On Quantificational Modal Logic

(S5-centric)

Selmer Bringsjord

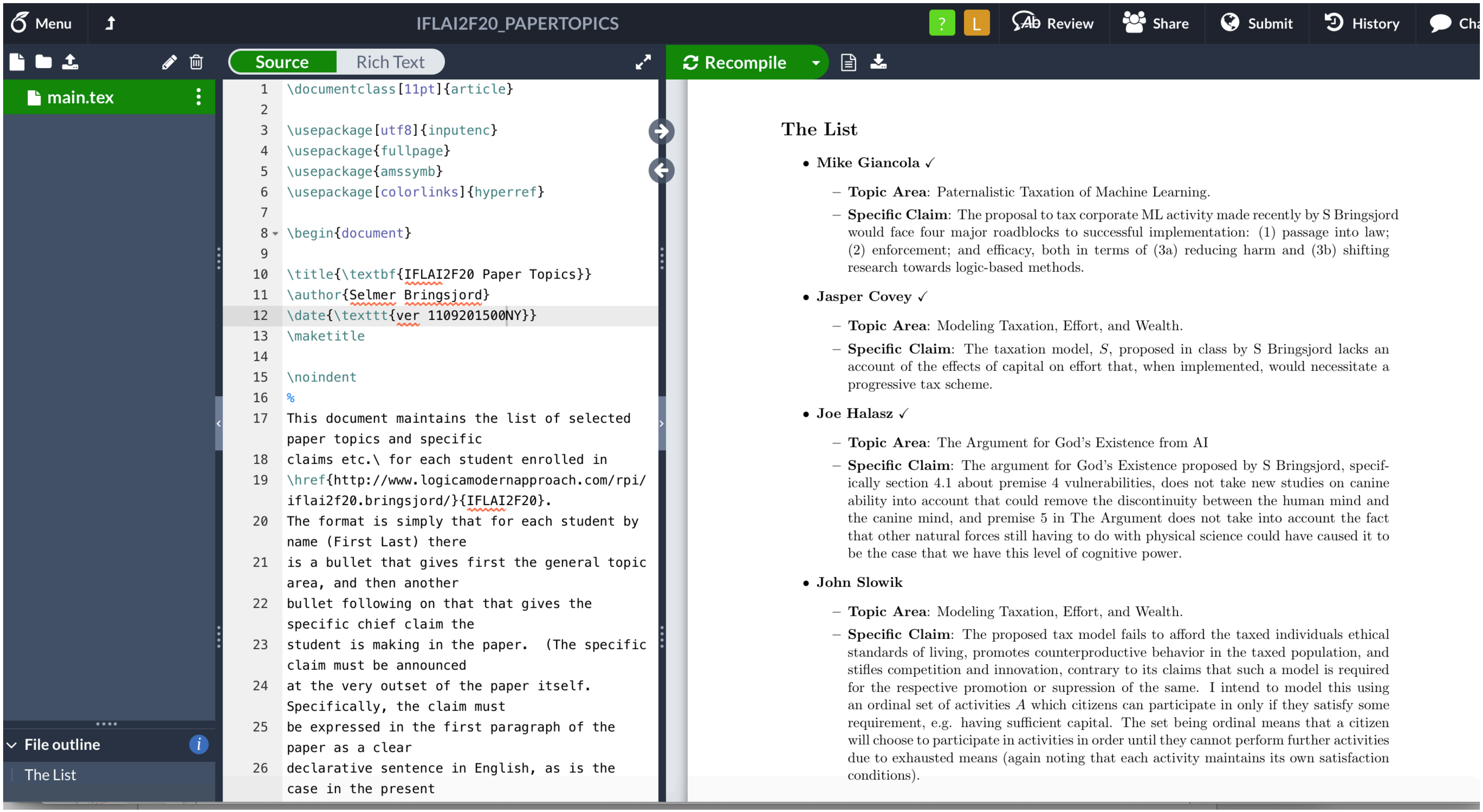
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Intro to Logic
4/12/2020
ver 1112202100NY

RAIR
Rensselaer AI and Reasoning Lab

Logistics …

Status? Some discussion …



Recall: Schedule Switcheroo

Recall: Schedule Switcheroo

- **Nov 5**: *Pure General Logic Programming, Functional Programming, Turing-Completeness, and Beyond.* We review the basic paradigms of computer programming. For the imperative case, we use the simple imperative language of (Davis, Sigal & Weyuker 1994), and also discuss register machines, Turing machines (again), KU machines. We also discuss whether programming beyond the Turing Limit makes sense and can be pursued.
- **Nov 9**: *Hypergraphical Proof and Programming in HyperSlate®.* We here introduce the availability of writing Clojure functions in the context of proofs in HyperSlate®.
- **Nov 12**: *Quantified Modal Logic.* We here explore quantified **S5**, the infamous Barcan Formula. HyperSlate® is used.
- **Nov 16**: *Killer Robots, **D**, and Beyond in HyperSlate® to $\mathcal{DCEC}$.* We begin here by stating the "PAID Problem," and then the approach to it from Bringsjord et al. advocates.

Recall: Schedule Switcheroo

- **Nov 5**: *Pure General Logic Programming, Functional Programming, Turing-Completeness, and Beyond.* We review the basic paradigms of computer programming. For the imperative case, we use the simple imperative language of (Davis, Sigal & Weyuker 1994), and also discuss register machines, Turing machines (again), KU machines. We also discuss whether programming beyond the Turing Limit makes sense and can be pursued.
- **Nov 9**: *Hypergraphical Proof and Programming in HyperSlate®.* We here introduce the availability of writing Clojure functions in the context of proofs in HyperSlate®.
- **Nov 12**: *Quantified Modal Logic.* We here explore quantified **S5**, the infamous Barcan Formula. HyperSlate® is used.
- **Nov 16**: *Killer Robots, **D**, and Beyond in HyperSlate® to $\mathcal{DCEC}$.* We begin here by stating the "PAID Problem," and then the approach to it from Bringsjord et al. advocates.

Recall: Schedule Switcheroo

√

- **Nov 5**: *Pure General Logic Programming, Functional Programming, Turing-Completeness, and Beyond*. We review the basic paradigms of computer programming. For the imperative case, we use the simple imperative language of (Davis, Sigal & Weyuker 1994), and also discuss register machines, Turing machines (again), KU machines. We also discuss whether programming beyond the Turing Limit makes sense and can be pursued.
- **Nov 9**: *Hypergraphical Proof and Programming in HyperSlate*®. We here introduce the availability of writing Clojure functions in the context of proofs in HyperSlate®.
- **Nov 12**: *Quantified Modal Logic*. We here explore quantified **S5**, the infamous Barcan Formula. HyperSlate® is used.

√

- **Nov 16**: *Killer Robots, **D**, and Beyond in HyperSlate® to $\mathcal{DCEC}$*. We begin here by stating the "PAID Problem," and then the approach to it from Bringsjord et al. advocates.

Recall: Schedule Switcheroo

- **Nov 5**: *Pure General Logic Programming, Functional Programming, Turing-Completeness, and Beyond.* We review the basic paradigms of computer programming. For the imperative case, we use the simple imperative language of (Davis, Sigal & Weyuker 1994), and also discuss register machines, Turing machines (again), KU machines. We also discuss whether programming beyond the Turing Limit makes sense and can be pursued.
- **Nov 9**: *Hypergraphical Proof and Programming in HyperSlate®.* We here introduce the availability of writing Clojure functions in the context of proofs in HyperSlate®.
- **Nov 12**: *Quantified Modal Logic.* We here explore quantified **S5**, the infamous Barcan Formula. HyperSlate® is used.
- **Nov 16**: *Killer Robots, **D**, and Beyond in HyperSlate® to $\mathcal{DCEC}$.* We begin here by stating the "PAID Problem," and then the approach to it from Bringsjord et al. advocates.

