M}}$], [L$_{\mathrm{N}}$], [L$_{\mathrm{O}}$], [L$_{\mathrm{P}}$], [L$_{\mathrm{Q}}$], [L$_{\mathrm{R}}$], [L$_{\mathrm{S}}$], [L$_{\mathrm{T}}$], [L$_{\mathrm{U}}$], [L$_{\mathrm{V}}$], [L$_{\mathrm{W}}$], [L$_{\mathrm{X}}$], [L$_{\mathrm{Y}}$], [L$_{\mathrm{Z}}$], [L$_{?}$], [Reflexivity], [Reflexivity$_1$], [Commutativity], [Commutativity$_1$], [<tactic>], [tactic], $[[* \overset{\mathrm{tactic}}{=} *]]$, $[\mathcal{P}(*, *, *)]$, $[\mathcal{P}^*(*, *, *)]$, [p$_0$], [conclude$_1$(*, *)], [conclude$_2$(*, *, *)], [conclude$_3$(*, *, *, *)], [conclude$_4$(*, *)], [check], $[[* \overset{\circ}{=} *]]$, [RootVisible(*)], [A], [R], [C], [T], [L], [{*}], $[\overline{*}]$, [a], [b], [c], [d], [e], [f], [g], [h], [i], [j], [k], [l], [m], [n], [o], [p], [q], [r], [s], [t], [u], [v], [w], [x], [y], [z], $[\langle * \equiv * \mid * := * \rangle]$, $[\langle * \equiv^0 * \mid * := * \rangle]$, $[\langle * \equiv^1 * \mid * := * \rangle]$, $[\langle * \equiv^* * \mid * := * \rangle]$, [Ded(*, *)], [Ded$_0$(*, *)], [Ded$_1$(*, *, *)], [Ded$_2$(*, *, *)], [Ded$_3$(*, *, *, *)], [Ded$_4$(*, *, *, *)], [Ded$_4^*$(*, *, *, *)], [Ded$_5$(*, *, *)], [Ded$_6$(*, *, *, *)], [Ded$_6^*$(*, *, *, *)], [Ded$_7$(*)], [Ded$_8$(*, *)], [Ded$_8^*$(*, *)], [S], [Neg], [MP], [Gen], [Ded], [S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [Repetition], [A1′], [A2′], [A4′], [A5′], [Prop 3.2a], [Prop 3.2b], [Prop 3.2c], [Prop 3.2d], [Prop 3.2e$_1$], [Prop 3.2e$_2$], [Prop 3.2e], [Prop 3.2f$_1$], [Prop 3.2f$_2$], [Prop 3.2f], [Prop 3.2g$_1$], [Prop 3.2g$_2$], [Prop 3.2g], [Prop 3.2h$_1$], [Prop 3.2h$_2$], [Prop 3.2h], [Block$_1$(*, *, *)], [Block$_2$(*)], [pred calc], [pc1], [pc2], [pc3], [pc4], [pc5], [pc6], [pc7], [pc8], [pc9], [pc10], [pc11], [pc12], [pcmp], [pcunsound], [pcia], [pcie], [pcdeduction], [trivia], [repeat], [andintro], [andelim1], [andelim2], [orintro1], [orintro2], [orelim], [notintro], [implyintro], [notnotintro], [notnotelim], [mt], [pbc], [lem], [forallintro], [forallelim], [existsintro], [existselim], [bottomelim], [lemnotintro], [hlplem1], [hlplem2], [hlplem3], [hlplem4], [hlplem5], [goal1], [hlplem6], [goal2];

Preassociative

[*_{*}], [*/indexintro(*, *, *, *)], [*/intro(*, *, *)], [*/bothintro(*, *, *, *, *)], [*/nameintro(*, *, *, *)], [*′], [*[*]], $[*[* \rightarrow *]]$, $[*[* \Rightarrow *]]$, [*0], [*1], [0b], [*-color(*)], [*-color*(*)], $[*^{\mathrm{H}}]$, $[*^{\mathrm{T}}]$, $[*^{\mathrm{U}}]$, $[*^{\mathrm{h}}]$, $[*^{\mathrm{t}}]$, $[*^{\mathrm{s}}]$, $[*^{\mathrm{c}}]$, $[*^{\mathrm{d}}]$, $[*^{\mathrm{a}}]$, $[*^{\mathrm{C}}]$, $[*^{\mathrm{M}}]$, $[*^{\mathrm{B}}]$, $[*^{\mathrm{r}}]$, $[*^{\mathrm{i}}]$, $[*^{\mathrm{d}}]$, $[*^{\mathrm{R}}]$, $[*^{0}]$, $[*^{1}]$, $[*^{2}]$, $[*^{3}]$, $[*^{4}]$, $[*^{5}]$, $[*^{6}]$, $[*^{7}]$, $[*^{8}]$, $[*^{9}]$, $[*^{\mathrm{E}}]$, $[*^{\mathcal{V}}]$, $[*^{\mathcal{C}}]$, $[*^{\mathcal{C}^*}]$, $[*^{\mathrm{hide}}]$;

