

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

IFLAI [Intro to (Formal) Logic (and AI)]
2/8/2021



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
2/8/2021



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 Philosophy

Lally School of Management & Technology

Rensselaer Polytechnic Institute (RPI)

Troy, New York 12180 USA

Questions about last time ...?

Intro to Logic

2/8/2021



Logistics ...

- Distribution to students eg in other countries started as planned (see syllabus) last week after class.

- Distribution to students eg in other countries started as planned (see syllabus) last week after class.
- Bookstore is good to go for in-person pickup today, Feb 8, as in syllabus.

- Distribution to students eg in other countries started as planned (see syllabus) last week after class.
- Bookstore is good to go for in-person pickup today, Feb 8, as in syllabus.
- “Students with processed orders can pick them up at the back of the store.”

- Distribution to students eg in other countries started as planned (see syllabus) last week after class.
- Bookstore is good to go for in-person pickup today, Feb 8, as in syllabus.
- “Students with processed orders can pick them up at the back of the store.”
- “Students waiting until we open can go to the front registers.”

The Starting Code to Purchase in Bookstore

M

Your code for starting the registration process is:

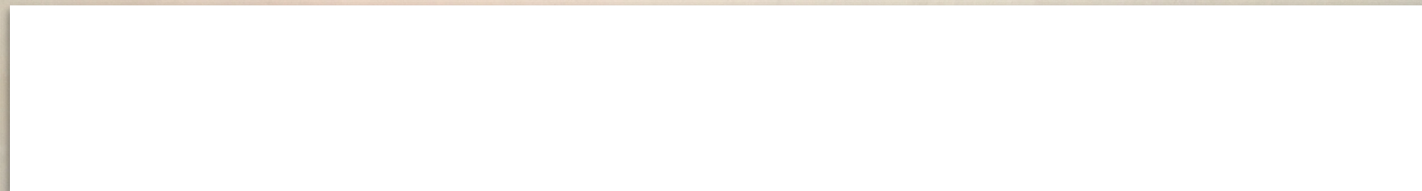
To access HyperGrader, HyperSlate, the license agreement,
and to obtain the textbook LAMA-BDLA, go to::

<https://rpi.logicamodernapproach.com>

The Starting Code to Purchase in Bookstore

M

Your code for starting the registration process is:

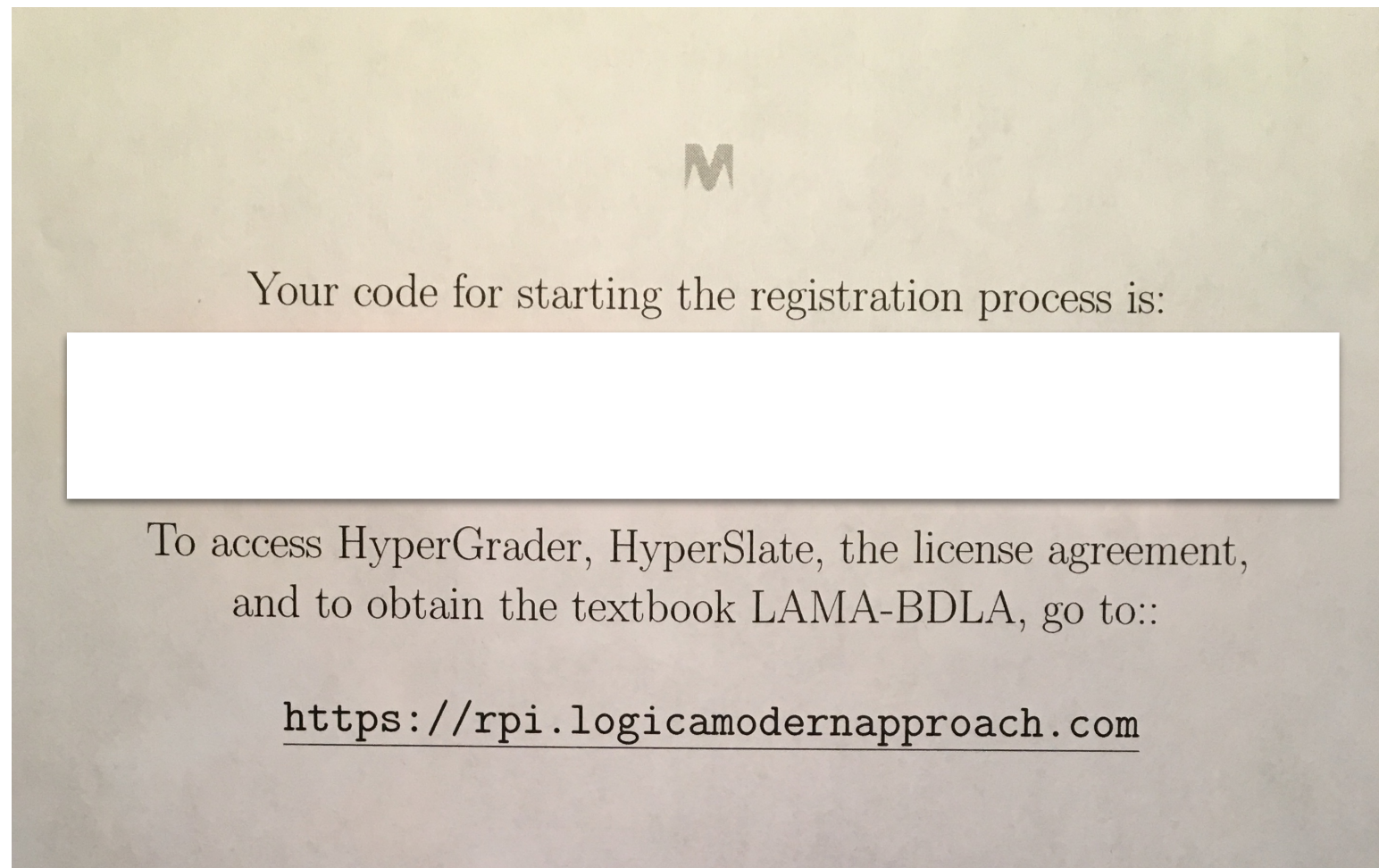


To access HyperGrader, HyperSlate, the license agreement,
and to obtain the textbook LAMA-BDLA, go to::

<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!

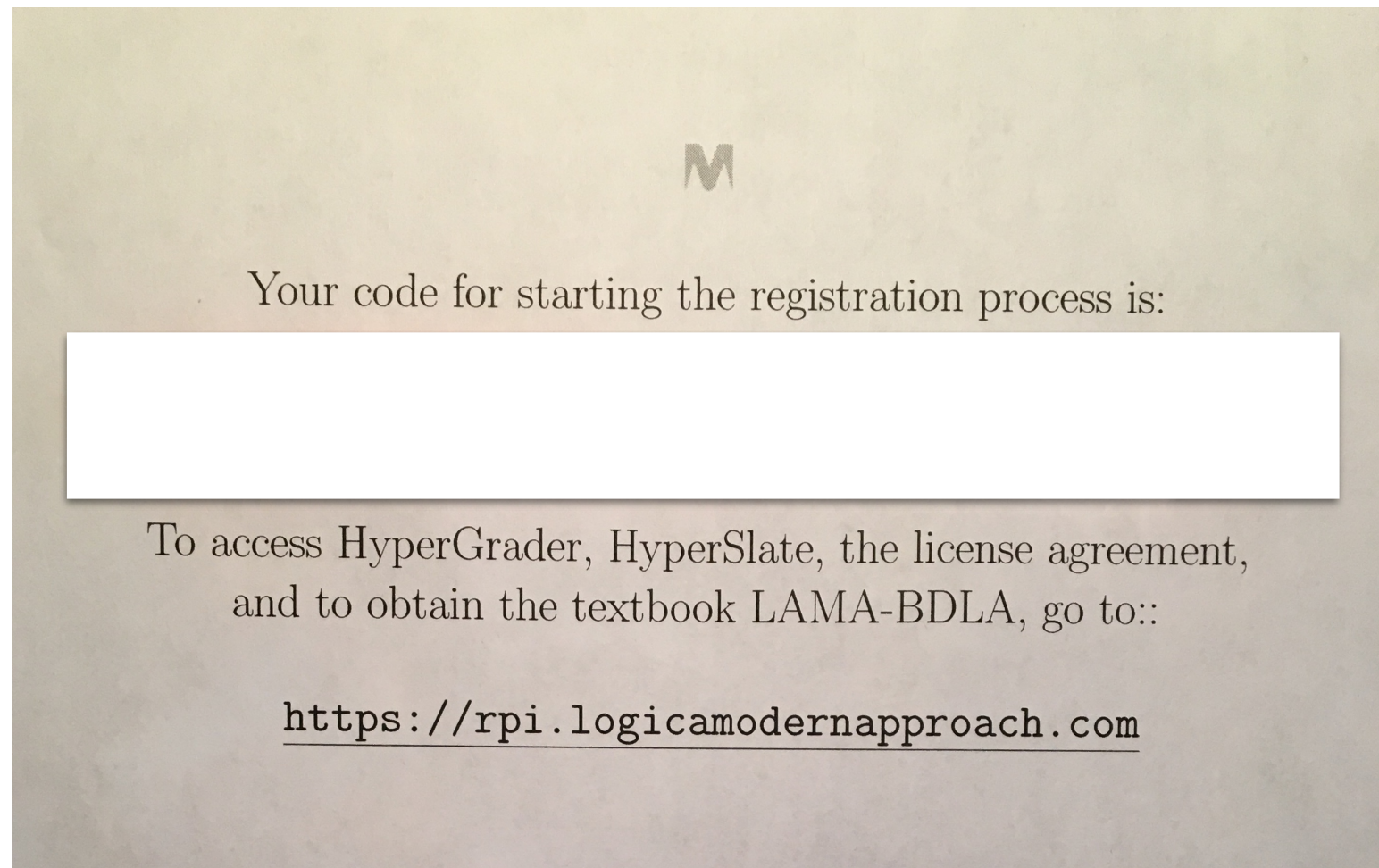
The Starting Code to Purchase in Bookstore



