Propositional Calculus I:
The Formal Language,
The Prop. Calc. Oracle (= AI),
Application to Some Motivating Problems

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

IFLAI1 [Intro to (Formal) Logic (and AI)]
1/27/2022

RAIR
Rensselaer AI and Reasoning Lab

How’d We Arrive Here?

(Selmer’s Leibnizian Whirlwind History of Logic, With Discussion of The Singularity)

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Intro to Logic
1/24/2022

RAIR
Rensselaer AI and Reasoning Lab

How’d We Arrive Here?

(Selmer’s Leibnizian Whirlwind History of Logic, With Discussion of The Singularity)

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Questions about last time …?

Intro to Logic
1/24/2022

Rensselaer AI and Reasoning Lab

Logistics …

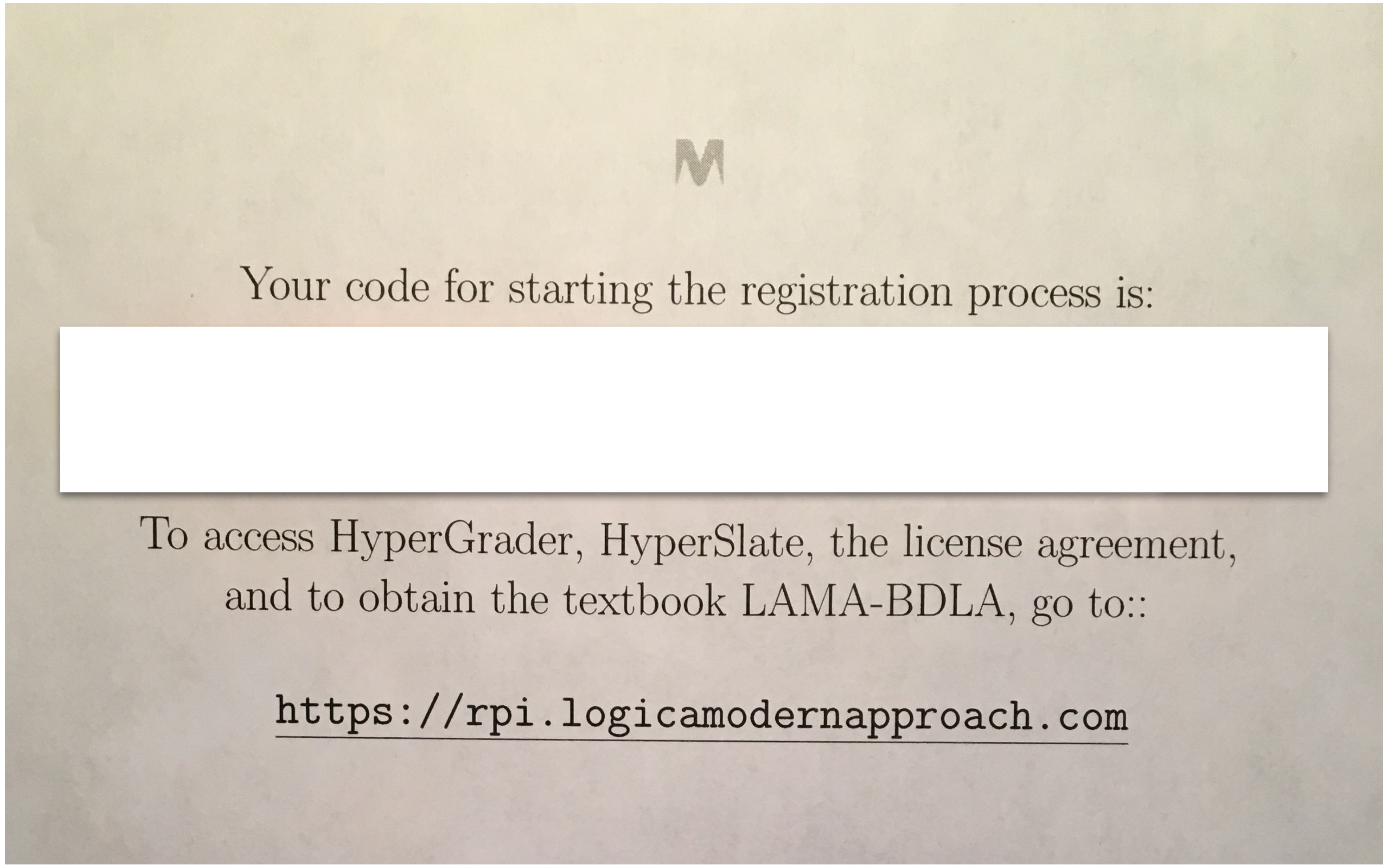
The Starting Code to Purchase in Bookstore

Your code for starting the registration process is:

To access HyperGrader, HyperSlate, the license agreement, and to obtain the textbook LAMA-BDLA, go to::