Q3

Consider a new propositional modal logic: *propositional provability logic*, or for short, **PPL**. We here make use of the familiar "box" and "diamond" we have seen in our propositional modal logics so far, which of course are available in HS®. In **PPL** we read $\Box\phi$ as saying that ϕ is provable, and $\Diamond\phi$ is simply an abbreviation for $\neg\Box\neg\phi$. In order to have **PPL** available to us for exploration in HS®, we simply use **K**, and add to our workspace a formula that expresses this new principle:

(Löb) If it's provable that (if ϕ is provable, then ϕ), then ϕ is provable.

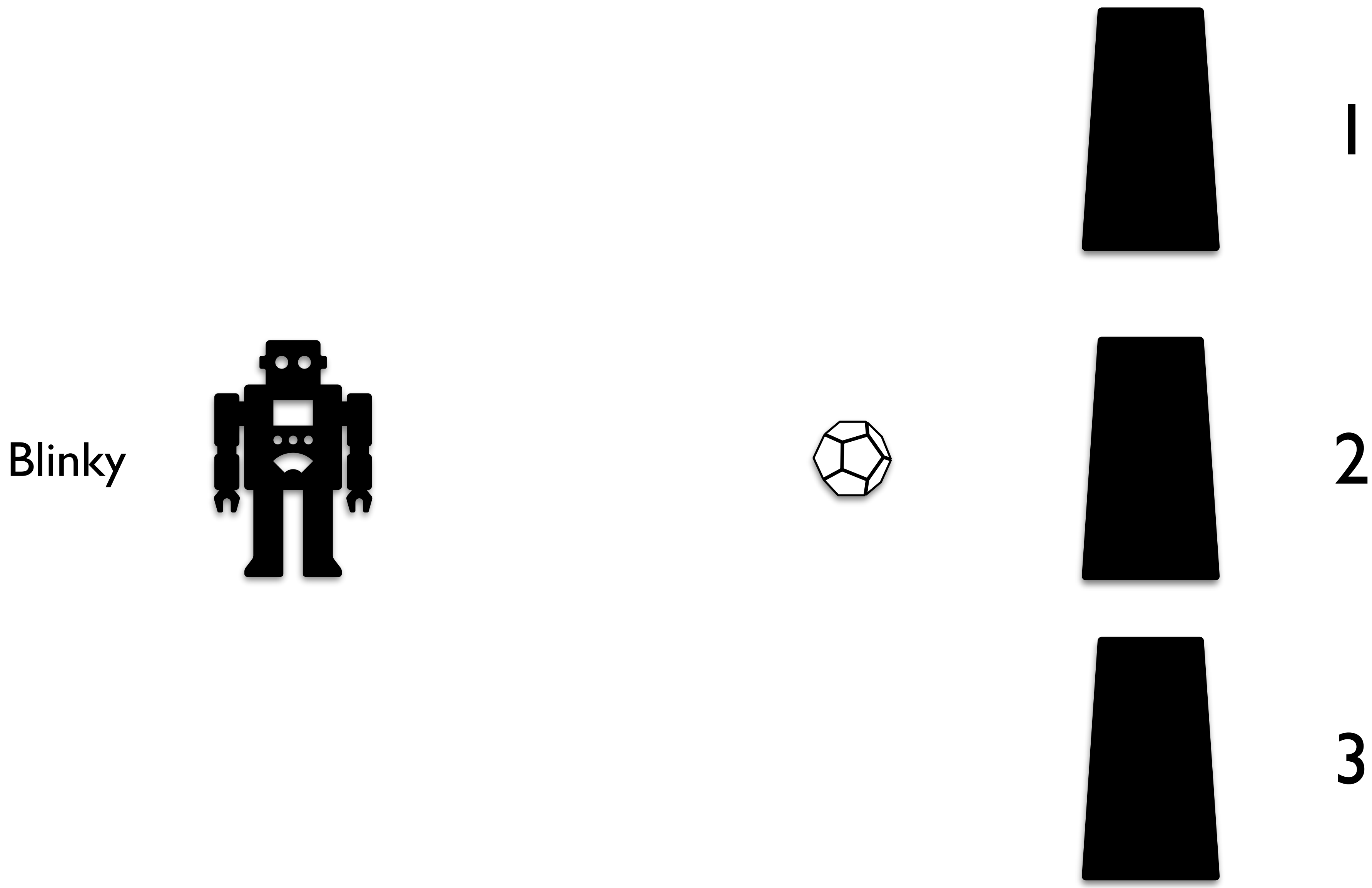
Let <(Löb)> denote this formula. Now here are the two tasks for you in Q3:

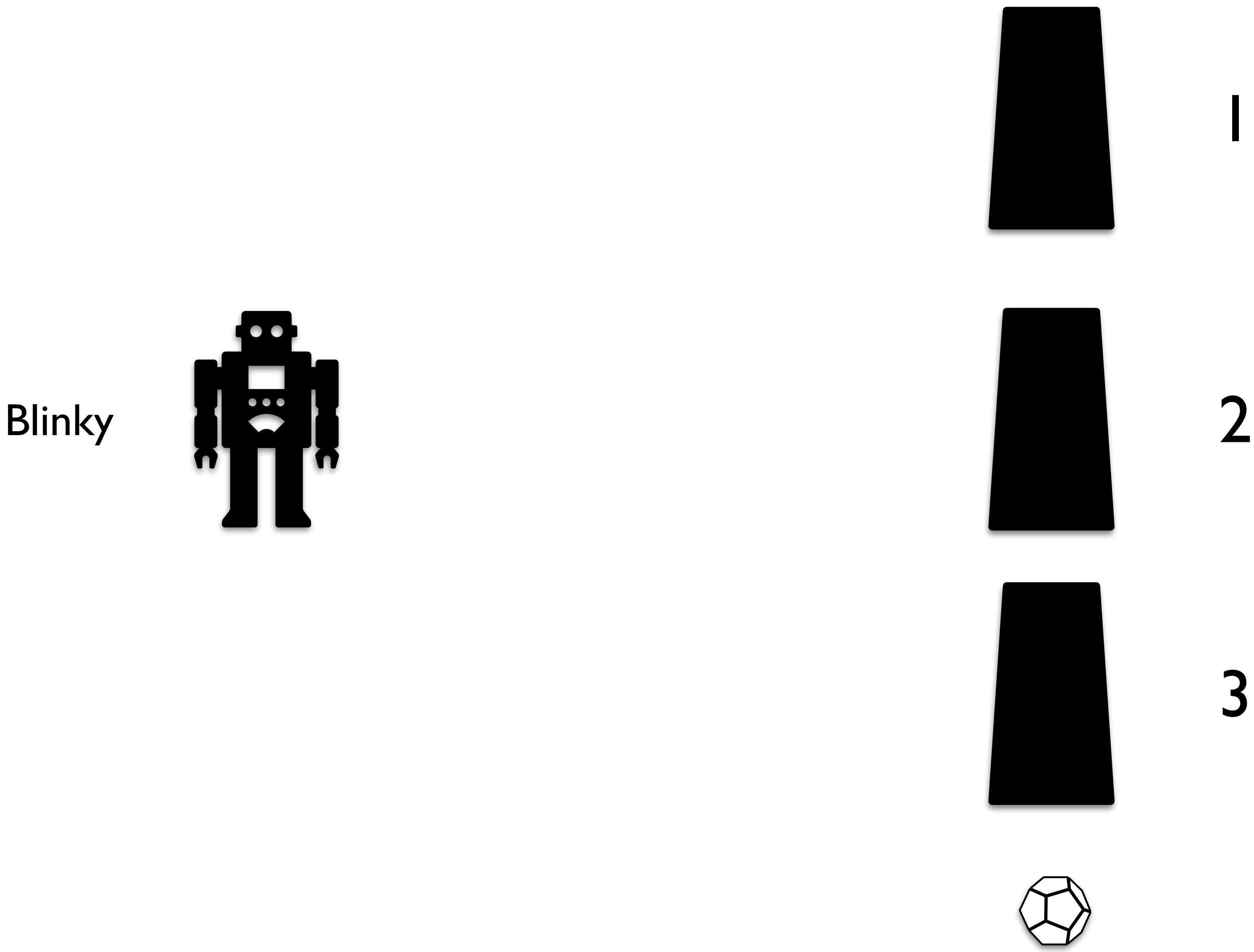
(i) Can the characteristic axiom of **S4** be proved in **PPL**? Prove your answer. (Max one page.)