Preassociative

[“ * ”], [], $[(*)^{\mathbf{t}}]$, [string(*) + *], [string(*) ++ *], [

∗], [∗], [!∗], ["∗], [#∗], [$∗], [%∗], [&∗], [’∗], [(∗], [)∗], [∗∗], [+∗], [, ∗], [-∗], [.∗], [/∗], [0∗], [1∗], [2∗], [3∗], [4∗], [5∗], [6∗], [7∗], [8∗], [9∗], [:∗], [; ∗], [<∗], [=∗], [>∗], [?∗], [@∗], [A∗], [B∗], [C∗], [D∗], [E∗], [F∗], [G∗], [H∗], [I∗], [J∗], [K∗], [L∗], [M∗], [N∗], [O∗], [P∗], [Q∗], [R∗], [S∗], [T∗], [U∗], [V∗], [W∗], [X∗], [Y∗], [Z∗], [[∗], [\∗], []∗], [ˆ∗], [_∗], [‘∗], [a∗], [b∗], [c∗], [d∗], [e∗], [f∗], [g∗], [h∗], [i∗], [j∗], [k∗], [l∗], [m∗], [n∗], [o∗], [p∗], [q∗], [r∗], [s∗], [t∗], [u∗], [v∗], [w∗], [x∗], [y∗], [z∗], [{∗], [|∗], [}∗], [˜∗], [**Preassociative** ∗; ∗], [**Postassociative** ∗; ∗], [[∗], ∗], [priority ∗ end], [newline ∗], [macro newline ∗], [MacroIndent(∗)];

Preassociative

[∗ ’ ∗], [∗ ‘ ∗];

Preassociative

$[*']$;

Preassociative

[∗ ’ ∗], [∗ ‘ ∗];

Preassociative

$[* \cdot *], [* \cdot_0 *]$;

Preassociative

$[* + *], [* +_0 *], [* +_1 *], [* - *], [* -_0 *], [* -_1 *]$;

Preassociative

$[* \cup \{*\}], [* \cup *], [* \backslash \{*\}]$;

Postassociative

$[* \therefore *], [* \underline{\therefore} *], [* \underline{::} *], [* \underline{+2*} *], [* :: *], [* +2* *]$;

Postassociative

$[*, *]$;

Preassociative

$[* \stackrel{\mathrm{B}}{\approx} *], [* \stackrel{\mathrm{D}}{\approx} *], [* \stackrel{\mathrm{C}}{\approx} *], [* \stackrel{\mathrm{P}}{\approx} *], [* \approx *], [* = *], [* \stackrel{+}{\to} *], [* \stackrel{\mathrm{t}}{=} *], [* \stackrel{\mathrm{t}^*}{=} *], [* \stackrel{\mathrm{r}}{=} *]$, $[* \in_{\mathrm{t}} *], [* \subseteq_{\mathrm{T}} *], [* \stackrel{\mathrm{T}}{=} *], [* \stackrel{\mathrm{s}}{=} *]$, [∗ free in ∗], [∗ free in* ∗], [∗ free for ∗ in ∗], [∗ free for* ∗ in ∗], $[* \in_{\mathrm{c}} *], [* < *], [* <' *], [* \leq' *], [* = *], [* \neq *], [*^{\mathrm{var}}]$, $[*\#^0*], [*\#^1*], [*\#^**], [* \equiv *], [* = *]$;

Preassociative

$[\neg *], [\neg *]$;

Preassociative

$[* \wedge *], [* \ddot{\wedge} *], [* \tilde{\wedge} *], [* \wedge_{\mathrm{c}} *], [* \wedge *]$;

Preassociative

$[* \vee *], [* \parallel *], [* \ddot{\vee} *], [* \vee *]$;

Preassociative

$[\exists * \colon *], [\forall * \colon *], [\forall_{\mathrm{obj}} * \colon *], [\forall * . (*)], [\exists * . (*)]$;

Postassociative

$[* \ddot{\Rightarrow} *], [* \Rightarrow *], [* \Leftrightarrow *]$;

Postassociative

[∗ : ∗], [∗ spy ∗], [∗!∗];

Preassociative

$[* \left\{ \begin{array}{c} * \\ * \end{array} \right.]$;

Preassociative

$[\lambda * .*], [\Lambda * .*], [\Lambda *], [\textbf{if } * \textbf{ then } * \textbf{ else } *], [\textbf{let } * = * \textbf{ in } *], [\textbf{let } * \ddot{=} * \text{ in } *];$
Preassociative
$[*\#*];$
Preassociative
$[*^{\mathrm{I}}], [*^{\triangleright}], [*^{\mathrm{V}}], [*^{+}], [*^{-}], [*^{*}];$
Preassociative
$[* @ *], [* \triangleright *], [* \,\triangleright\!\!\triangleright\, *], [* \gg *], [* \trianglerighteq *];$
Postassociative
$[* \vdash *], [* \Vdash *], [* \text{ i.e. } *];$
Preassociative
$[\forall * \colon *], [\Pi * \colon *];$
Postassociative
$[* \oplus *];$
Postassociative
$[*; *];$
Preassociative
$[* \text{ proves } *];$
Preassociative
$[* \textbf{ proof of } * : *], [\text{Line} * : * \gg *; *], [\text{Last line} * \gg * \square],$
$[\text{Line} * : \text{Premise} \gg *; *], [\text{Line} * : \text{Side-condition} \gg *; *], [\text{Arbitrary} \gg *; *],$
$[\text{Local} \gg * = *; *], [\text{Begin} *; * : \text{End}; *], [\text{Last block line} * \gg *;],$
$[\text{Arbitrary} \gg *; *];$
Postassociative
$[* \mid *];$
Postassociative
$[* , *], [*[*]*];$
Preassociative
$[*\&*], [\rightarrow];$
Preassociative
$[*\backslash\backslash *], [* \text{ linebreak[4] } *], [*\backslash\backslash *];$
Preassociative
$[* \in *];$ **End table**