https://rpi.logicamodernapproach.com

The Starting Code to Purchase in Bookstore

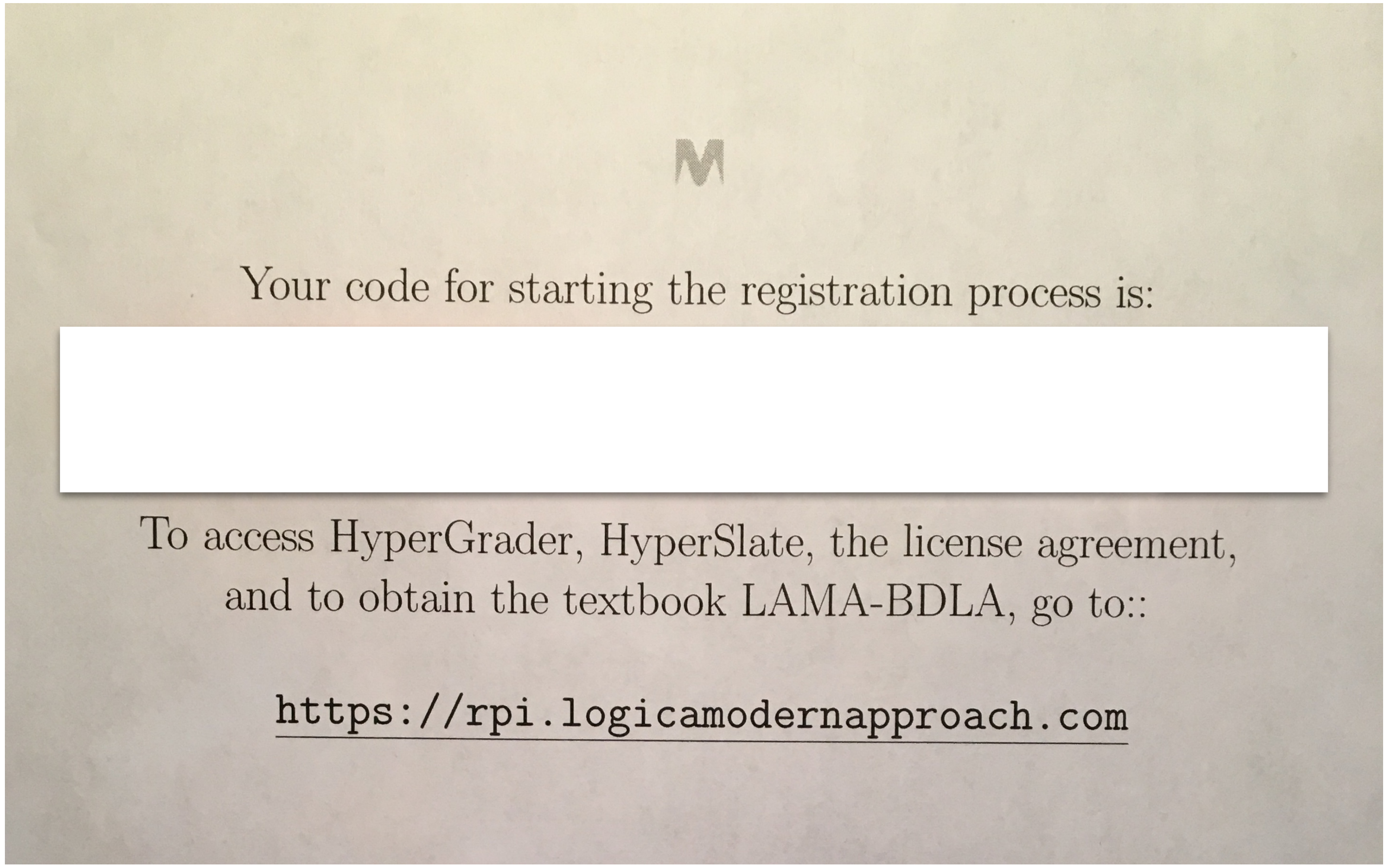

Your code for starting the registration process is:

To access HyperGrader, HyperSlate, the license agreement, and to obtain the textbook LAMA-BDLA, go to::

https://rpi.logicamodernapproach.com

Once seal broken on envelope, no return. Remember from first class, any reservations, opt for "Stanford" paradigm, with its software instead of LAMA® paradigm!

The Starting Code to Purchase in Bookstore



Your code for starting the registration process is:

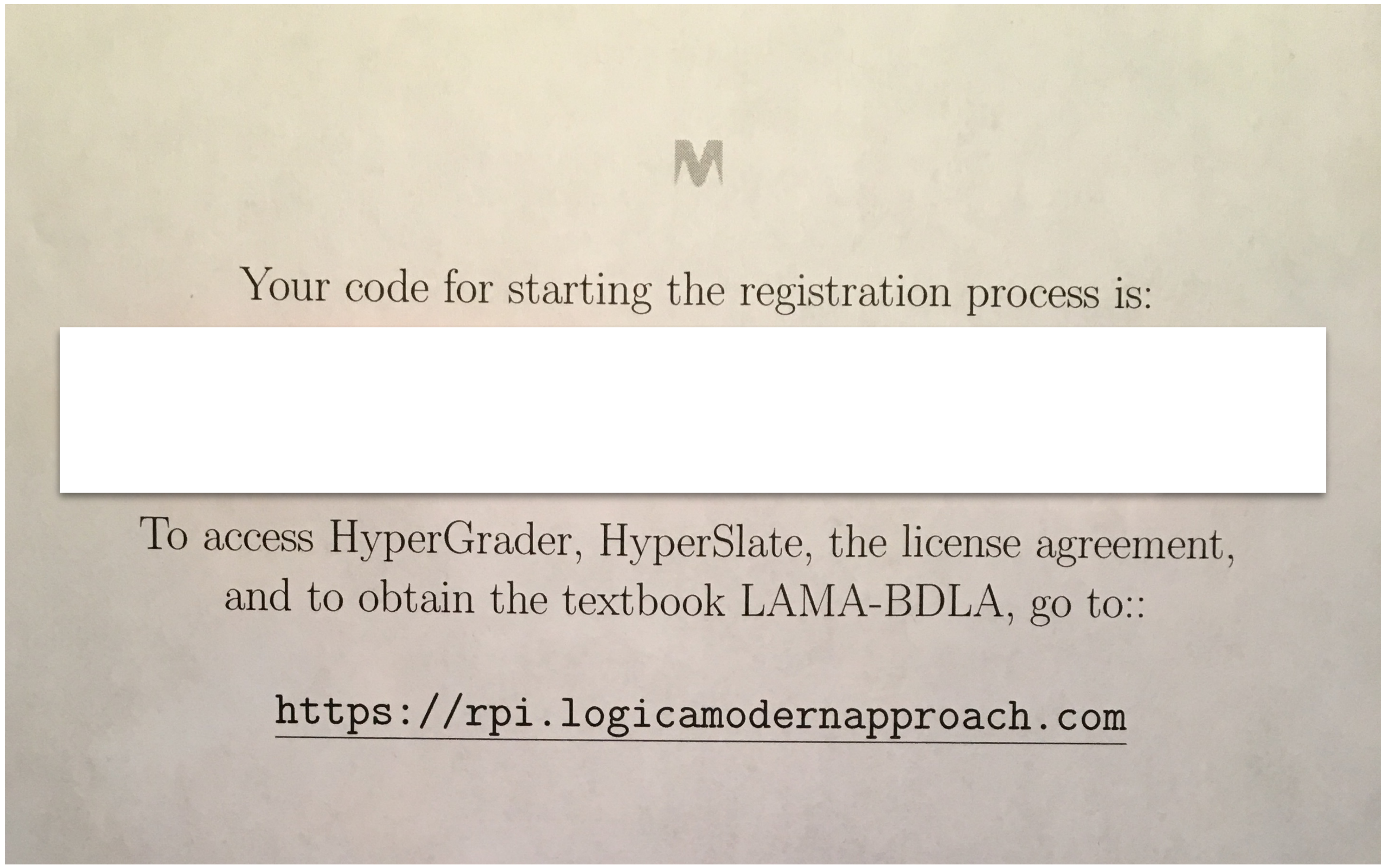
To access HyperGrader, HyperSlate, the license agreement, and to obtain the textbook LAMA-BDLA, go to::

https://rpi.logicamodernapproach.com

Once seal broken on envelope, no return. Remember from first class, any reservations, opt for "Stanford" paradigm, with its software instead of LAMA® paradigm!

The email address you enter is case-sensitive!