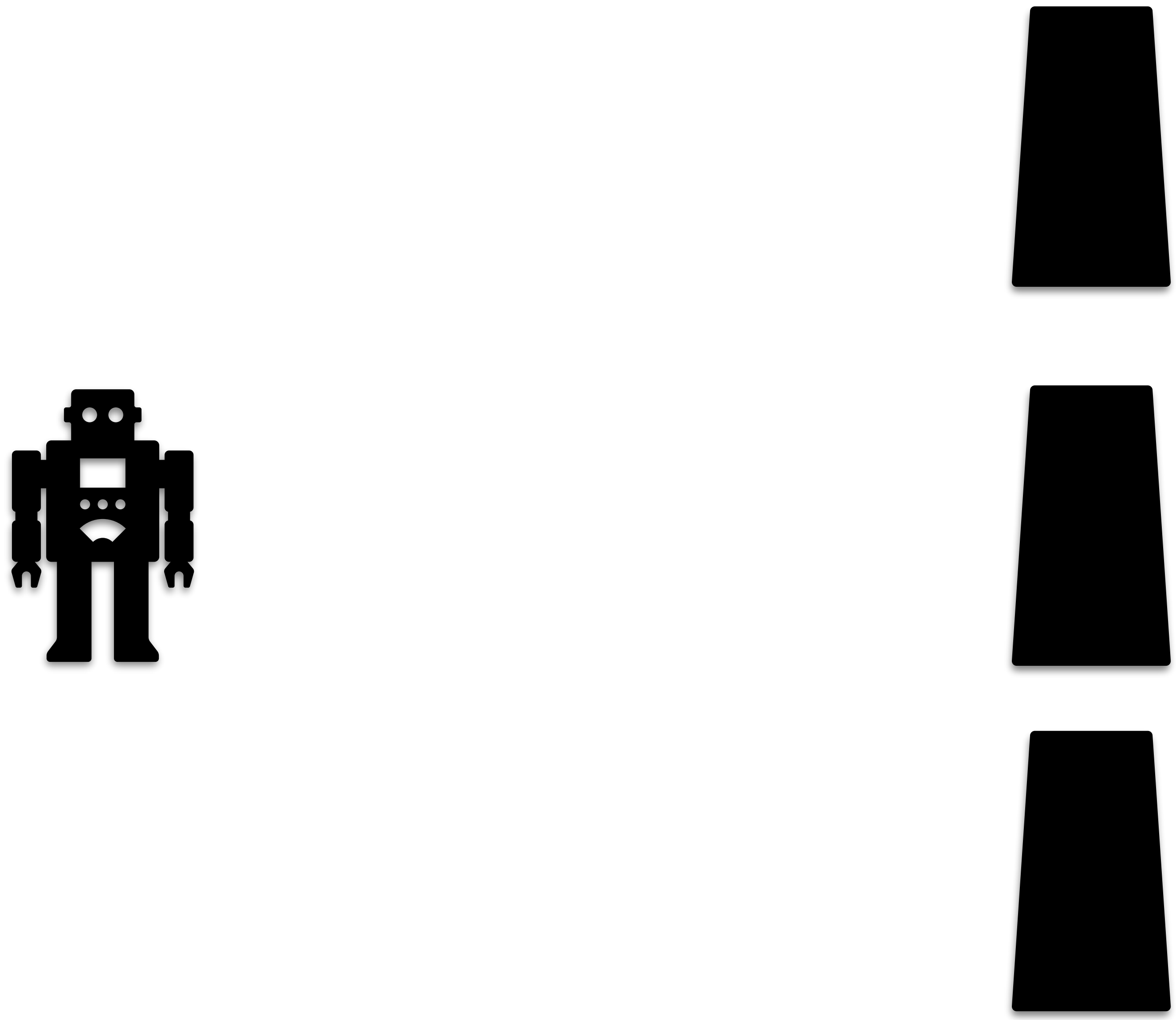
(ii) It would seem that a more interesting and (given what those in the business of proving things do) accurate logic would be *quantified* provability logic (**QPL**), since after all, all interesting theorems have quantifiers and relation symbols in them. After you are clear on what **QPL** amounts to formally, answer the following question, and justify your answer with cogent argumentation.
Question: Can an artificial agent can be engineered which productively uses **QPL**? Max one page.

Return to Return-to-Blinky

...