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

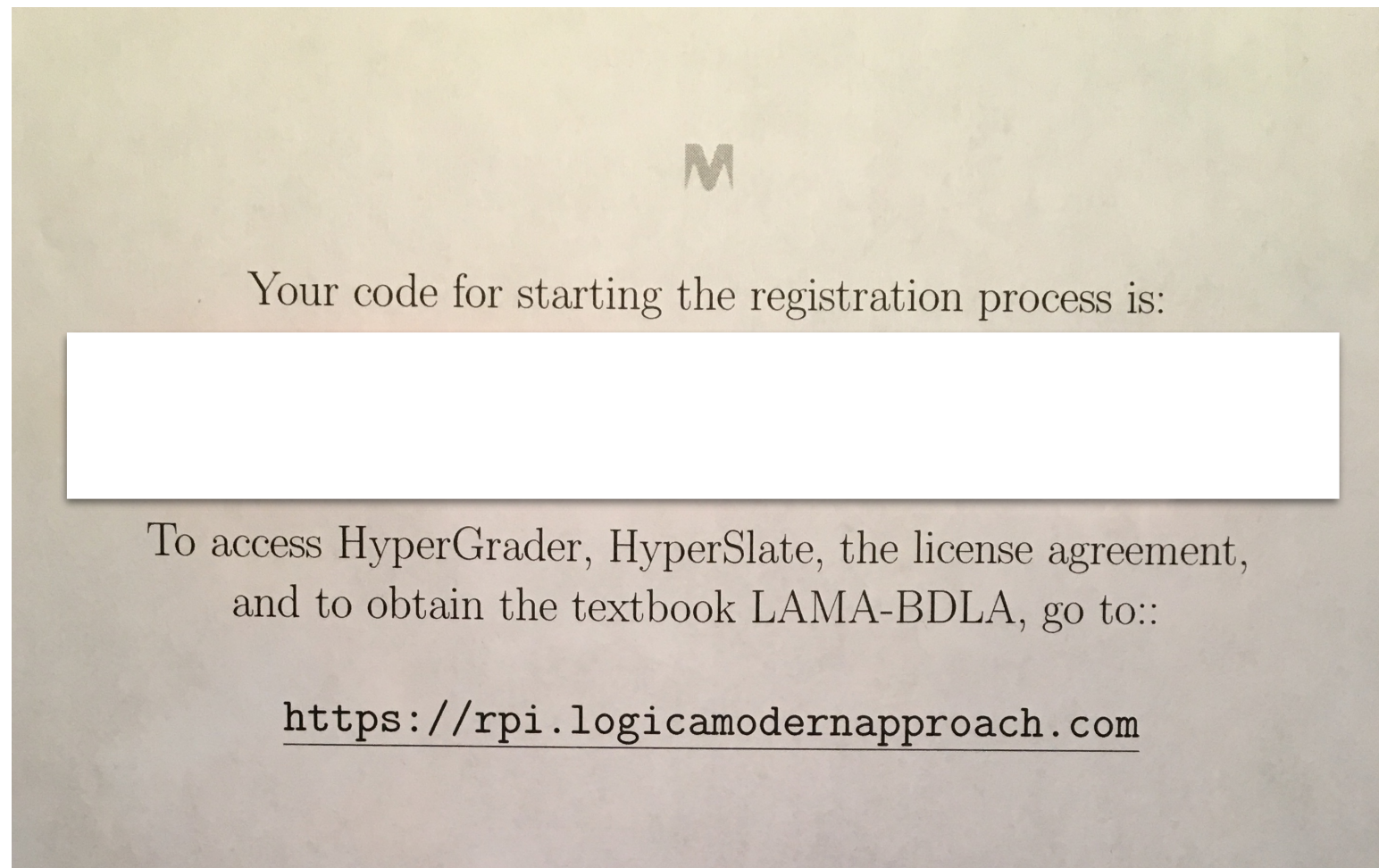


Once seal broken on envelope, no return. Remember from first class, any reservations, opt for “Stanford” paradigm, with its software instead of LAMA® paradigm!

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



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

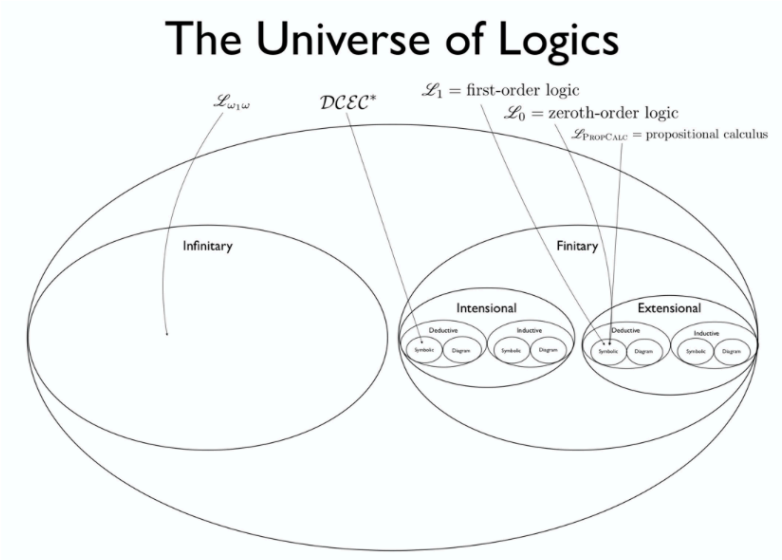
[Selmer Bringsjord](#)

Table of Contents

- [Terminology & General Orientation](#)
- [Texts/Readings](#)
- [Syllabus](#)
- [HyperSlate®](#)
- [HyperGrader®](#)
- [LAMA-BDLAHS Textbook](#)
- [Lectures](#)
- [Tutorials](#)
- [Pop Problems](#)
- [Homeworks](#)
- [Tests](#)

A fully online course, thanks to singular AI technology.

with [Naveen Sundar G.](#)
^ KB Foush   ^ 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
$P, P_1, P_2, Q, Q_1, \dots$	Atomic Formulas	"Larry is lucky." as L_l
$\neg\phi$	Negation	"Gary isn't lucky." as $\neg L_g$
$\phi_1 \wedge \dots \wedge \phi_n$	Conjunction	"Both Larry and Carl are lucky." as $L_l \wedge L_c$
$\phi_1 \vee \dots \vee \phi_n$	Disjunction	"Either Billy is lucky or Alvin is." as $L_b \vee L_a$
$\phi \rightarrow \psi$	Conditional (Implication)	"If Ron is lucky, so is Frank." as $L_r \rightarrow L_f$
$\phi \leftrightarrow \psi$	Biconditional (Coimplication)	"Tim is lucky if and only if Kim is." as $L_t \leftrightarrow L_k$

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
$P, P_1, P_2, Q, Q_1, \dots$	Atomic Formulas	"Larry is lucky." as L_l
$\neg\phi$	Negation	"Gary isn't lucky." as $\neg L_g$
$\phi_1 \wedge \dots \wedge \phi_n$	Conjunction	"Both Larry and Carl are lucky." as $L_l \wedge L_c$
$\phi_1 \vee \dots \vee \phi_n$	Disjunction	"Either Billy is lucky or Alvin is." as $L_b \vee L_a$
$\phi \rightarrow \psi$	Conditional (Implication)	"If Ron is lucky, so is Frank." as $L_r \rightarrow L_f$
$\phi \leftrightarrow \psi$	Biconditional (Coimplication)	"Tim is lucky if and only if Kim is." as $L_t \leftrightarrow L_k$

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)

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

The Formal Language

(presented as formal grammar)

$Formula \Rightarrow AtomicFormula$

$\begin{array}{|l} (Formula \text{ Connective } Formula) \\ \neg Formula \end{array}$

$AtomicFormula \Rightarrow P_1 \mid P_2 \mid P_3 \mid \dots$

$Connective \Rightarrow \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

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

As S-expressions

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

$$\textit{AtomicFormula} \Rightarrow \text{P}_1 \mid \text{P}_2 \mid \text{P}_3 \mid \dots$$

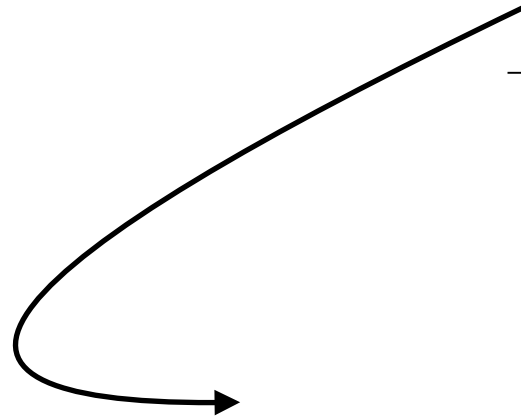
$$\textit{Connective} \Rightarrow \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$$

As S-expressions

Formula $\Rightarrow$ *AtomicFormula*
| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

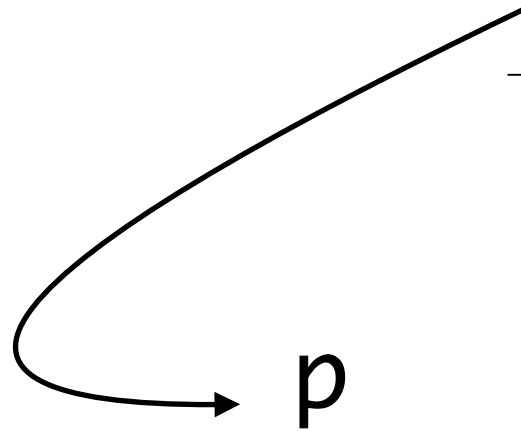
Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$



As S-expressions

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

$$\textit{AtomicFormula} \Rightarrow \text{P}_1 \mid \text{P}_2 \mid \text{P}_3 \mid \dots$$

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


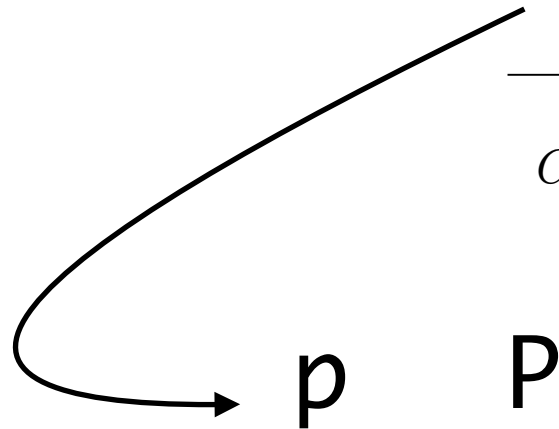
p

As S-expressions

Formula $\Rightarrow$ *AtomicFormula*
| *(Formula Connective Formula)*
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