The Starting Code to Purchase in Bookstore



Your code for starting the registration process is:

To access HyperGrader, HyperSlate, the license agreement, and to obtain the textbook LAMA-BDLA, go to::

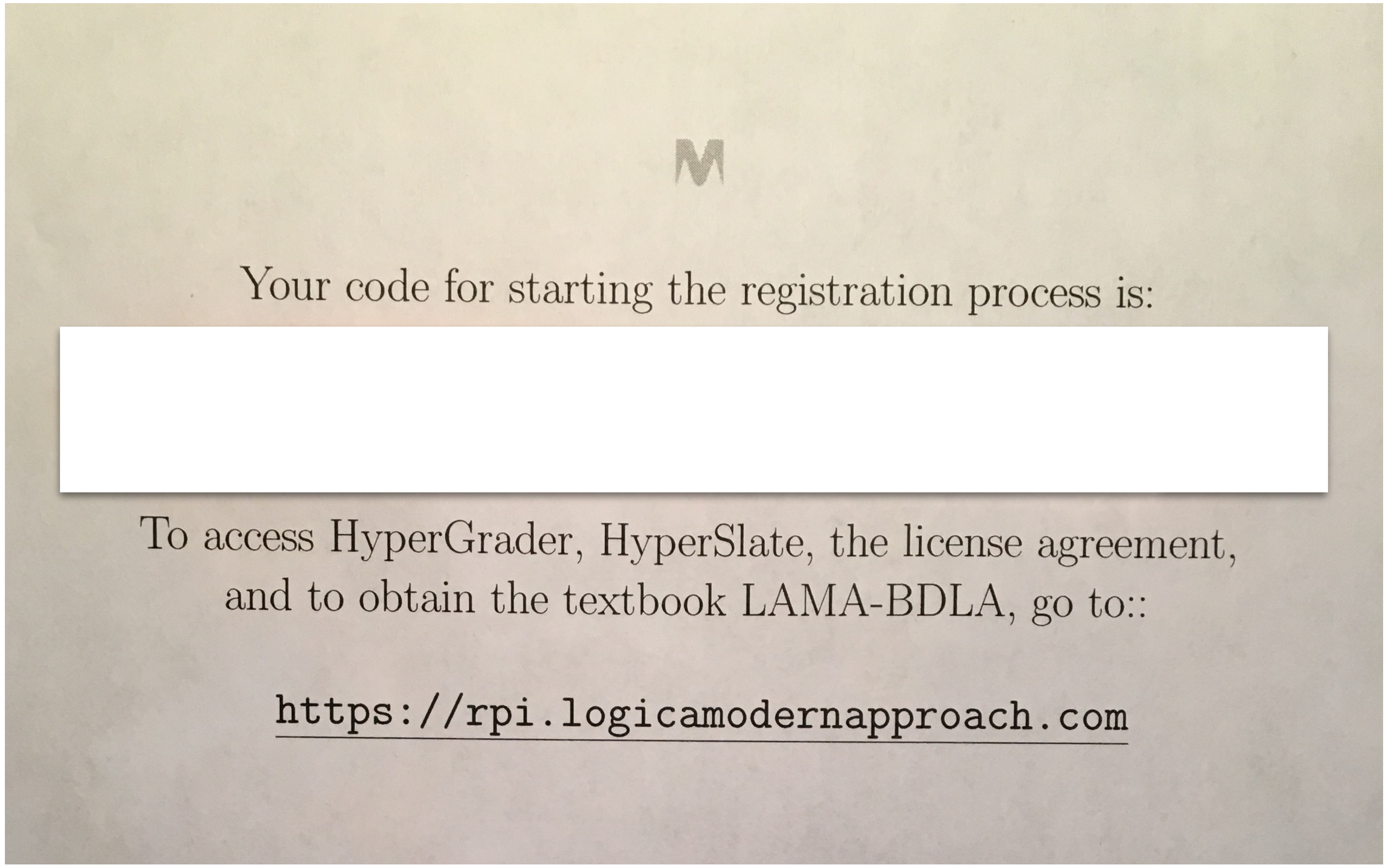
https://rpi.logicamodernapproach.com

Once seal broken on envelope, no return. Remember from first class, any reservations, opt for "Stanford" paradigm, with its software instead of LAMA® paradigm!

The email address you enter is case-sensitive!

Your OS and browser must be fully up-to-date; Chrome is the best choice, browser-wise (though I use Safari).

The Starting Code to Purchase in Bookstore


Your code for starting the registration process is:

To access HyperGrader, HyperSlate, the license agreement, and to obtain the textbook LAMA-BDLA, go to::

https://rpi.logicamodernapproach.com

Once seal broken on envelope, no return. Remember from first class, any reservations, opt for "Stanford" paradigm, with its software instead of LAMA® paradigm!

The email address you enter is case-sensitive!

Your OS and browser must be fully up-to-date; Chrome is the best choice, browser-wise (though I use Safari).

Watch that the link emailed to you doesn't end up being classified as spam.

Introduction to (Formal) Logic (and AI)

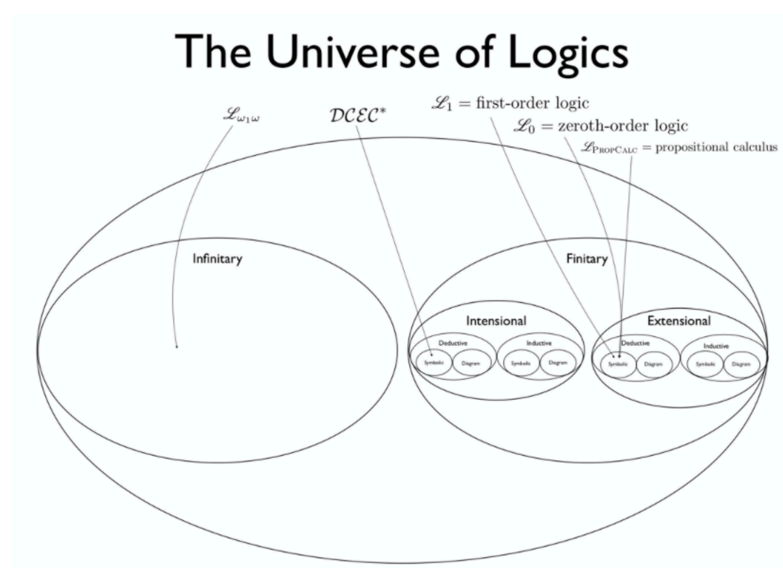
Spring 2021 edition of IFLAI1

Selmer Bringsjord

Table of Contents

A fully online course, thanks to singular AI technology.

with Naveen Sundar G.
∧ KB Foushée ∧ Joshua Taylor ∧ …



Micro-homily:

Micro-homily:

skipping to ~ p. 34!

Micro-homily:

skipping to ~ p. 34!

Micro-homily:

skipping to ~ p. 34!

M. Chi: Self-testers end up being self-made.

Micro-homily:

skipping to ~ p. 34!

M. Chi: Self-testers end up being self-made.

Micro-homily:

skipping to ~ p. 34!

M. Chi: Self-testers end up being self-made.

"What category of English sentences does logic focus on?"

The Formal Language

CHAPTER 2. PROPOSITIONAL CALCULUS

Syntax	Formula Type	Sample Representation
$\mathsf{P}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{Q}, \mathsf{Q}_1, \ldots$	Atomic Formulas	"Larry is lucky." as L_l
$\neg\phi$	Negation	"Gary isn't lucky." as $\neg\mathsf{L}_g$
$\phi_1 \wedge \ldots \wedge \phi_n$	Conjunction	"Both Larry and Carl are lucky." as $\mathsf{L}_l \wedge \mathsf{L}_c$
$\phi_1 \vee \ldots \vee \phi_n$	Disjunction	"Either Billy is lucky or Alvin is." as $\mathsf{L}_b \vee \mathsf{L}_a$
$\phi \rightarrow \psi$	Conditional (Implication)	"If Ron is lucky, so is Frank." as $\mathsf{L}_r \rightarrow L_f$
$\phi \leftrightarrow \psi$	Biconditional (Coimplication)	"Tim is lucky if and only if Kim is." as $\mathsf{L}_t \longleftrightarrow L_k$

Table 2.1: Syntax of the Propositional Calculus. Note that ϕ, ψ, and ϕ_i stand for arbitrary formulas.