Blinky
1
2
3


Blinky
1
2
3



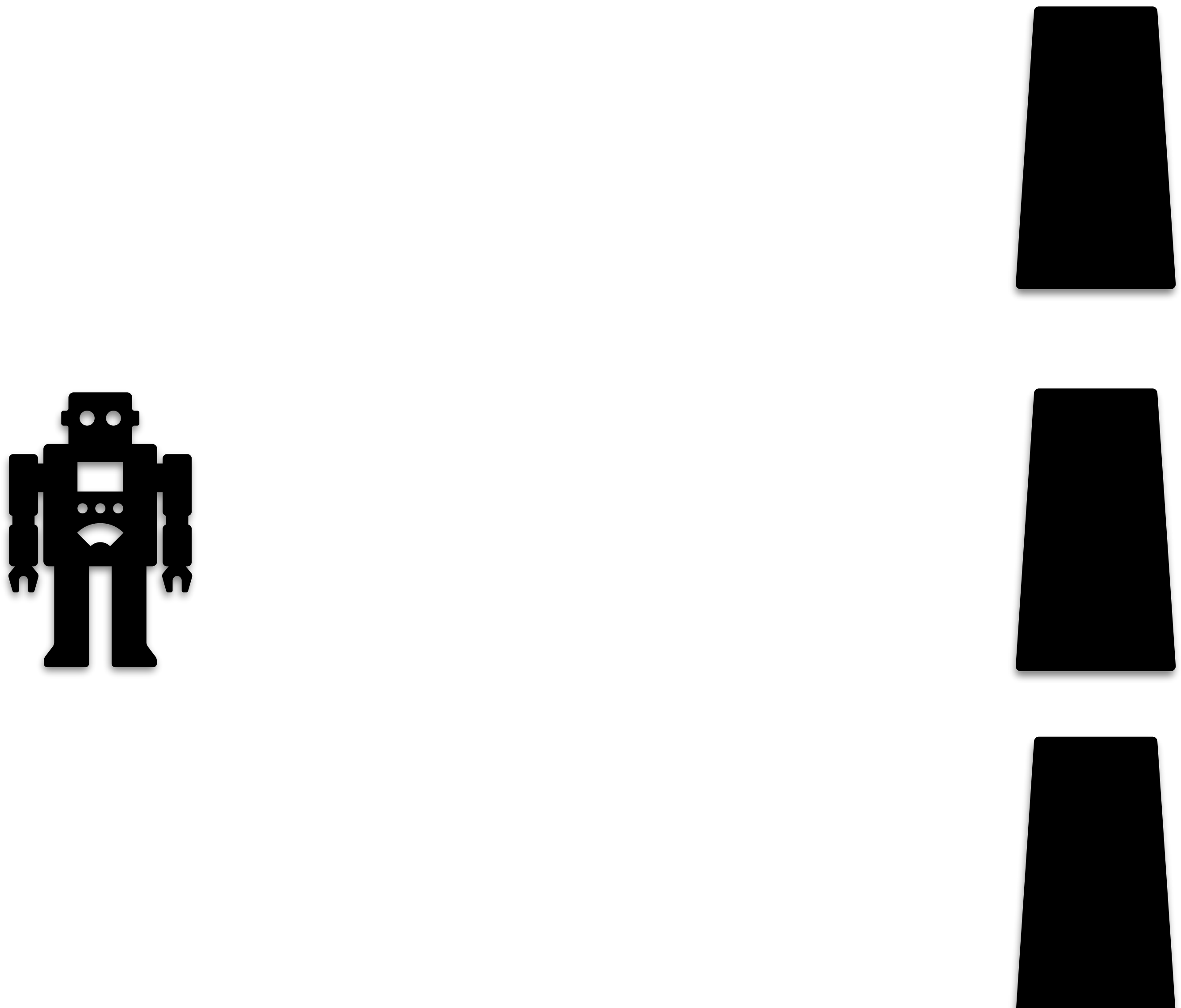
Blinky

1

2

3

Blinky

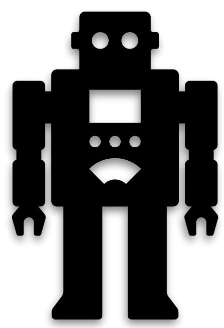


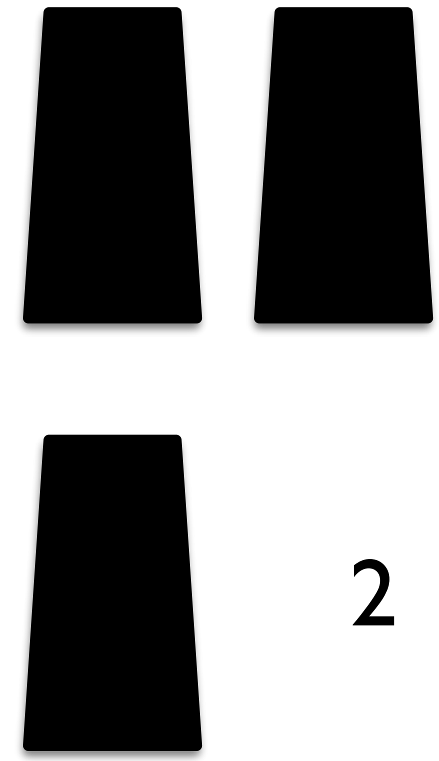
1

2

3

Blinky

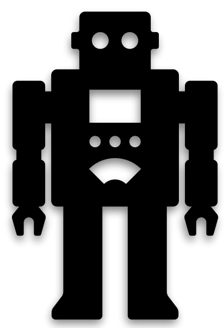


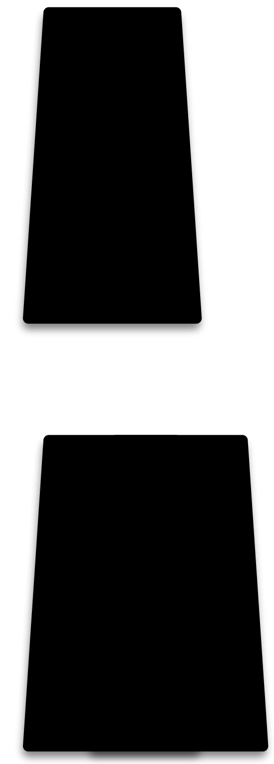


2

3

Blinky



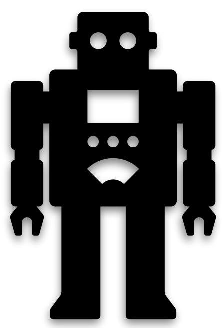


1

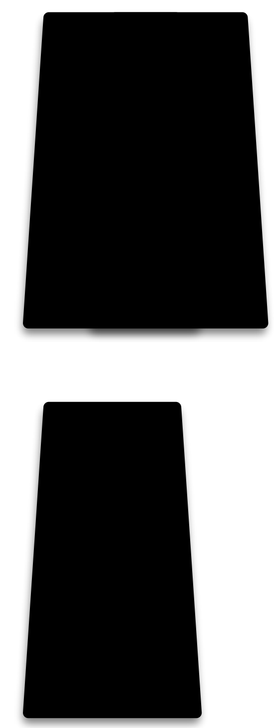