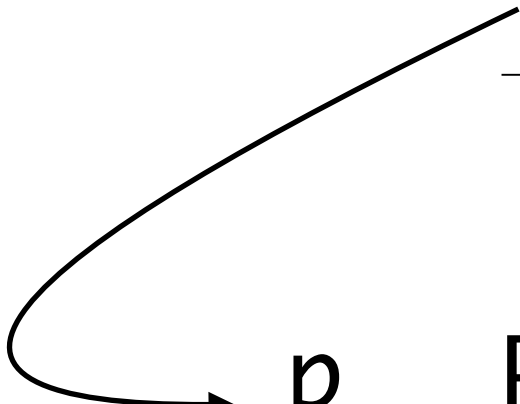


p **P**

As S-expressions

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

$$\textit{AtomicFormula} \Rightarrow P_1 \mid P_2 \mid P_3 \mid \dots$$

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


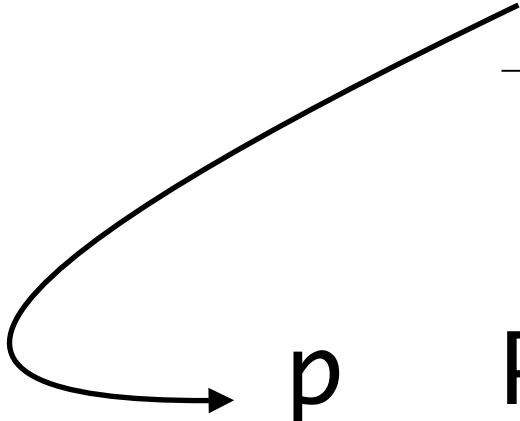
p P bradyisleaving

As S-expressions

$$\begin{array}{l} \textit{Formula} \Rightarrow \textit{AtomicFormula} \\ \quad | \quad (\textit{Formula} \textit{Connective} \textit{Formula}) \\ \quad | \quad \neg \textit{Formula} \end{array}$$

$$\textit{AtomicFormula} \Rightarrow P_1 \mid P_2 \mid P_3 \mid \dots$$

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

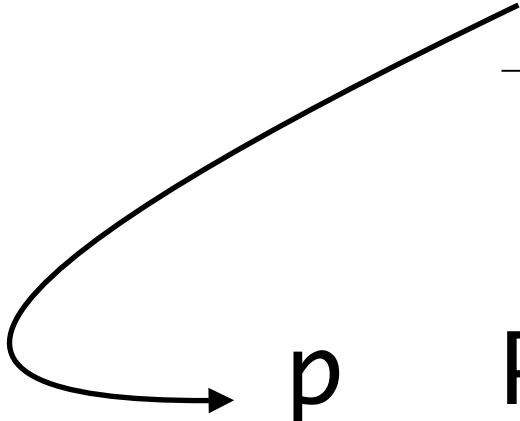

 p P bradyisleaving P26

As S-expressions

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

$$\textit{AtomicFormula} \Rightarrow P_1 \mid P_2 \mid P_3 \mid \dots$$

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

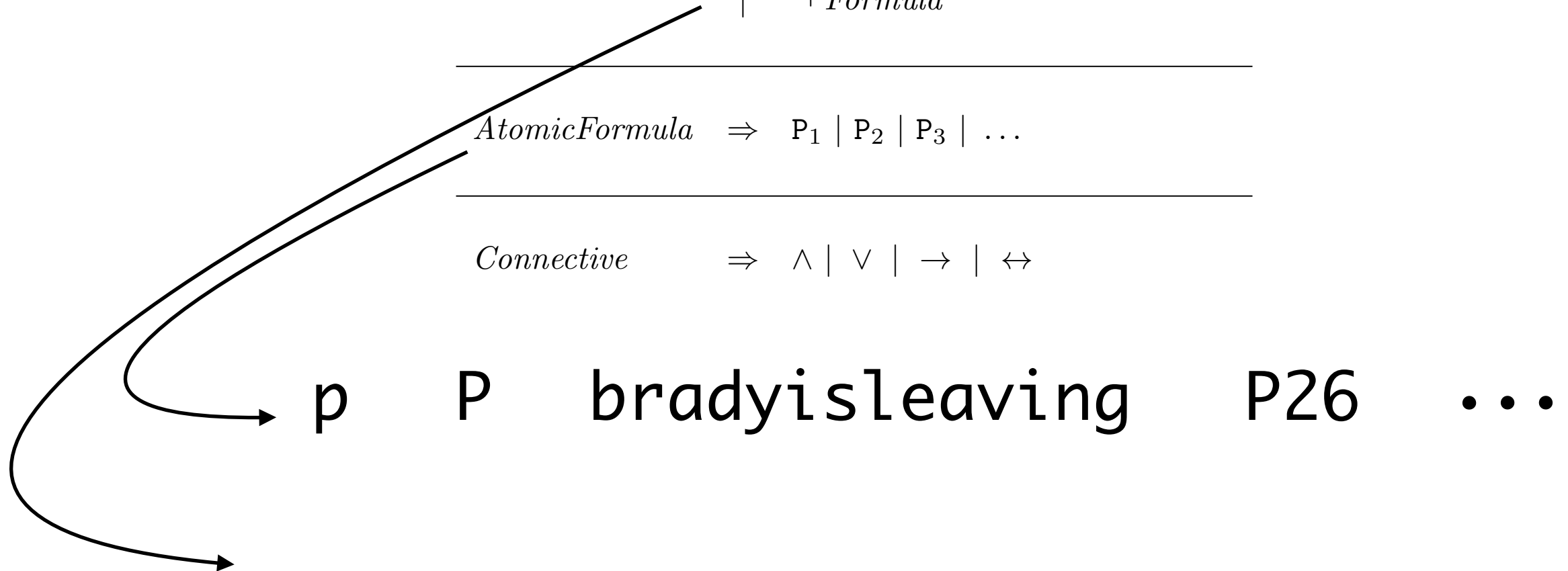

 p P bradyisleaving P26 ...

As S-expressions

Formula $\Rightarrow$ *AtomicFormula*
 $\mid$ (*Formula* *Connective* *Formula*)
 $\mid \neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$



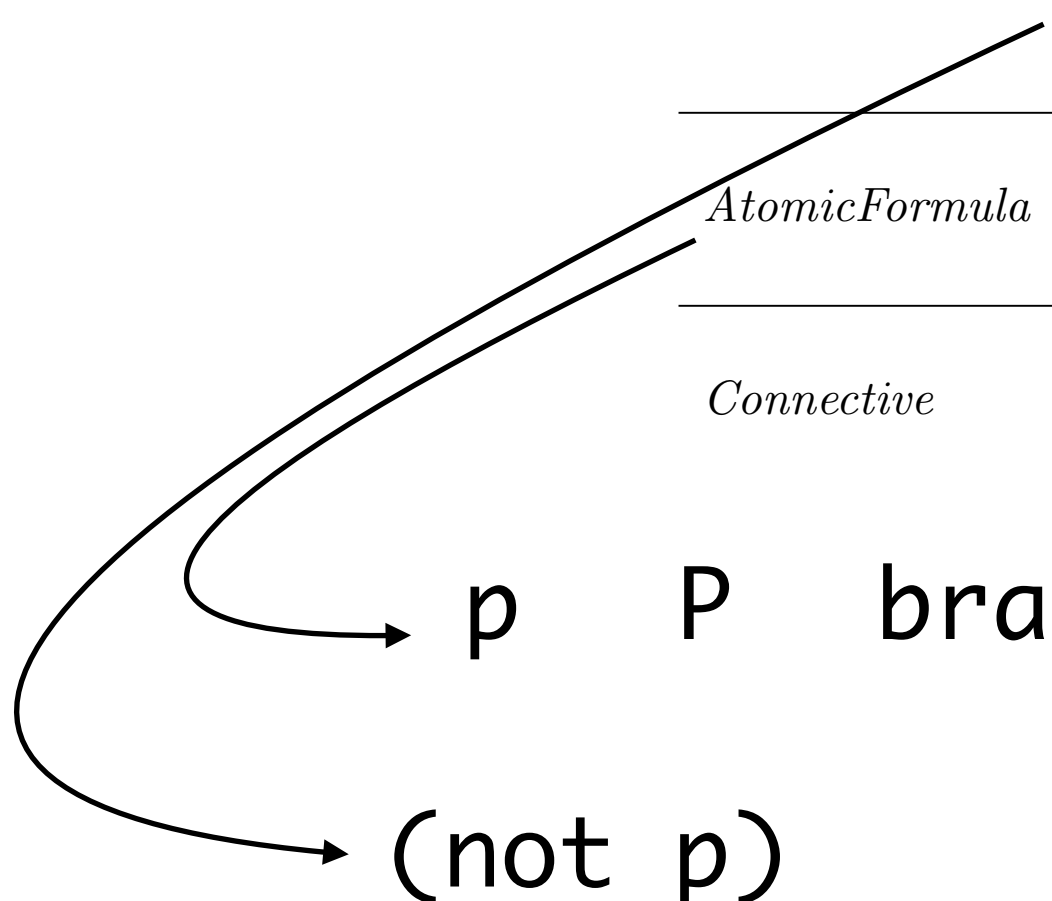
As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$


 p P bradyisleaving P26 ...
 (not p)

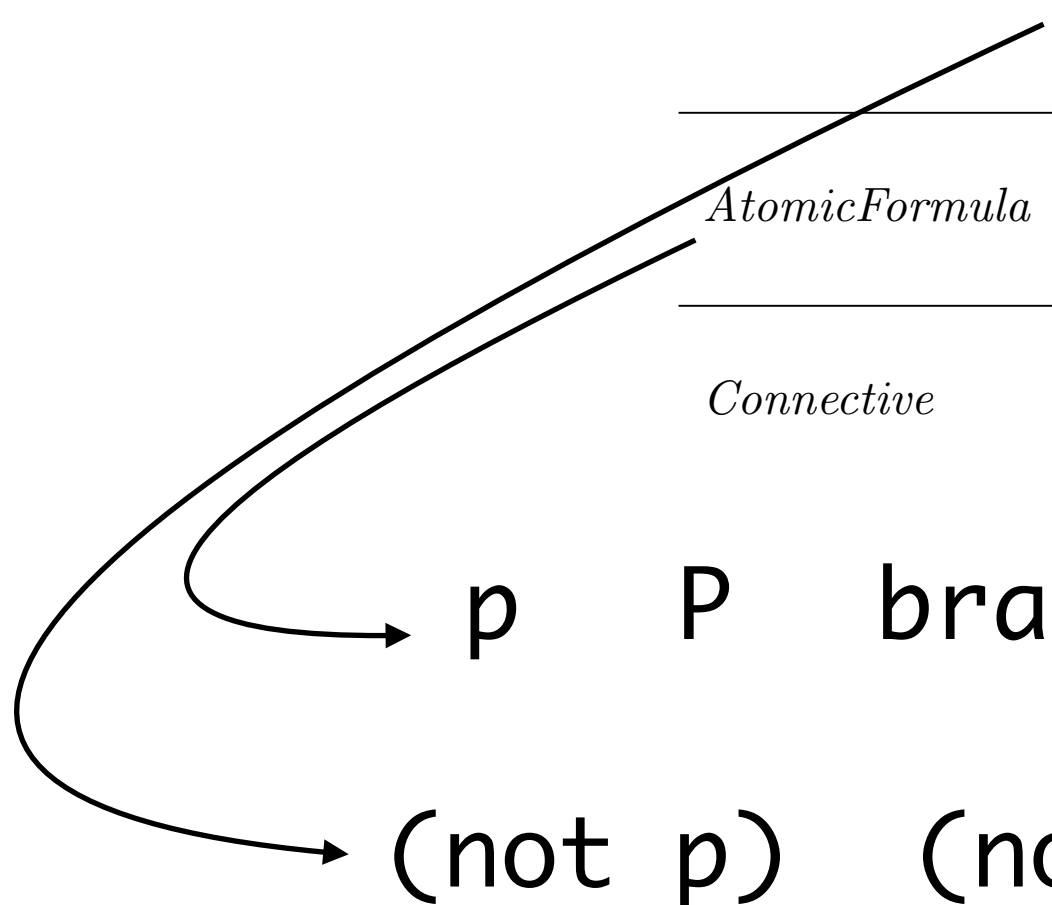
As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$