The Formal Language

CHAPTER 2. PROPOSITIONAL CALCULUS

Syntax	Formula Type	Sample Representation
$\mathsf{P}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{Q}, \mathsf{Q}_1, \ldots$	Atomic Formulas	"Larry is lucky." as L_l
$\neg\phi$	Negation	"Gary isn't lucky." as $\neg\mathsf{L}_g$
$\phi_1 \wedge \ldots \wedge \phi_n$	Conjunction	"Both Larry and Carl are lucky." as $\mathsf{L}_l \wedge \mathsf{L}_c$
$\phi_1 \vee \ldots \vee \phi_n$	Disjunction	"Either Billy is lucky or Alvin is." as $\mathsf{L}_b \vee \mathsf{L}_a$
$\phi \rightarrow \psi$	Conditional (Implication)	"If Ron is lucky, so is Frank." as $\mathsf{L}_r \rightarrow L_f$
$\phi \leftrightarrow \psi$	Biconditional (Coimplication)	"Tim is lucky if and only if Kim is." as $\mathsf{L}_t \longleftrightarrow L_k$

Table 2.1: Syntax of the Propositional Calculus. Note that ϕ, ψ, and ϕ_i stand for arbitrary formulas.

Exercise: Is this language Roger-decidable? Prove it!

The Formal Language

(presented as formal grammar)

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
& | & (\textit{Formula Connective Formula}) \\
& | & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

The Formal Language

(presented as formal grammar)

$$\begin{array}{lcl} \mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\ & | & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\ & | & \neg\ \mathit{Formula} \end{array}$$

$$\mathit{AtomicFormula} \Rightarrow \mathrm{P}_1 \mid \mathrm{P}_2 \mid \mathrm{P}_3 \mid \ldots$$

$$\mathit{Connective} \Rightarrow \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$$

Exercise: Is this language Roger-decidable? Prove it!

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
 & & \\
 & \mid & (\textit{Formula}\ \ \textit{Connective}\ \ \textit{Formula}) \\
 & \mid & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \ \vee \ \mid \ \rightarrow \ \mid \ \leftrightarrow
\end{array}
$$

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid\ \vee\ \mid\ \rightarrow\ \mid\ \leftrightarrow
\end{array}
$$

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
& \mid & (\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula}) \\
& \mid & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

p

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & \mid & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & \mid & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \ \vee \ \mid \ \rightarrow \ \mid \ \leftrightarrow
\end{array}
$$

p P

As S-expressions

Formula	$\Rightarrow$	*AtomicFormula*
	$\mid$	(*Formula Connective Formula*)
	$\mid$	$\neg$ *Formula*
AtomicFormula	$\Rightarrow$	$\texttt{P}_1 \mid \texttt{P}_2 \mid \texttt{P}_3 \mid \ldots$
Connective	$\Rightarrow$	$\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

```
p   P   bradywillbeback
```

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \ \vee \ \mid \ \rightarrow \ \mid \ \leftrightarrow
\end{array}
$$

`p   P   bradywillbeback  P26`

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
 & | & (\textit{Formula Connective Formula}) \\
 & | & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \texttt{P}_1 \mid \texttt{P}_2 \mid \texttt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

`p   P   bradywillbeback   P26   ...`

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

`p   P   bradywillbeback   P26   ...`

As S-expressions

$\mathit{Formula}$	$\Rightarrow$	$\mathit{AtomicFormula}$
	$\mid$	$(\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula})$
	$\mid$	$\neg\ \mathit{Formula}$
$\mathit{AtomicFormula}$	$\Rightarrow$	$\mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots$
$\mathit{Connective}$	$\Rightarrow$	$\land \mid \lor \mid \rightarrow \mid \leftrightarrow$

`p   P   bradywillbeback   P26   ...`

`(not p)`

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & \mid & (\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula}) \\
 & \mid & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

`p   P   bradywillbeback  P26   ...`

`(not p)  (not P)`

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
 & | & (\textit{Formula}\ \textit{Connective}\ \textit{Formula}) \\
 & | & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

`p` `P` `bradywillbeback` `P26` ...

`(not p)` `(not P)` `(not P26)`

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

```
p   P   bradywillbeback  P26    ...

(not p)  (not P)  (not P26)     ...
```

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

```
p   P   bradywillbeback  P26   ...

(not p)  (not P)  (not P26)    ...
```

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
 & | & (\textit{Formula}\ \ \textit{Connective}\ \ \textit{Formula}) \\
 & | & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

p P bradywillbeback P26 ...

(not p) (not P) (not P26) ...

(and P Q)

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
& \mid & (\textit{Formula} \;\; \textit{Connective} \;\; \textit{Formula}) \\
& \mid & \neg\; \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

```
p   P   bradywillbeback   P26   ...

(not p)  (not P)  (not P26)   ...

(and P Q)  (or P Q)
```

As S-expressions

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula} \\
\hline
\mathit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

```
p   P   bradywillbeback  P26   ...

(not p)  (not P)  (not P26)    ...

(and P Q)  (or P Q) (if P Q)
```

As S-expressions

$$
\begin{array}{lcl}
\textit{Formula} & \Rightarrow & \textit{AtomicFormula} \\
 & | & (\textit{Formula}\ \textit{Connective}\ \textit{Formula}) \\
 & | & \neg\ \textit{Formula} \\
\hline
\textit{AtomicFormula} & \Rightarrow & \mathtt{P}_1 \mid \mathtt{P}_2 \mid \mathtt{P}_3 \mid \ldots \\
\hline
\textit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

`p   P   bradywillbeback   P26   ...`

`(not p)   (not P)   (not P26)   ...`

`(and P Q)   (or P Q)   (if P Q)   (iff P Q)`

Better Formal Language: Pure Predicate Calculus (presented via formal grammar)

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & | & (\mathit{Formula}\ \mathit{Connective}\ \mathit{Formula}) \\
 & | & \neg\ \mathit{Formula}
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{AtomicFormula} & \Rightarrow & (\mathit{Predicate}\ \mathit{Term}_1 \ldots \mathit{Term}_k)
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Term} & \Rightarrow & (\mathit{Function}\ \mathit{Term}_1\ \ldots\ \mathit{Term}_k) \\
 & | & \mathit{Constant} \\
 & | & \mathit{Variable}
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Predicate} & \Rightarrow & P_1 \mid P_2 \mid P_3\ \ldots \\
\mathit{Constant} & \Rightarrow & c_1 \mid c_2 \mid c_3\ \ldots \\
\mathit{Variable} & \Rightarrow & v_1 \mid v_2 \mid v_3\ \ldots \\
\mathit{Function} & \Rightarrow & f_1 \mid f_2 \mid f_3\ \ldots
\end{array}
$$