2

3

Blinky

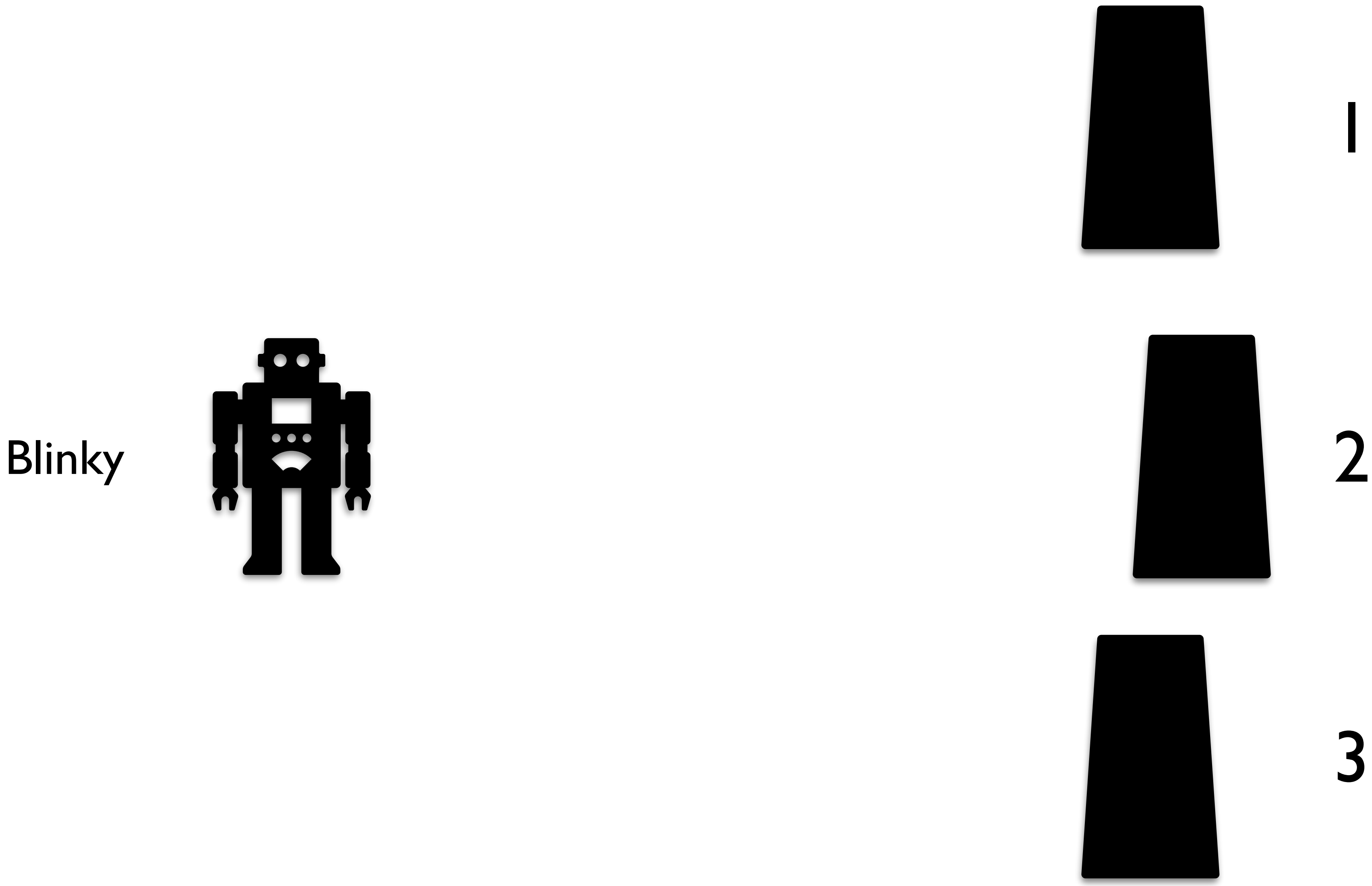


1

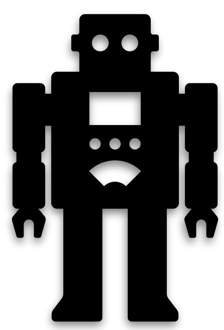


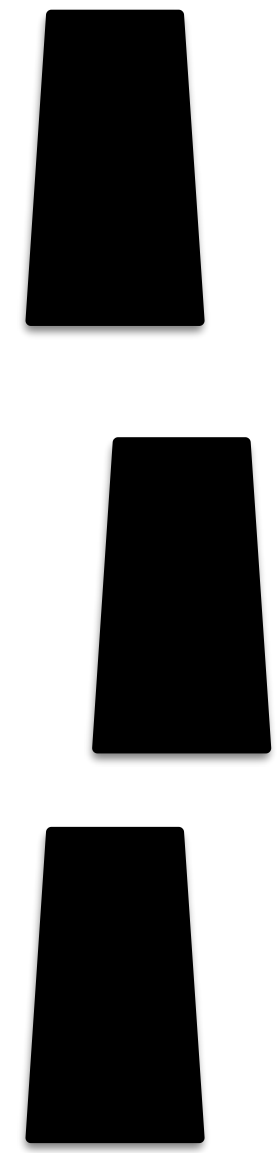
2

3


Blinky
1
2
3

Blinky

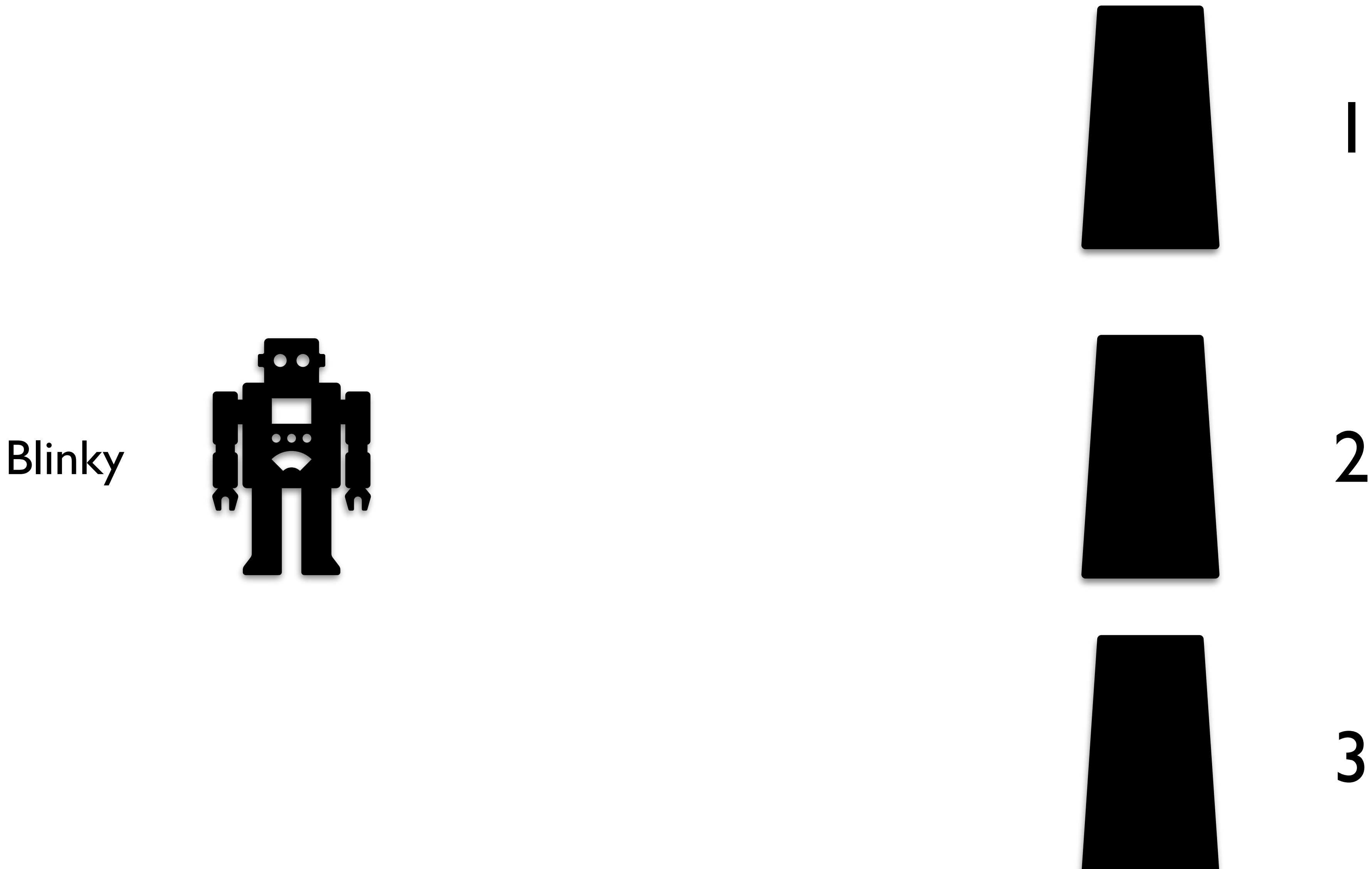




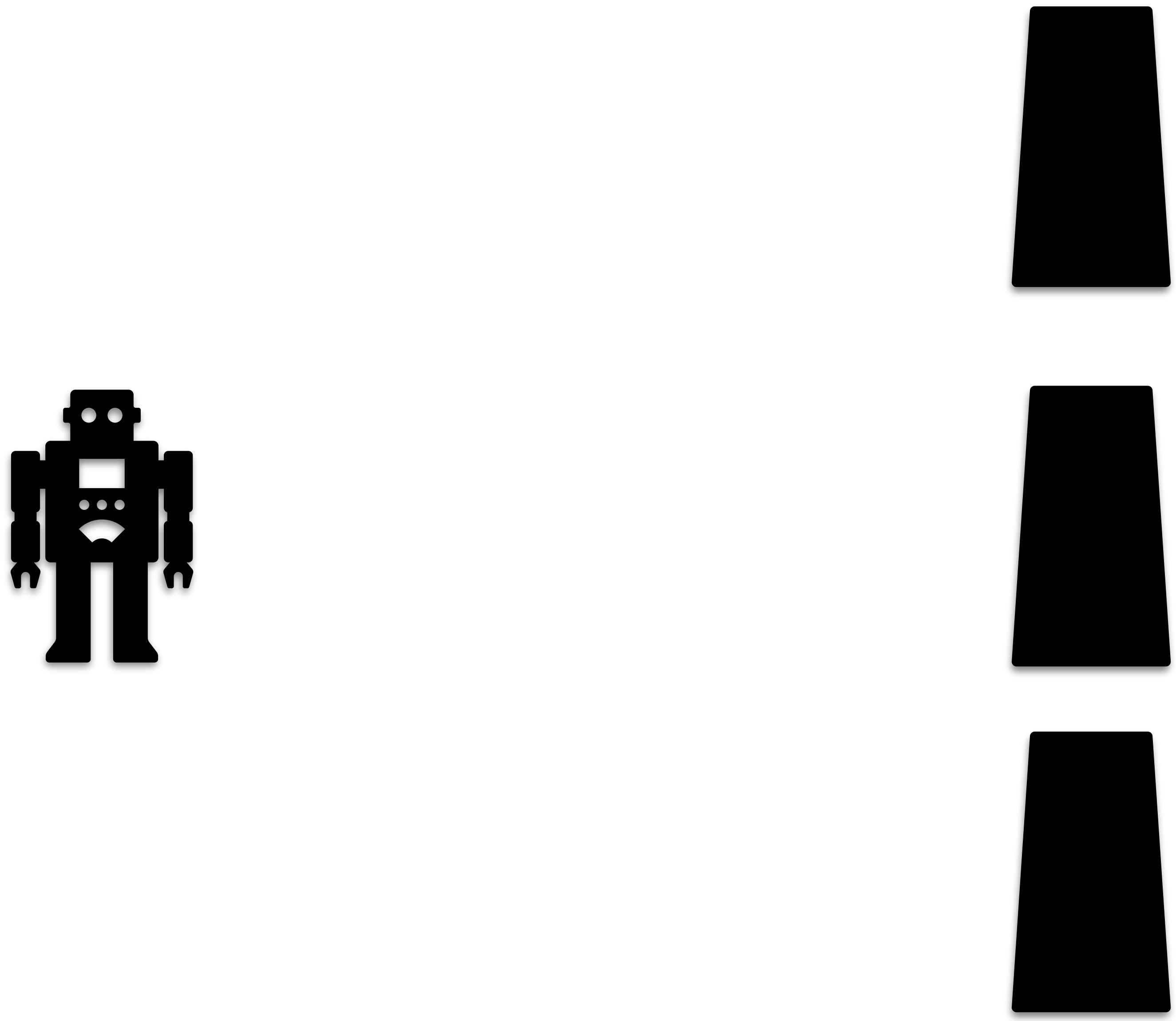
1

2

3

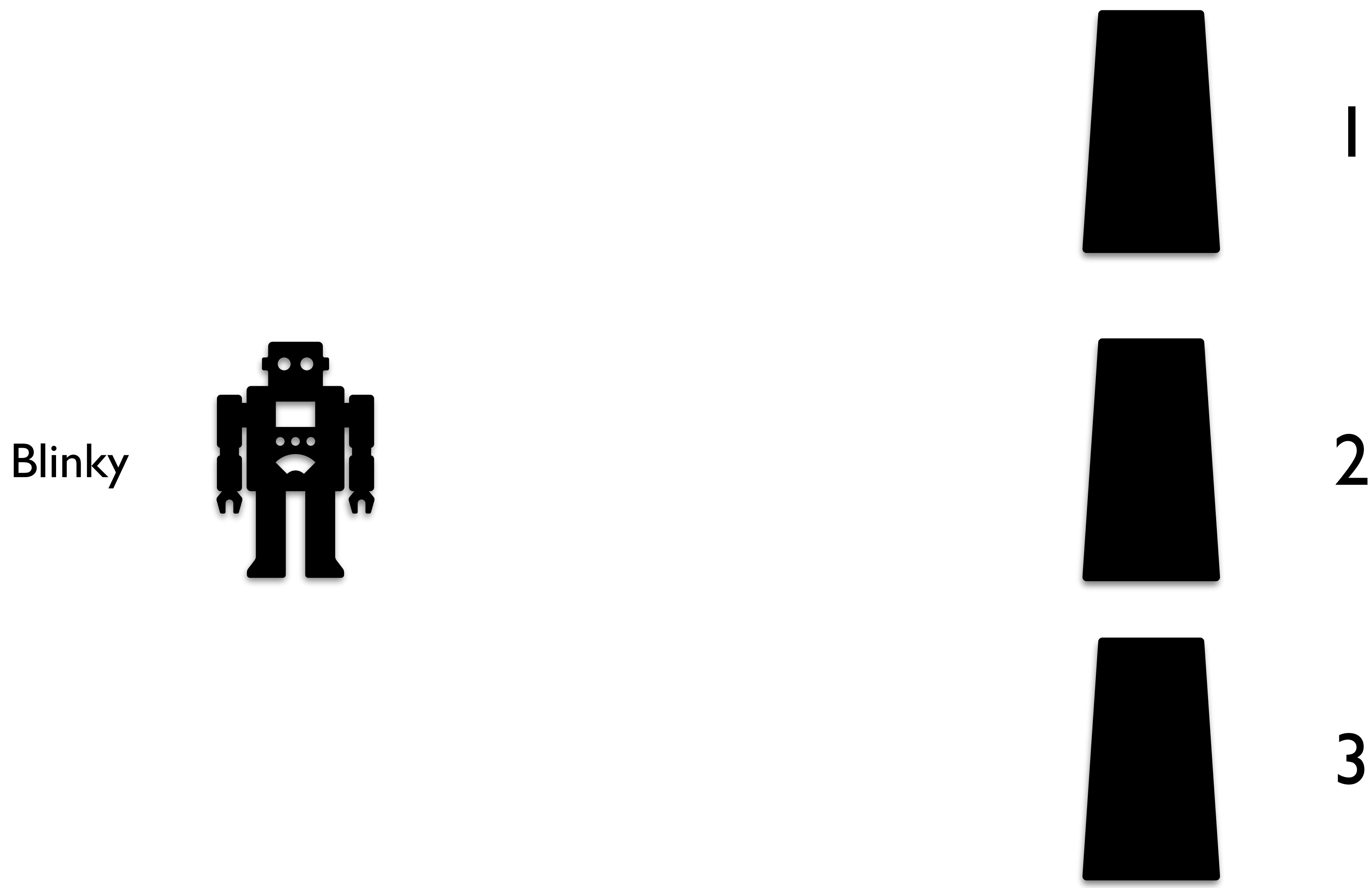

Blinky
1
2
3

The ball is in the cup at location #1.