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

As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

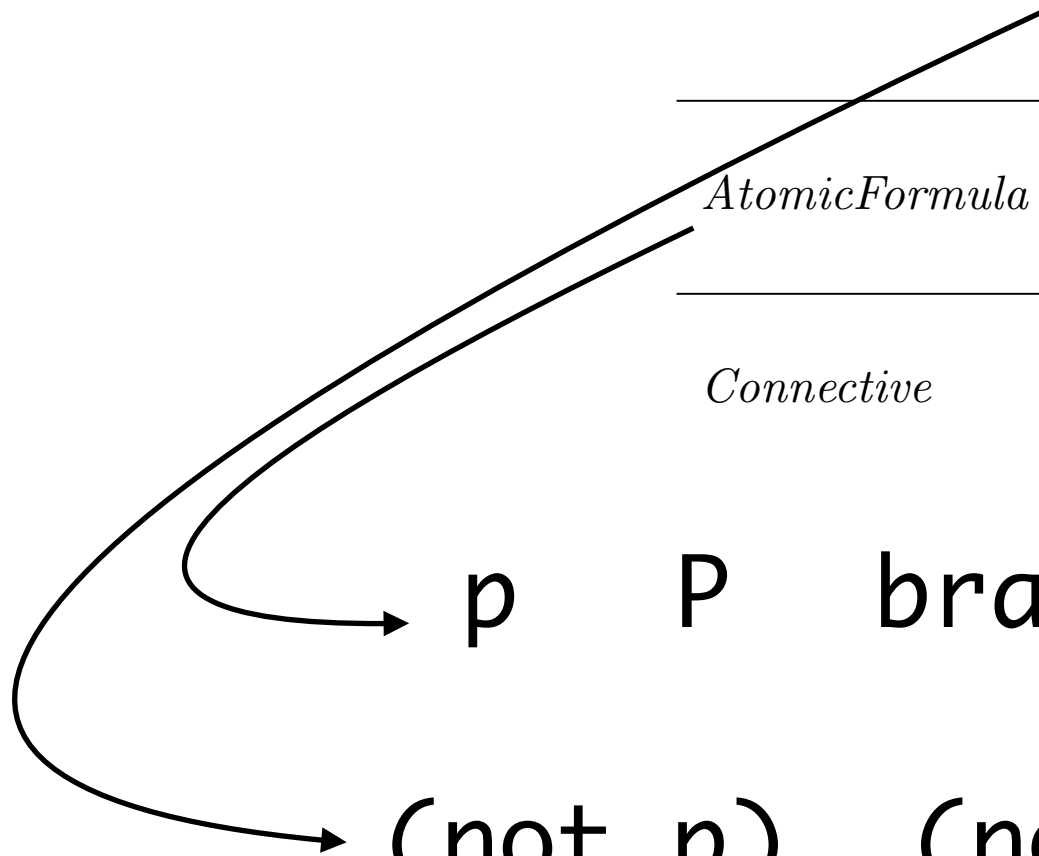
| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

p P bradyisleaving P26 ...

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



As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ $P_1 \mid P_2 \mid P_3 \mid \dots$

Connective $\Rightarrow$ $\wedge \mid \vee \mid \rightarrow \mid \leftrightarrow$

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

As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

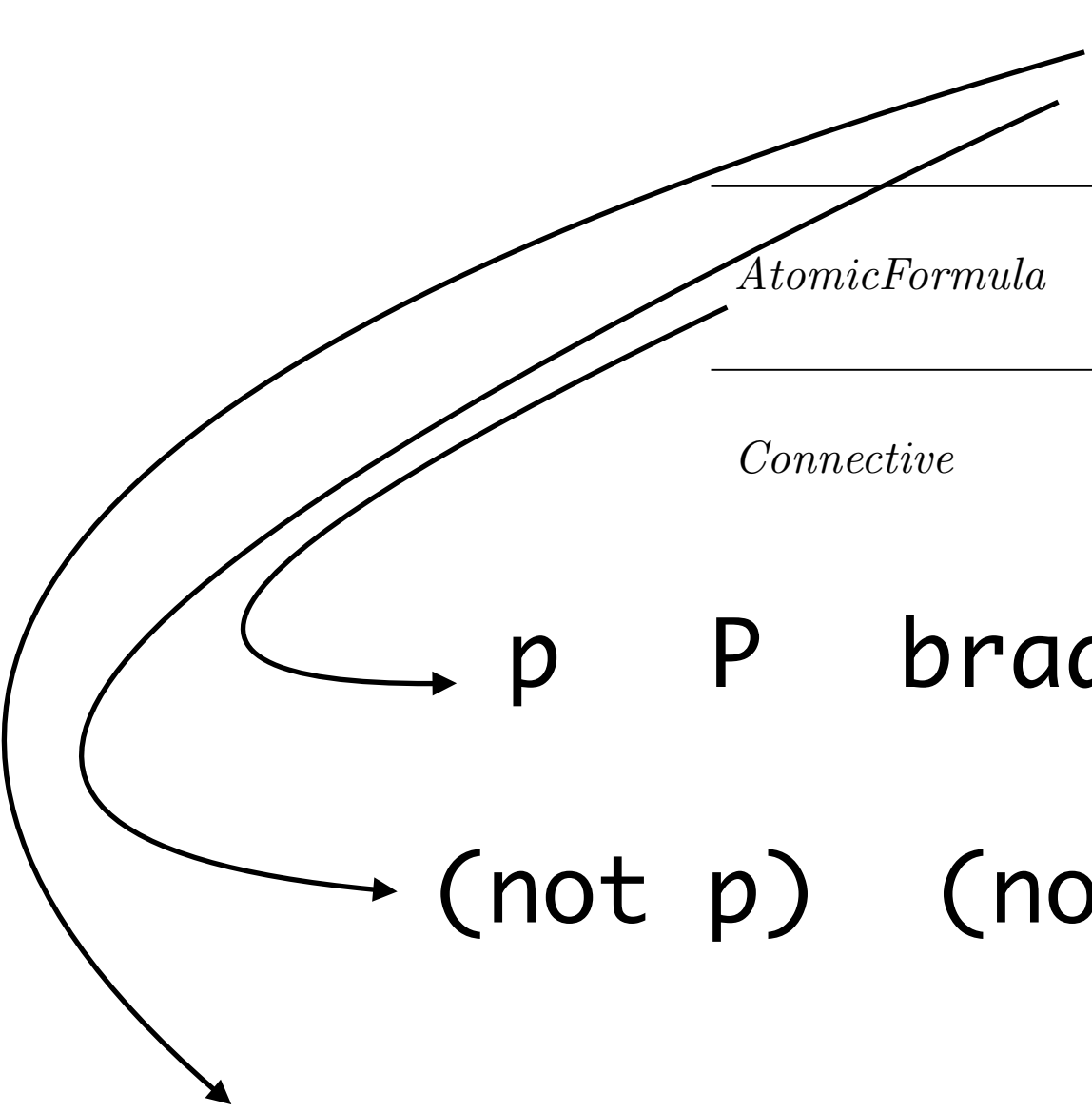
| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

AtomicFormula $\Rightarrow$ P_1 | P_2 | P_3 | ...

Connective $\Rightarrow$ $\wedge$ | $\vee$ | $\rightarrow$ | $\leftrightarrow$

p P bradyisleaving P26 ...

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



As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

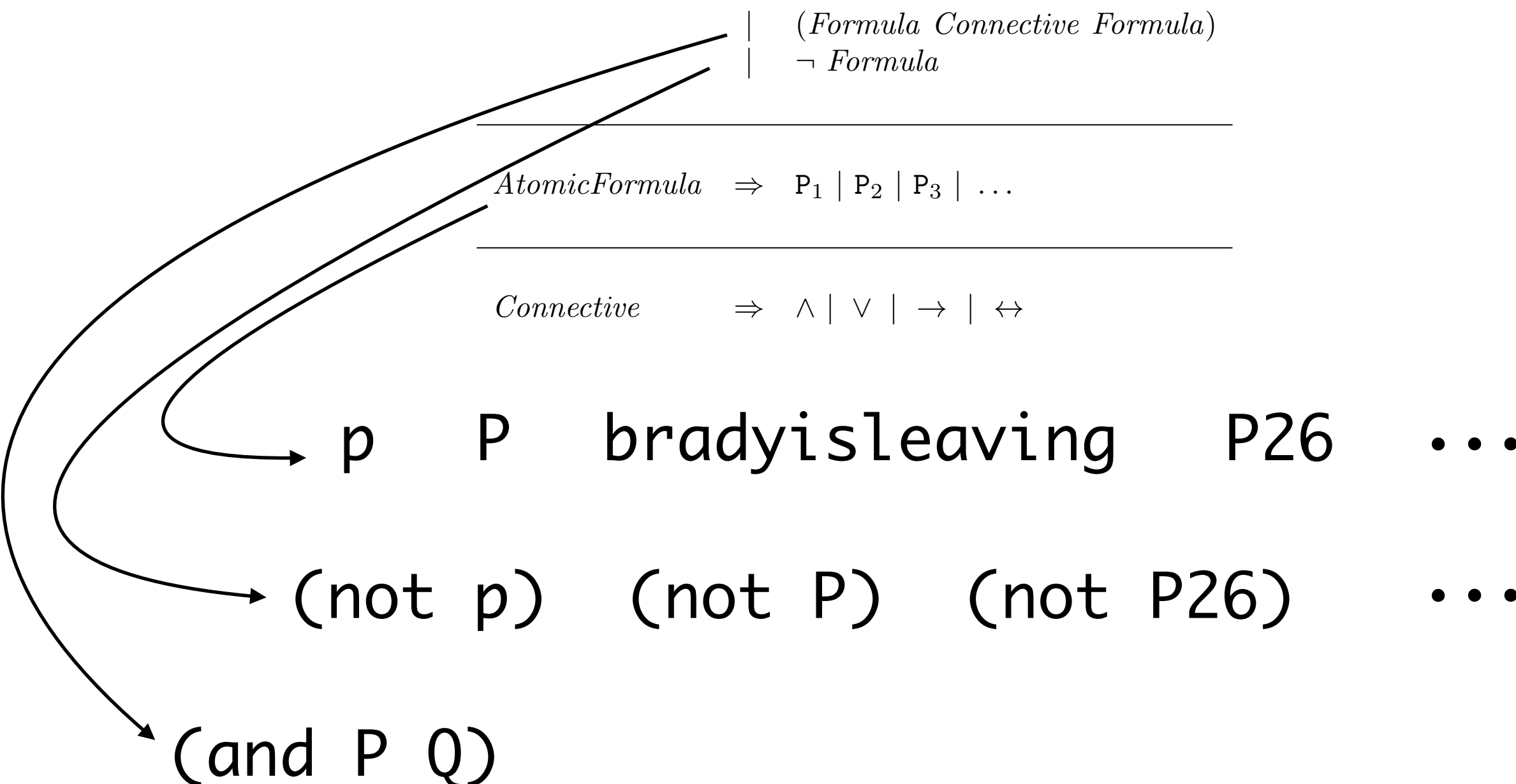
AtomicFormula $\Rightarrow$ P_1 | P_2 | P_3 | ...