Better Formal Language: Pure Predicate Calculus (presented via formal grammar)

$$
\begin{array}{lcl}
\mathit{Formula} & \Rightarrow & \mathit{AtomicFormula} \\
 & \mid & (\mathit{Formula}\ \ \mathit{Connective}\ \ \mathit{Formula}) \\
 & \mid & \neg\ \mathit{Formula}
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{AtomicFormula} & \Rightarrow & (\mathit{Predicate}\ \ \mathit{Term}_1 \ldots \mathit{Term}_k)
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Term} & \Rightarrow & (\mathit{Function}\ \ \mathit{Term}_1\ \ \ldots\ \ \mathit{Term}_k) \\
 & \mid & \mathit{Constant} \\
 & \mid & \mathit{Variable}
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Connective} & \Rightarrow & \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow
\end{array}
$$

$$
\begin{array}{lcl}
\mathit{Predicate} & \Rightarrow & P_1 \mid P_2 \mid P_3\ \ldots \\
\mathit{Constant} & \Rightarrow & c_1 \mid c_2 \mid c_3\ \ldots \\
\mathit{Variable} & \Rightarrow & v_1 \mid v_2 \mid v_3\ \ldots \\
\mathit{Function} & \Rightarrow & f_1 \mid f_2 \mid f_3\ \ldots
\end{array}
$$

Exercise: Is this language also Roger-decidable? Prove it!

“NYS 1” Revisited

Given the statements

$\neg a \vee \neg b$

b

$c \rightarrow a$

which one of the following statements is provable?

c

$\neg b$

$\neg c$

h

a

none of the above

“NYS 1” Revisited

Given the statements

¬a ∨ ¬b

b

c → a

which one of the following statements is provable?