The ball is in the cup at location #1.

Loc(ball,1)

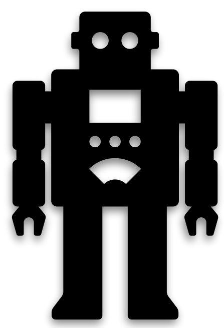


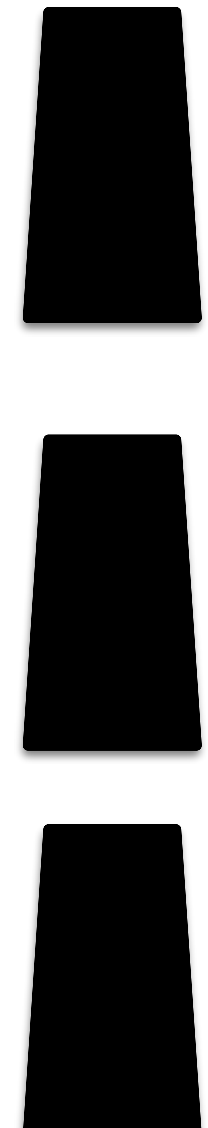
The ball is in the cup at location #1.

Loc(ball,1)

```
(Loc ball 1)
```

Blinky





1

2

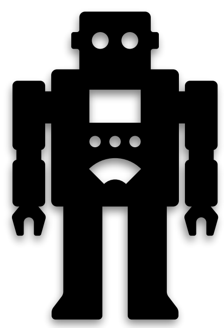
3

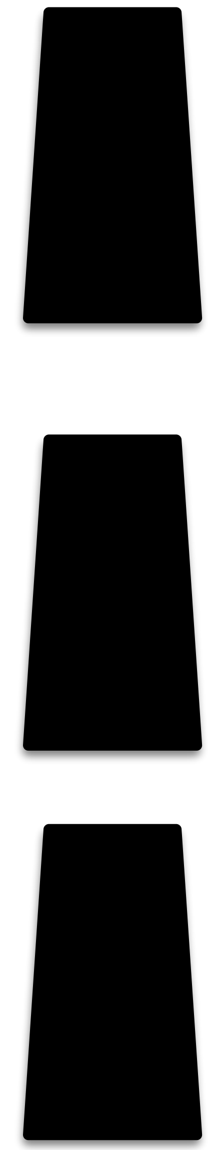
The ball is in the cup at location #1.