Connective $\Rightarrow$ $\wedge$ | $\vee$ | $\rightarrow$ | $\leftrightarrow$

p P bradyisleaving P26 ...

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

(and P Q)



As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

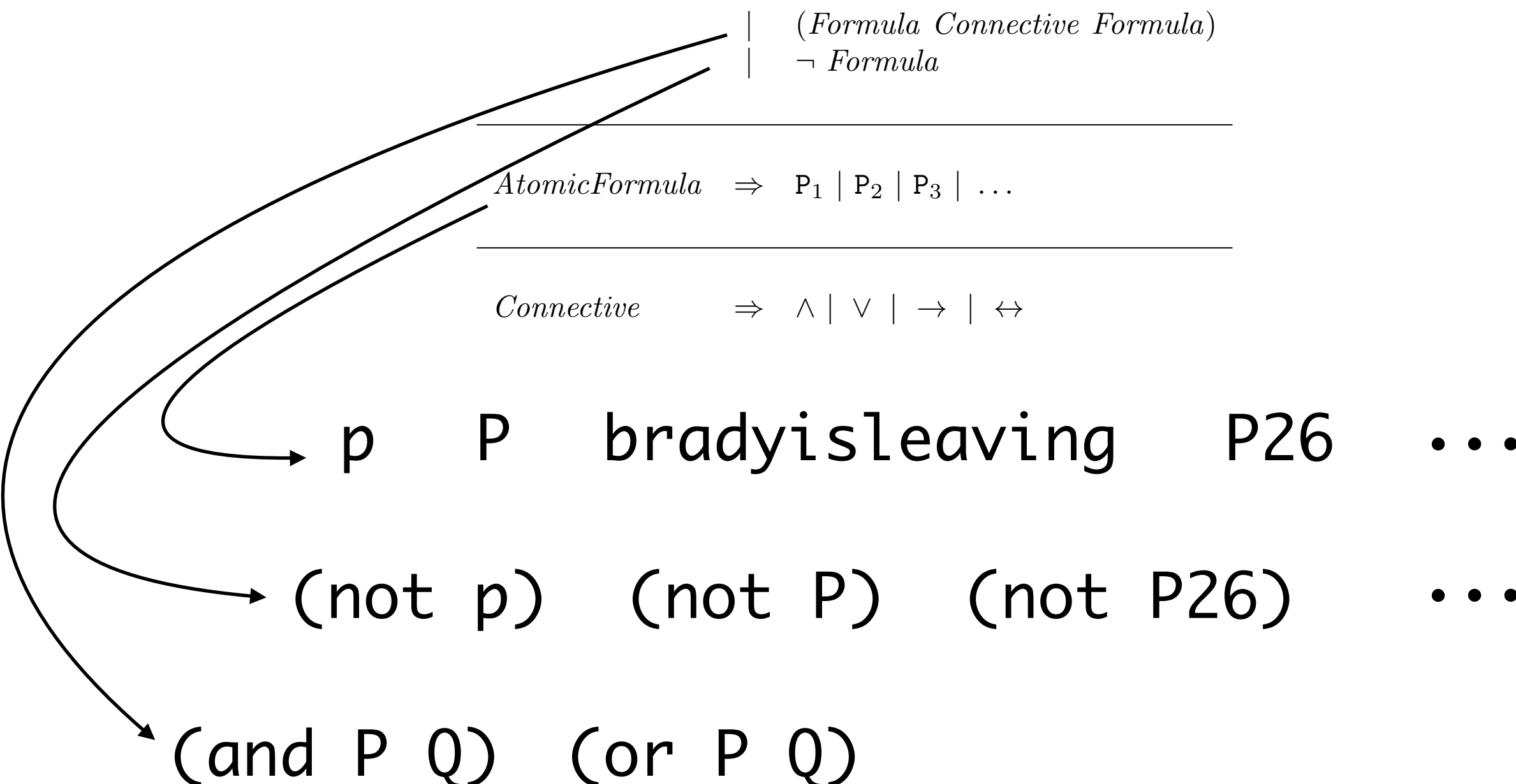
AtomicFormula $\Rightarrow$ P₁ | P₂ | P₃ | ...

Connective $\Rightarrow$ $\wedge$ | $\vee$ | $\rightarrow$ | $\leftrightarrow$

p P bradyisleaving P26 ...

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

(and P Q) (or P Q)



As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

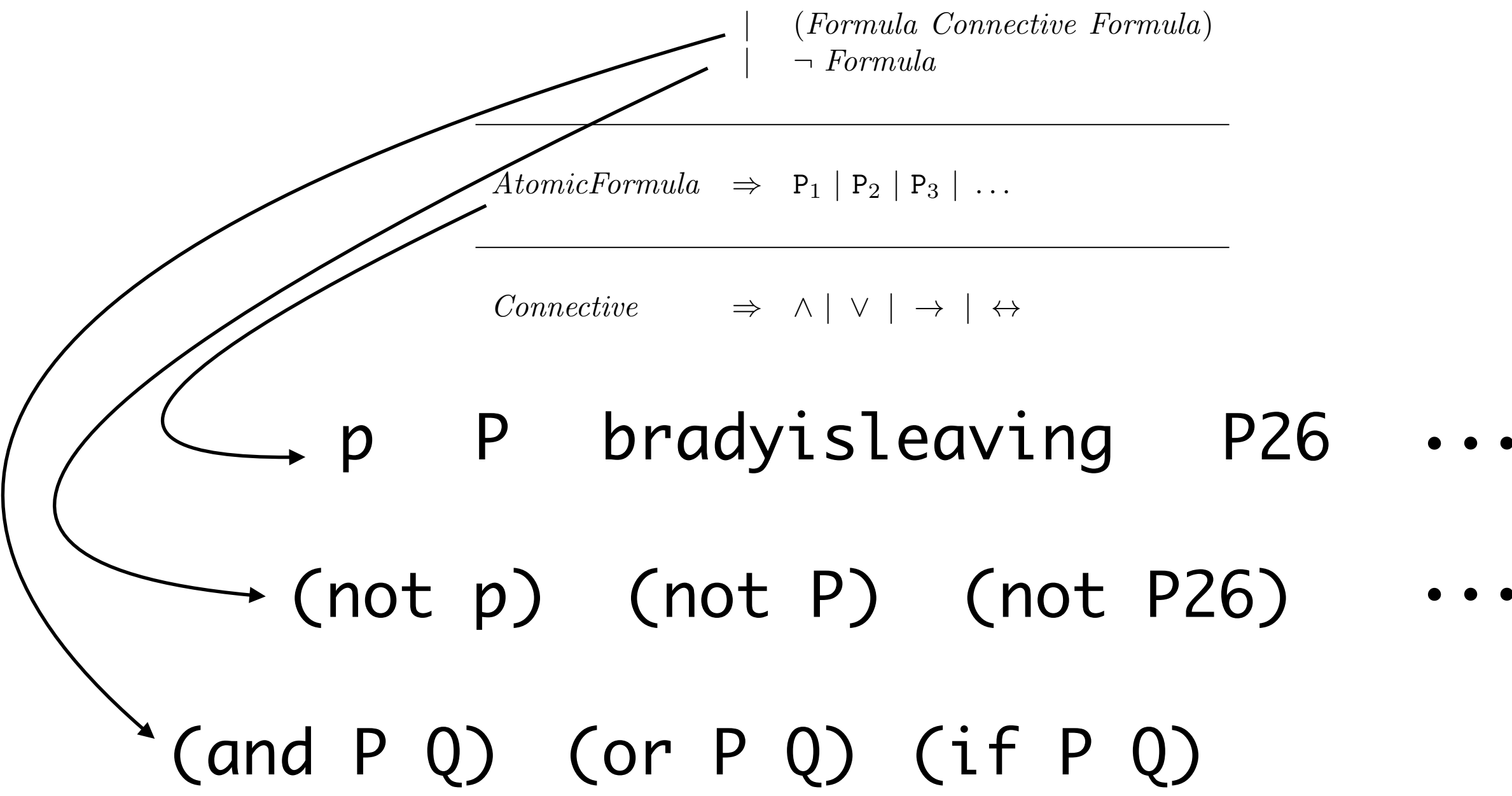
AtomicFormula $\Rightarrow$ P_1 | P_2 | P_3 | ...

Connective $\Rightarrow$ $\wedge$ | $\vee$ | $\rightarrow$ | $\leftrightarrow$

p P bradyisleaving P26 ...