c

¬b

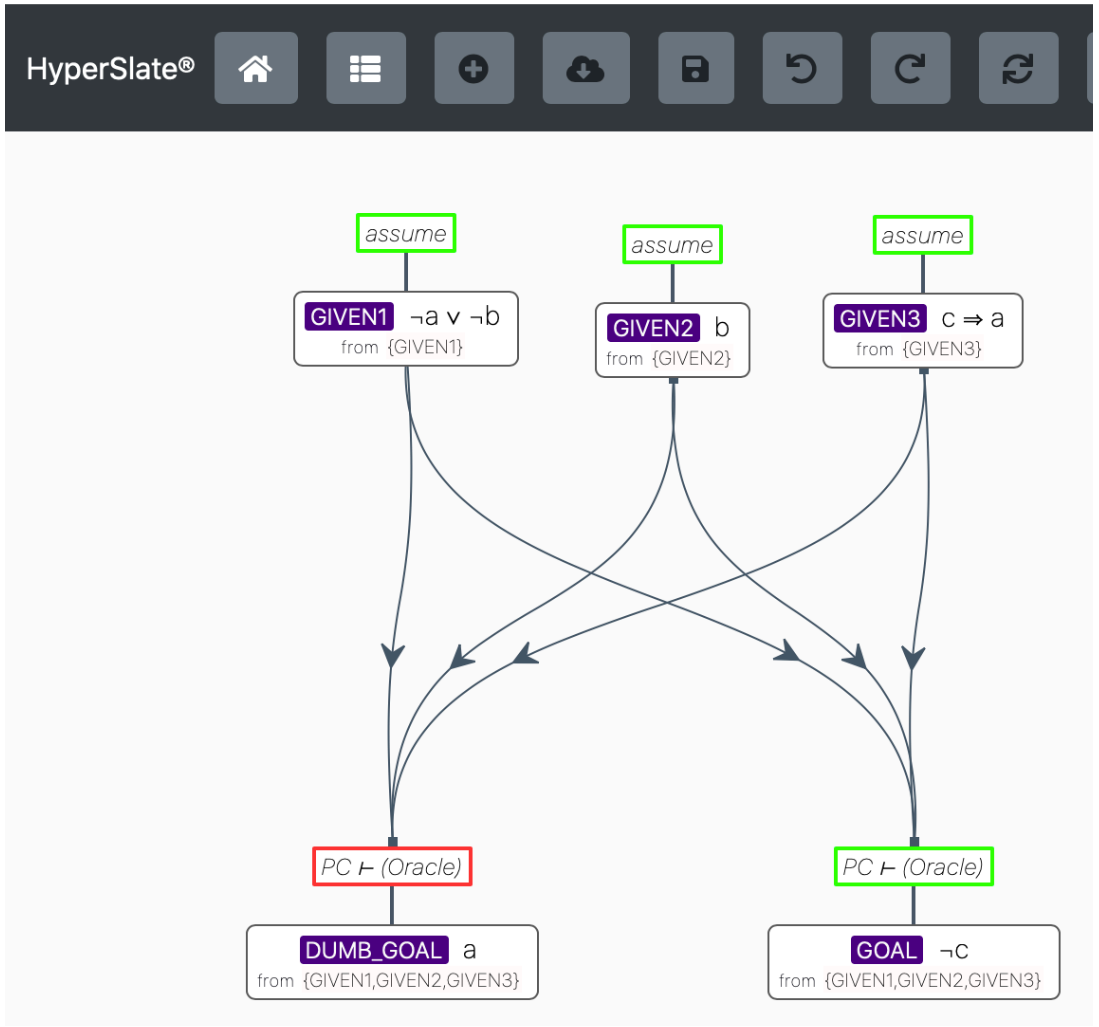
¬c

h

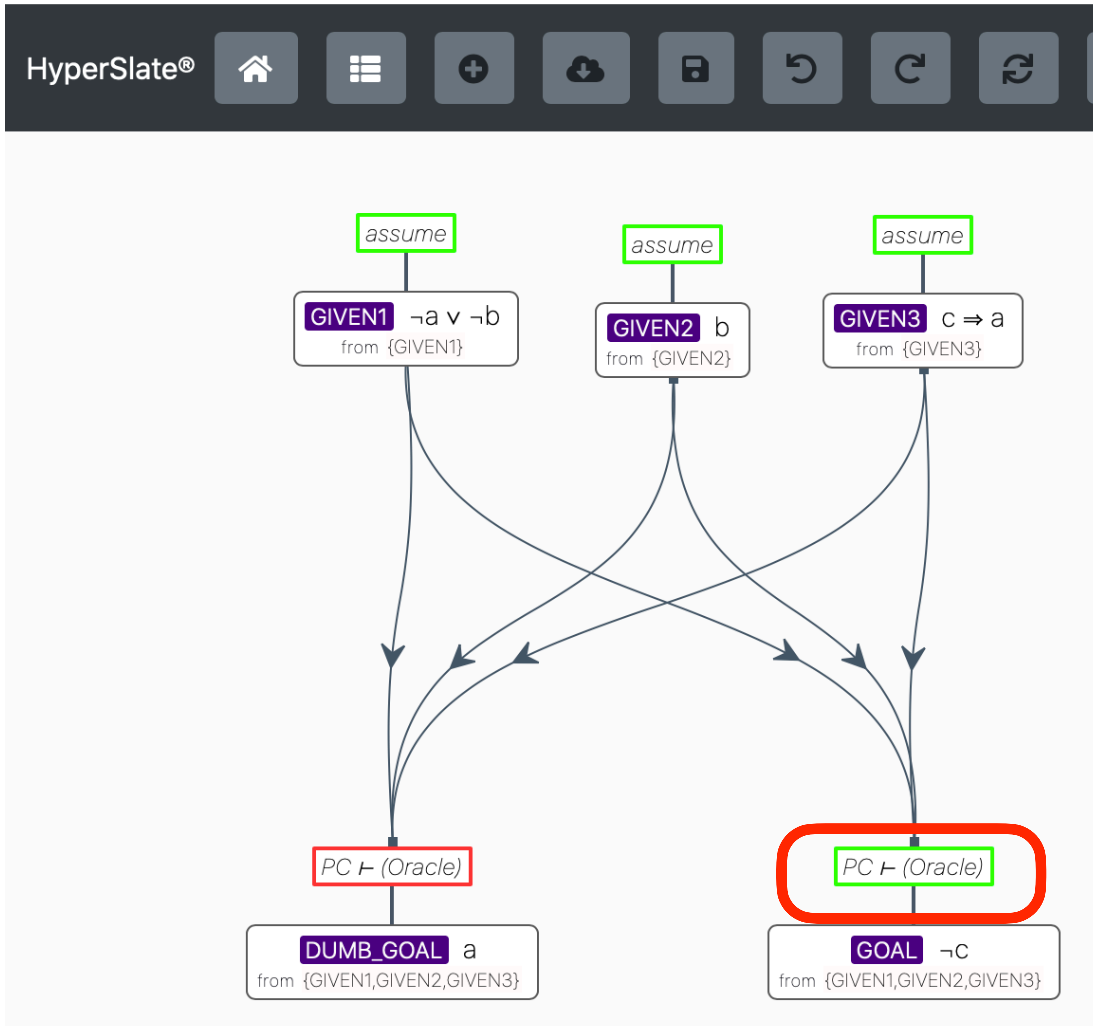
a

none of the above

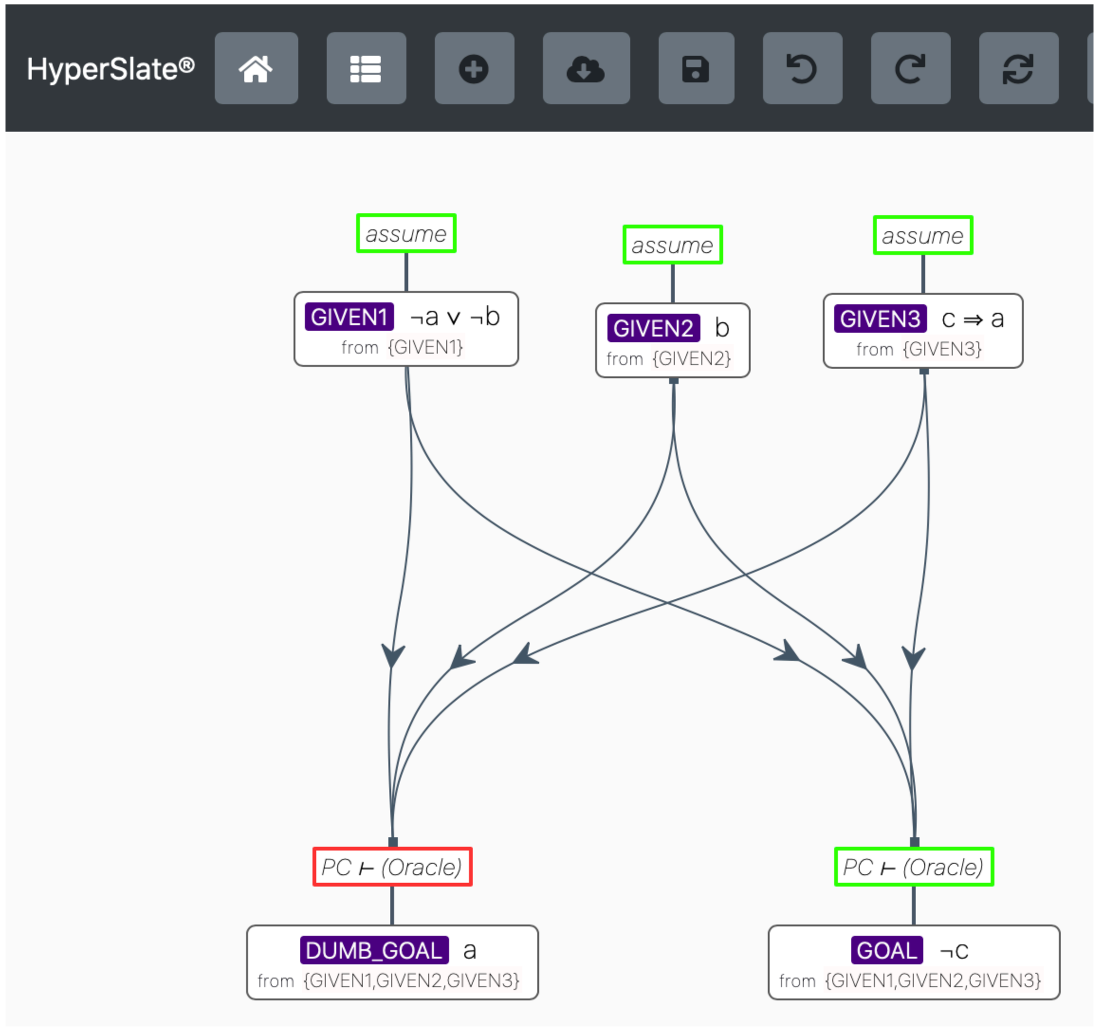
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