FALSE $\mathrm{Loc}(\mathrm{ball}, 1)$

```
(Loc ball 1)
```

Blinky





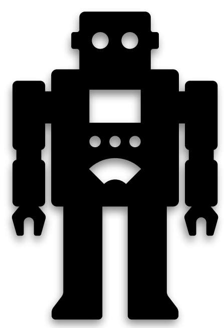
1

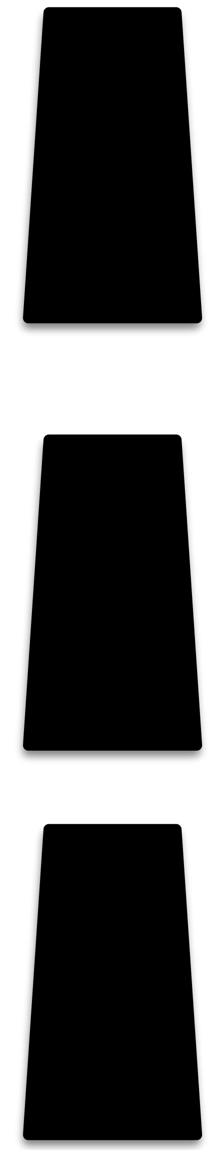
2

3

FALSE Loc(ball,1)

```
(Loc ball 1)
```

Blinky 



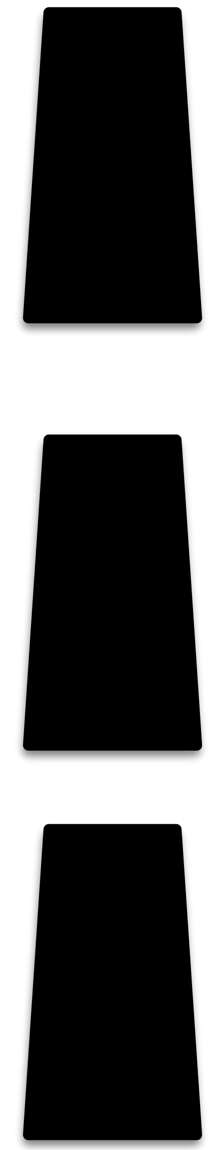
1

2

3

FALSE

`(Loc ball 1)`

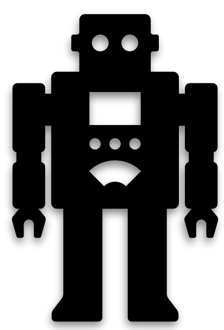


1

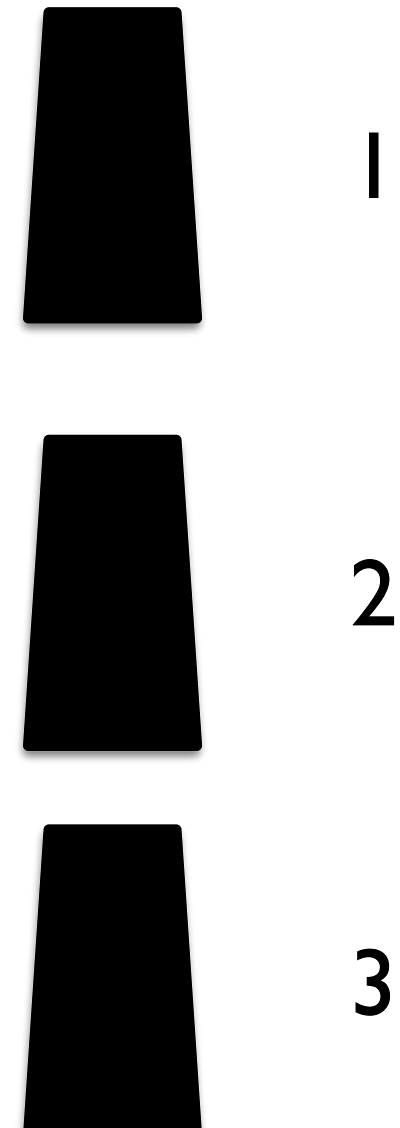
2

3

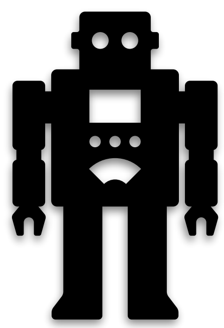
Blinky

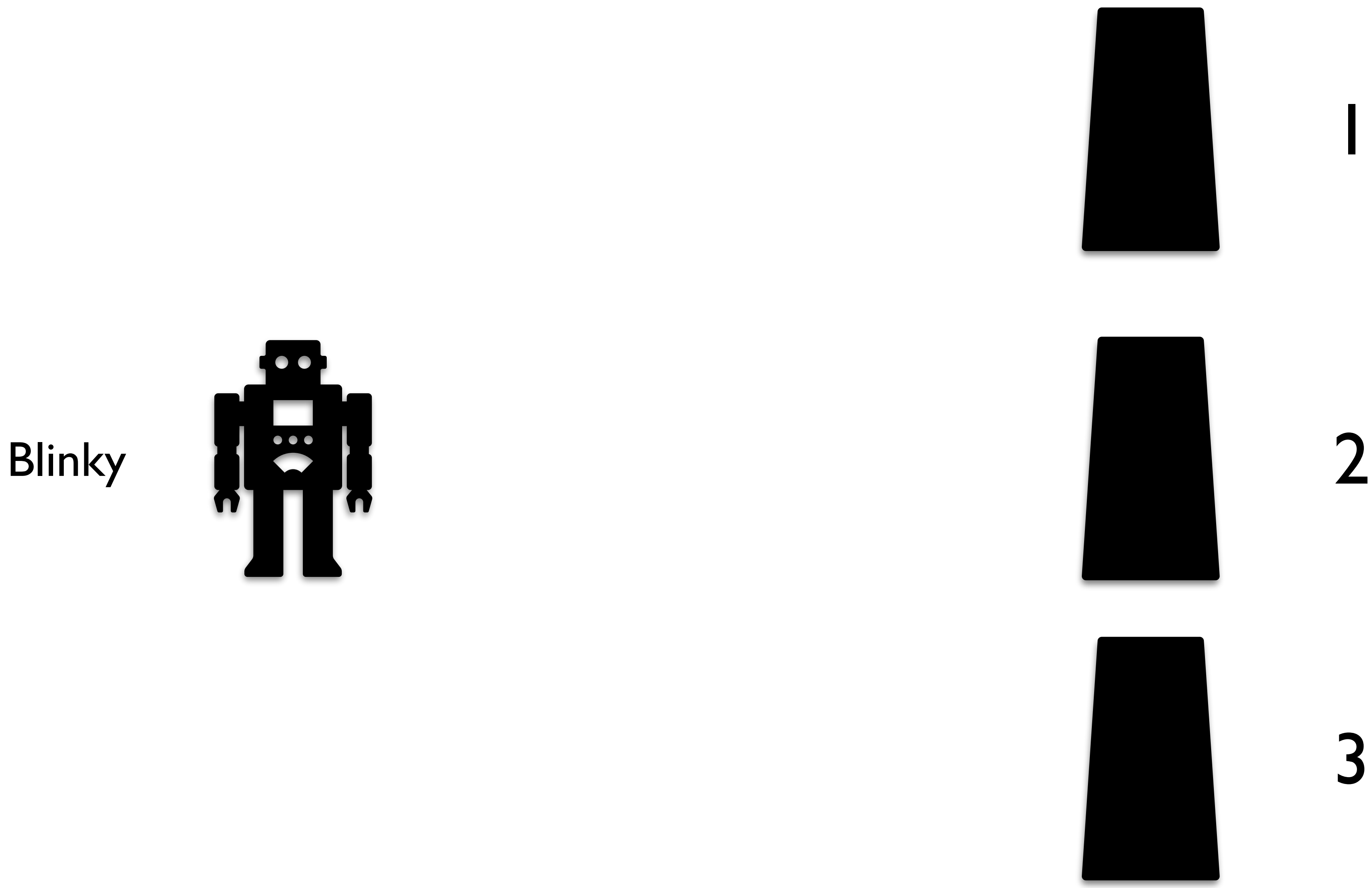


(Loc ball 1)