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

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



As S-expressions

Formula $\Rightarrow$ *AtomicFormula*

| (*Formula* *Connective* *Formula*)
| $\neg$ *Formula*

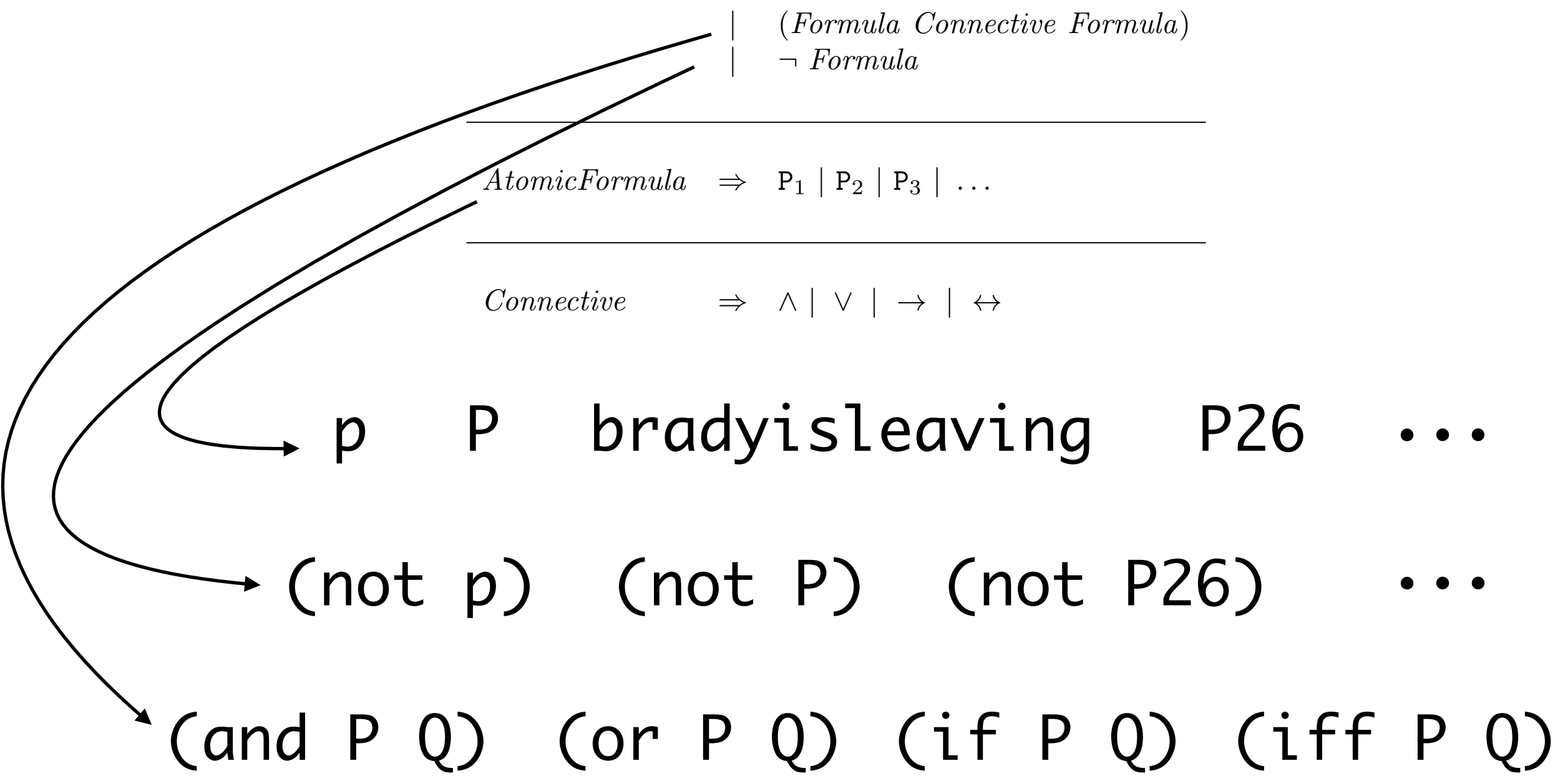
AtomicFormula $\Rightarrow$ P_1 | P_2 | P_3 | ...

Connective $\Rightarrow$ $\wedge$ | $\vee$ | $\rightarrow$ | $\leftrightarrow$

p P bradyisleaving 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)

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

$$\textit{AtomicFormula} \Rightarrow (\textit{Predicate} \textit{Term}_1 \dots \textit{Term}_k)$$

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

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

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

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

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

$$\textit{AtomicFormula} \Rightarrow (\textit{Predicate} \textit{Term}_1 \dots \textit{Term}_k)$$

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

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

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

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

“NYS I” Revisited

Given the statements

$$\neg a \vee \neg b$$

$$b$$

$$c \rightarrow a$$

which one of the following statements must also be true?

$$c$$

$$\neg b$$

$$\neg c$$

$$h$$

$$a$$

none of the above

“NYS I” Revisited

Given the statements

$$\neg a \vee \neg b$$

b

$$c \rightarrow a$$

which one of the following statements must also be true?