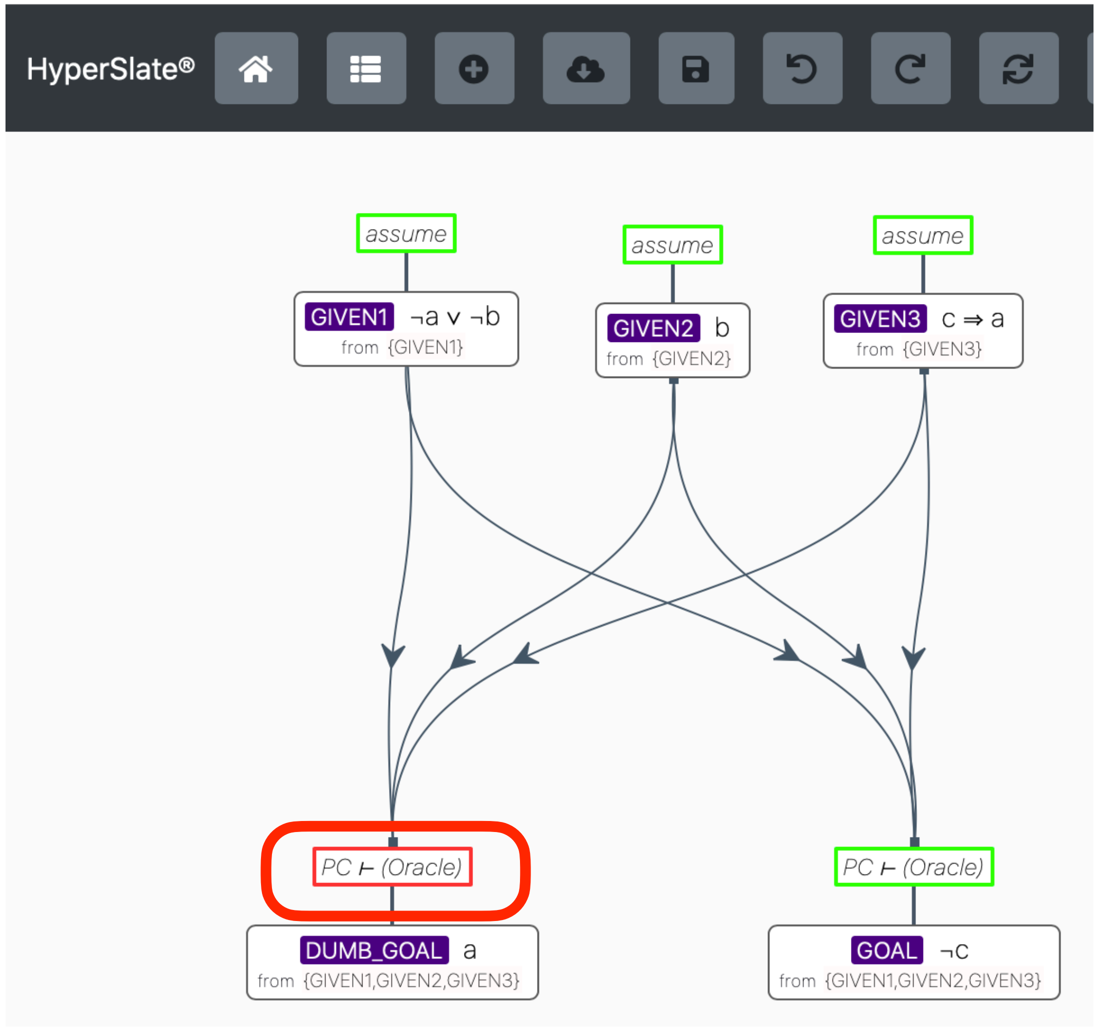
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



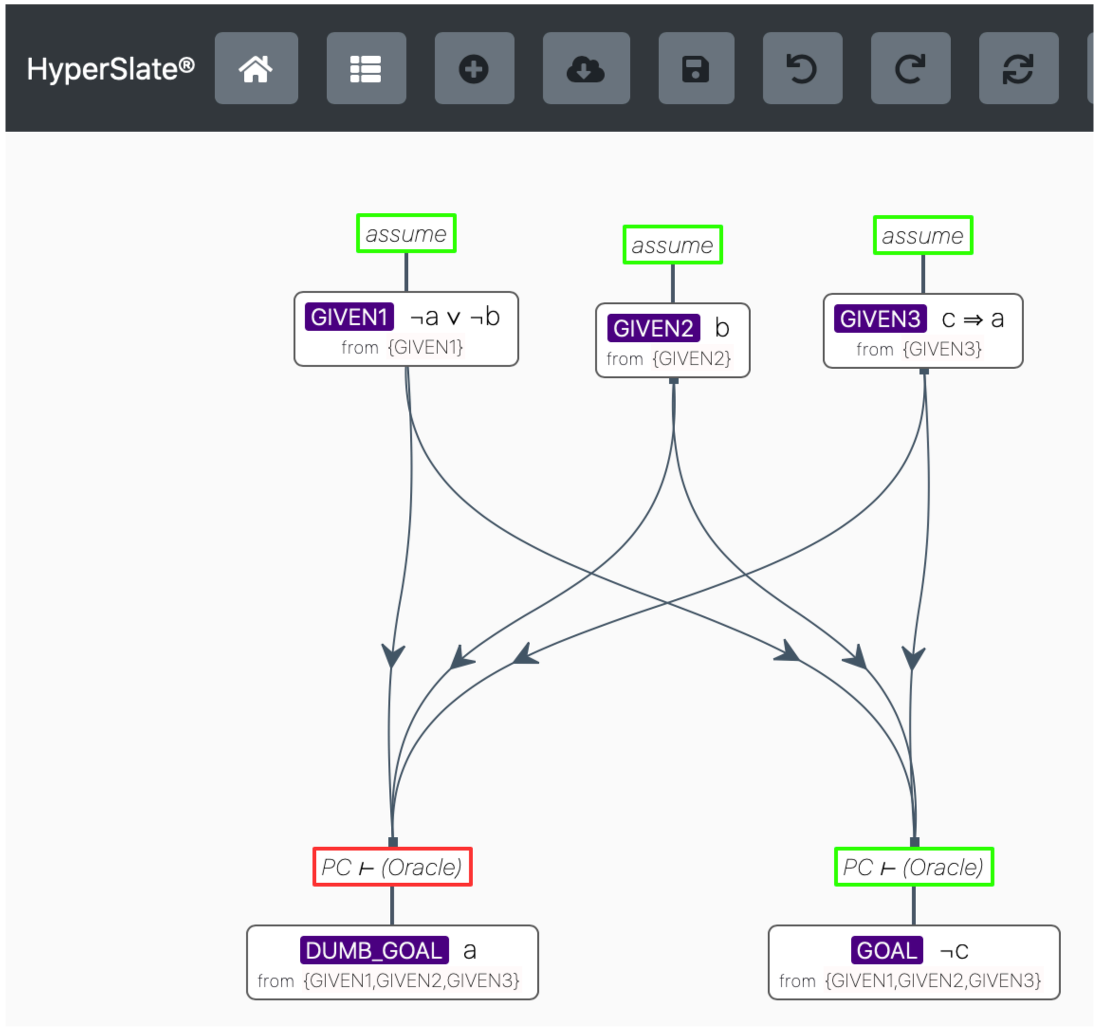
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



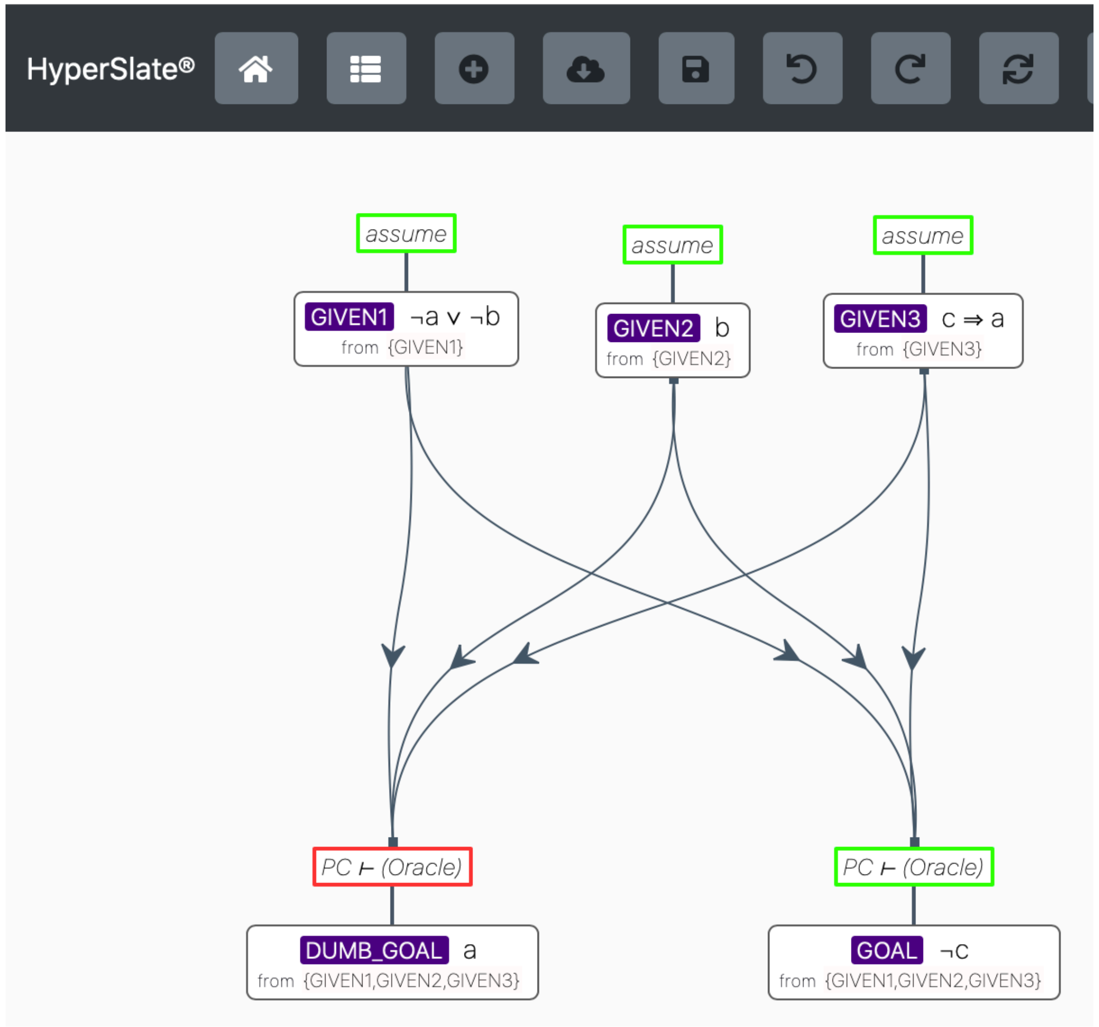
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



“NYS 3” Revisited

Given the statements

$\neg\neg c$

$c \rightarrow a$

$\neg a \lor b$

$b \rightarrow d$

$\neg(d \lor e)$

which one of the following statements are provable?

$\neg c$

e

h

$\neg a$

all of the above

“NYS 3” Revisited

Given the statements

$\neg\neg c$

$c \rightarrow a$

$\neg a \vee b$

$b \rightarrow d$

$\neg(d \vee e)$

which one of the following statements are provable?

$\neg c$

e

h

$\neg a$

all of the above

“NYS 3” Revisited

Given the statements

$\neg\neg c$

$c \rightarrow a$

$\neg a \vee b$

$b \rightarrow d$

$\neg(d \vee e)$

Show in HyperSlate® that each of the first four options can be proved using the PC entailment oracle.

which one of the following statements are provable?

$\neg c$

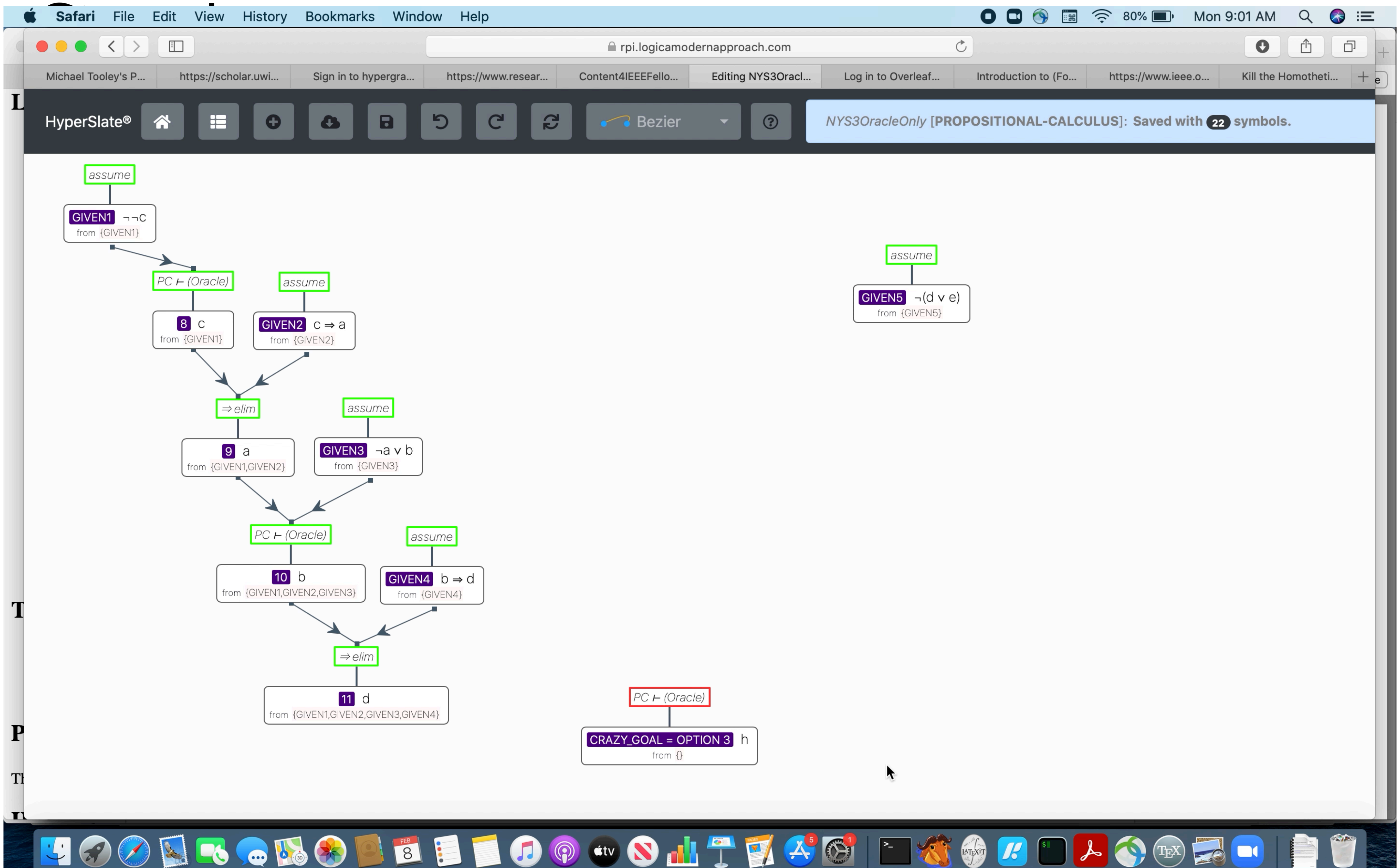
e

h

$\neg a$

all of the above

“NYS 3” Revisited



all of the above

Det er en ære å lære formell logikk!