c

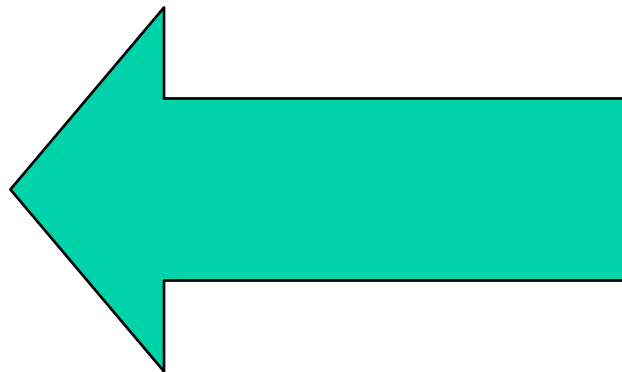
$\neg b$

$\neg c$

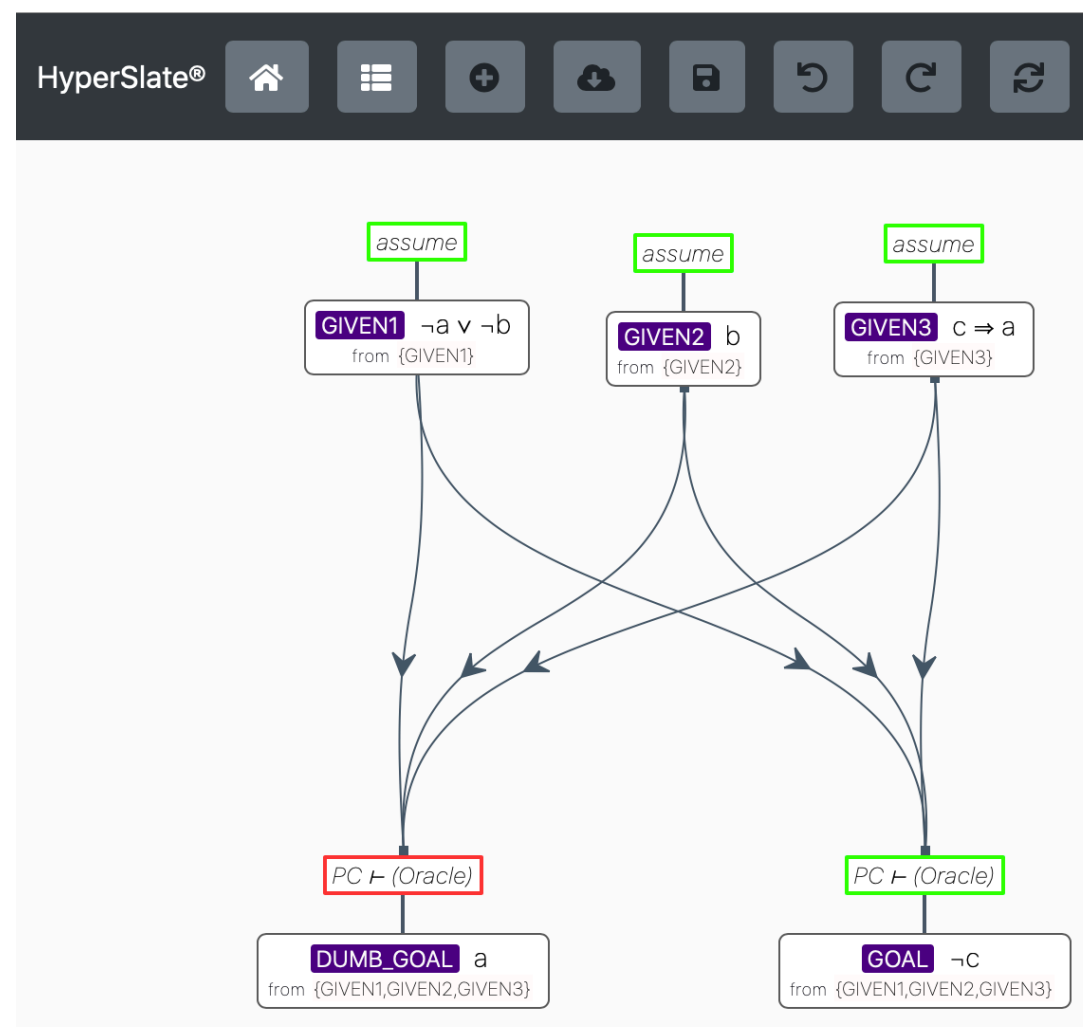
h

a

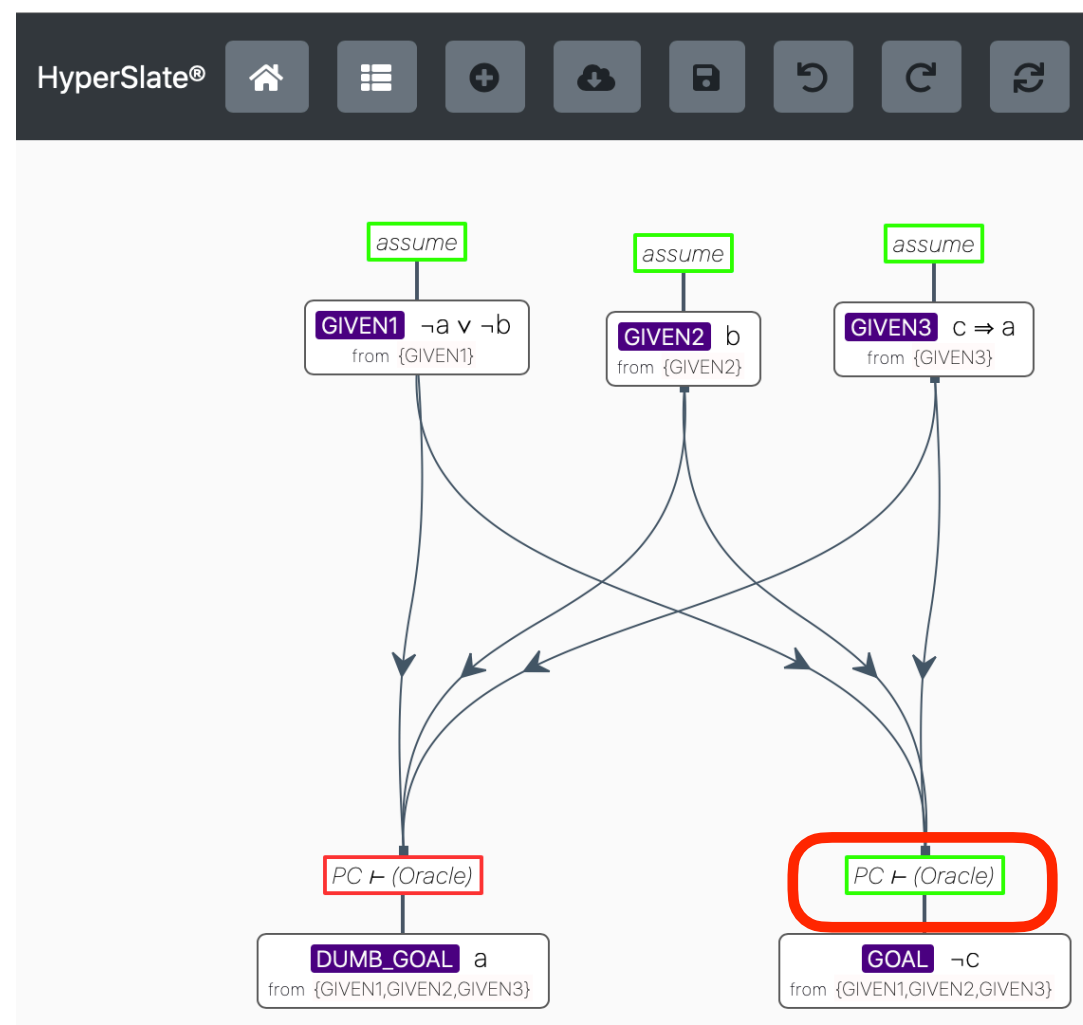
none of the above



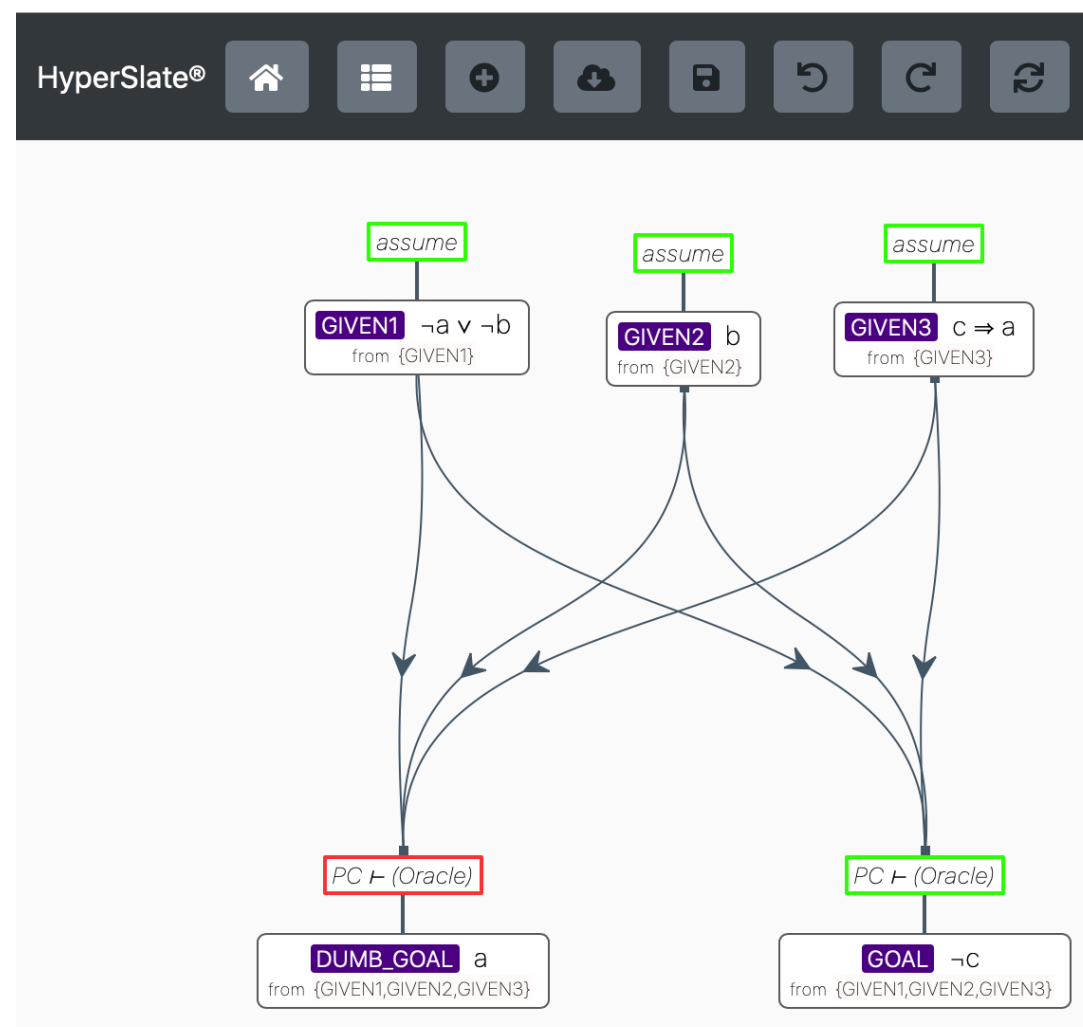
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



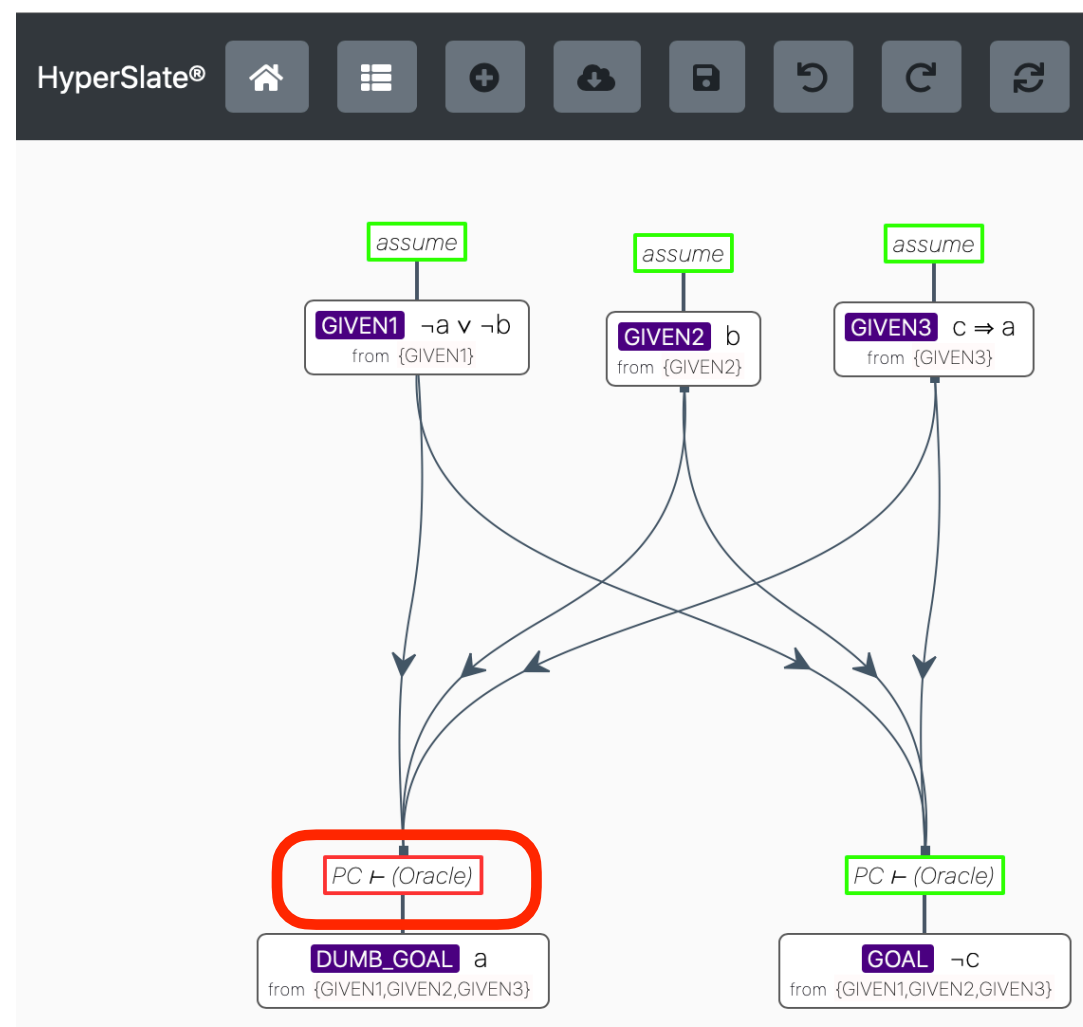
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



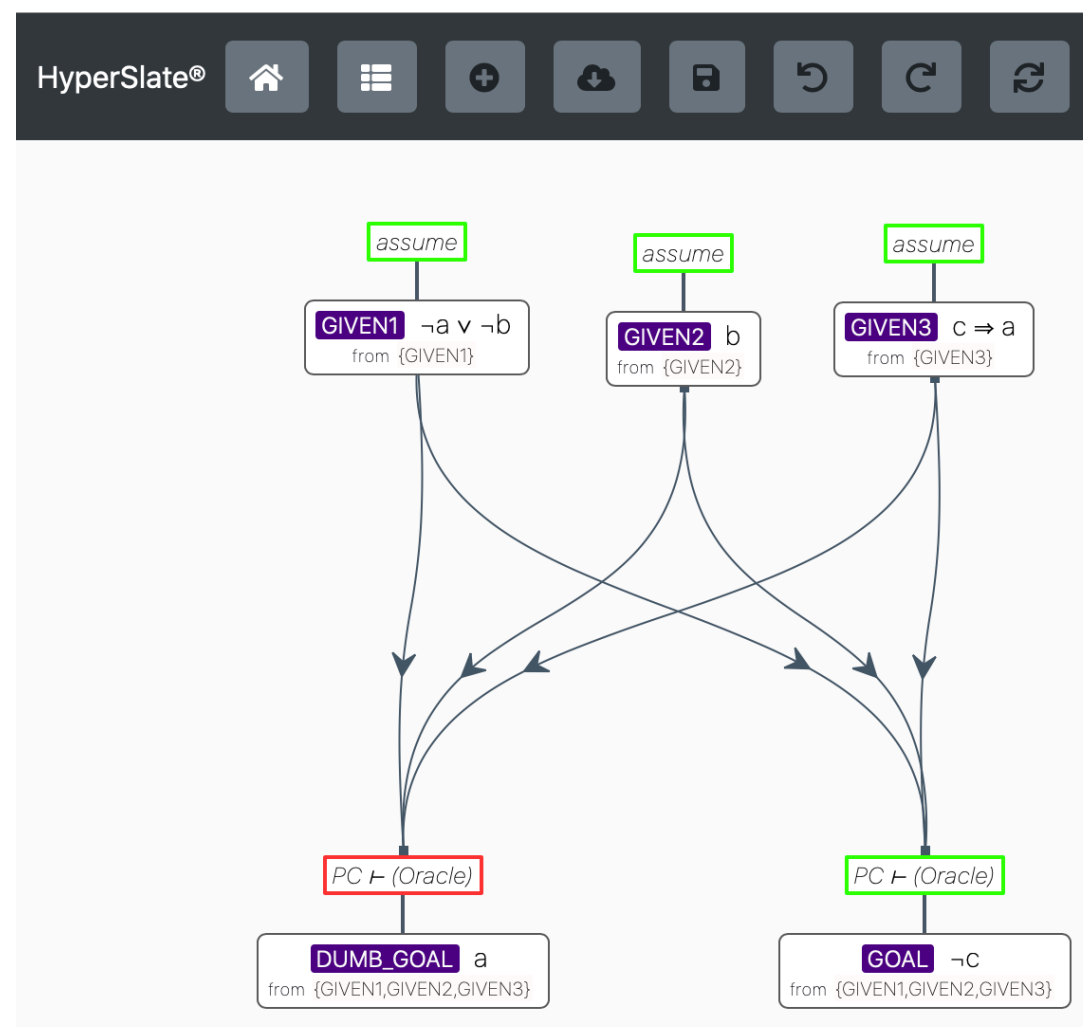
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



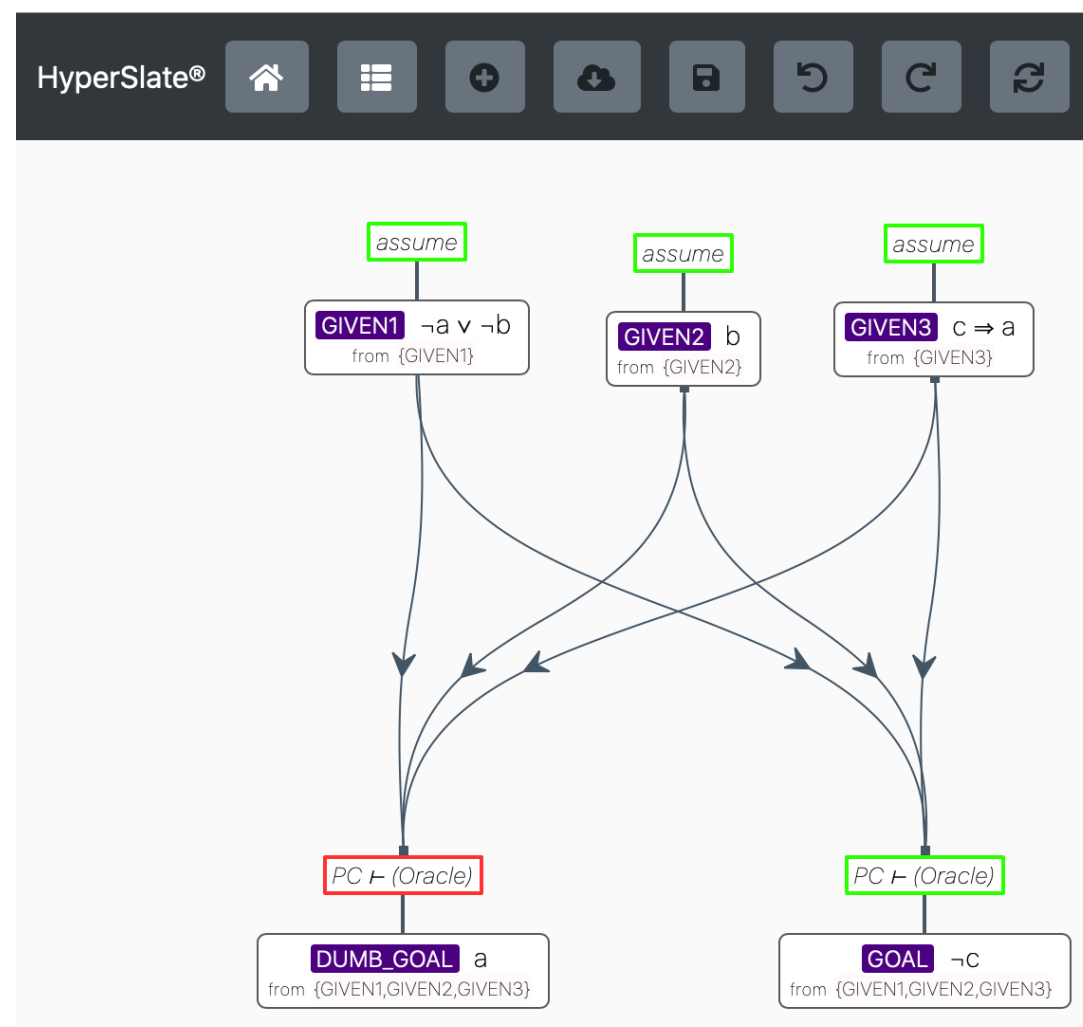
Our First Rule of Inference (= Inference Schema): PC (Entailment) Oracle



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

$$\neg\neg c$$

$$c \rightarrow a$$

$$\neg a \vee b$$

$$b \rightarrow d$$

$$\neg(d \vee e)$$

which one of the following statements must also be true?

$$\neg c$$

$$e$$

$$h$$

$$\neg a$$

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 must also be true?

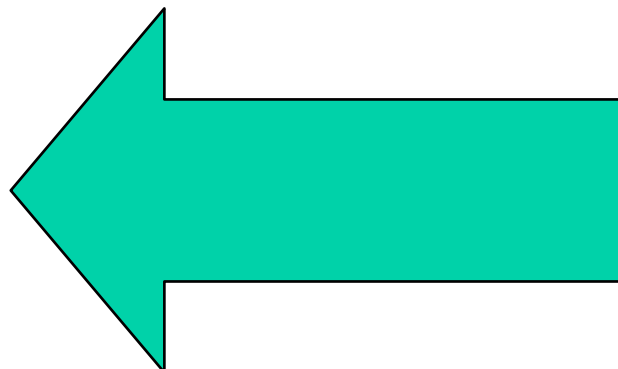
$$\neg c$$

$$e$$

$$h$$

$$\neg a$$

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 must also be true?

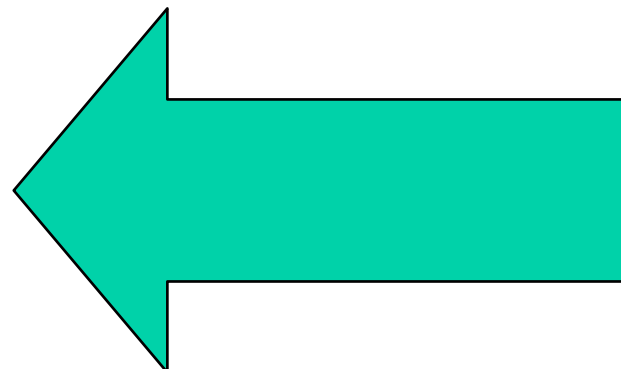
$\neg c$

e

h

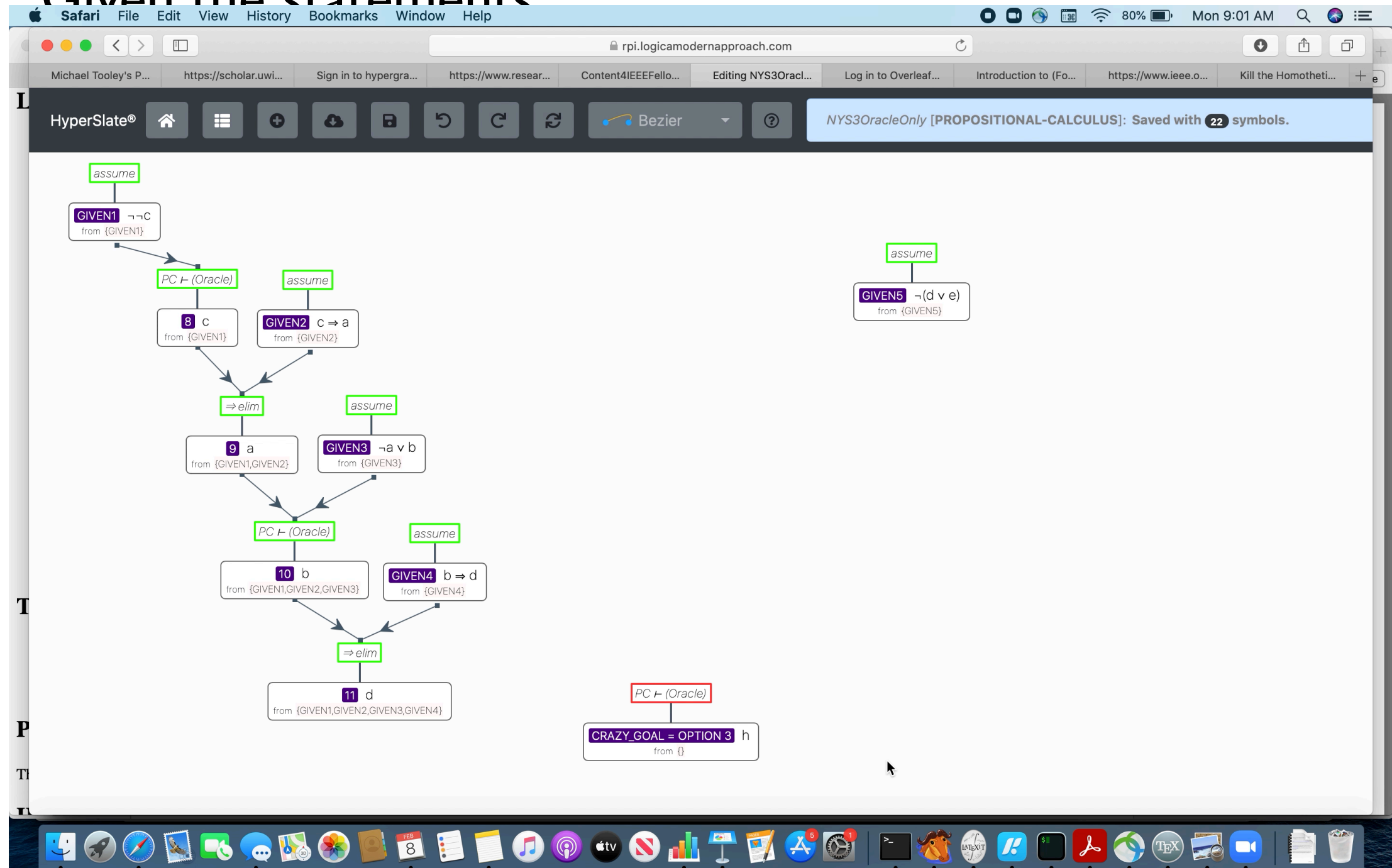
$\neg a$

all of the above



“NYS 3” Revisited

Given the statements



all of the above

Det er en ære å lære formell logikk!