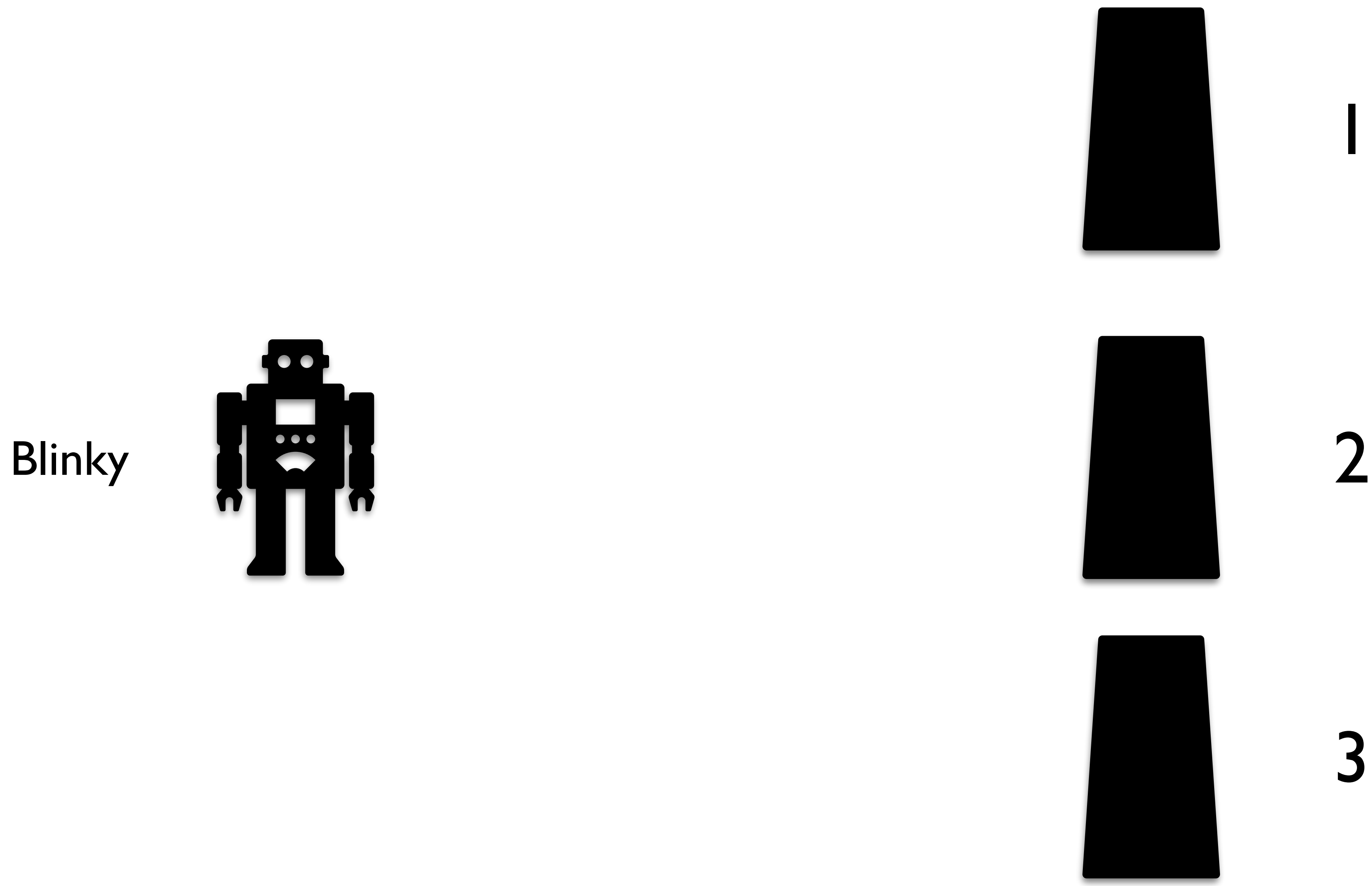


Blinky



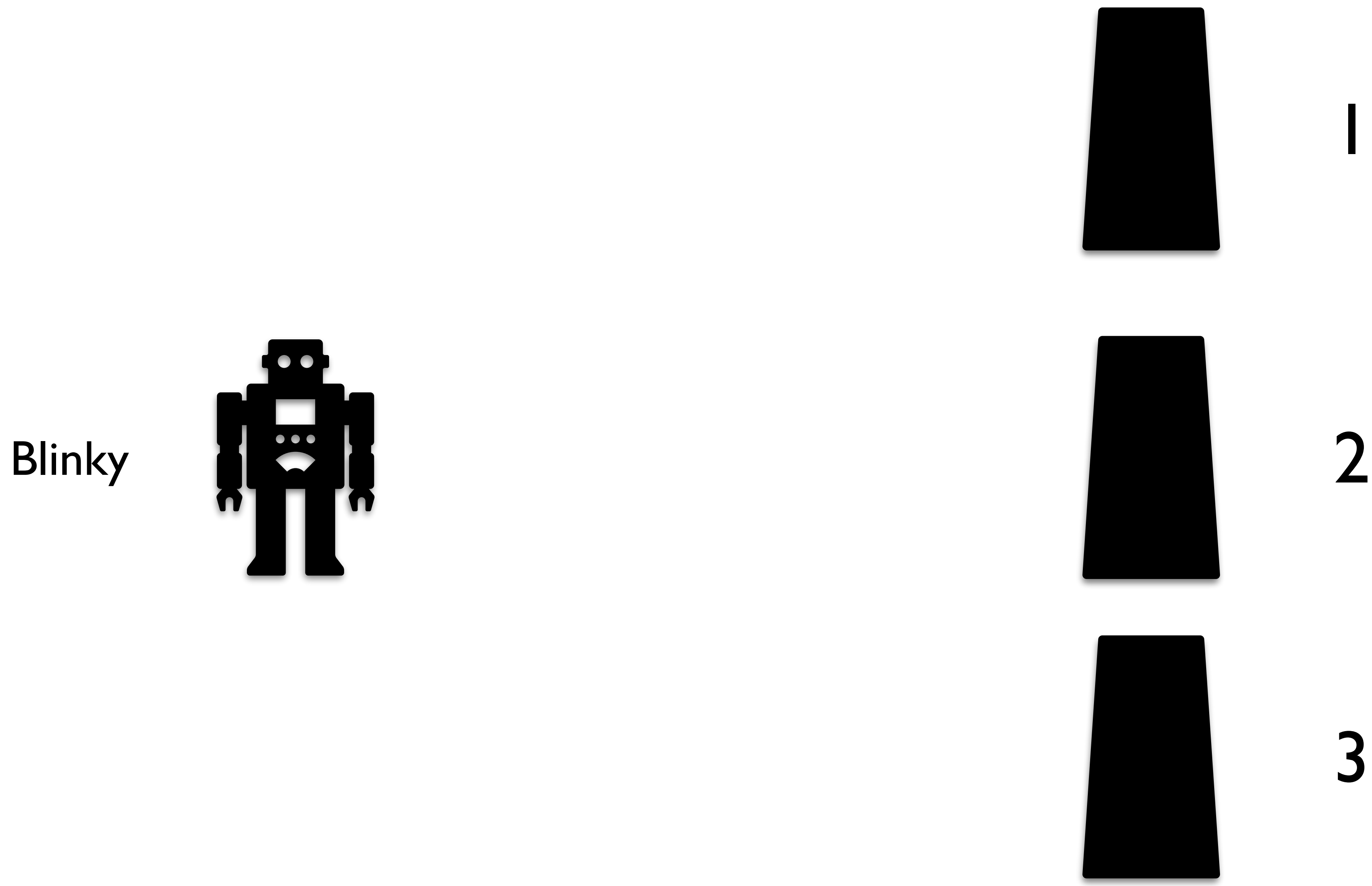

Blinky
1
2
3

The ball is in the cup at location #1 or the ball is at location #3.



The ball is in the cup at location #1 or the ball is at location #3.

$$\mathrm{Loc}(\mathrm{ball},1) \vee \mathrm{Loc}(\mathrm{ball},3)$$

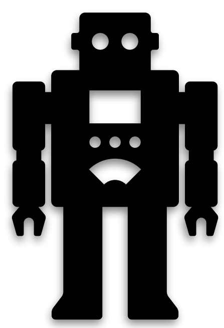


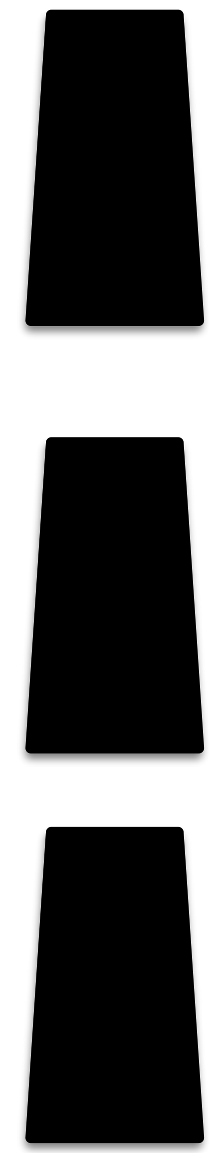
The ball is in the cup at location #1 or the ball is at location #3.

$$\text{Loc}(\text{ball},1) \vee \text{Loc}(\text{ball},3)$$

```
(or (Loc ball 1) (Loc ball 3))
```

Blinky





1

2

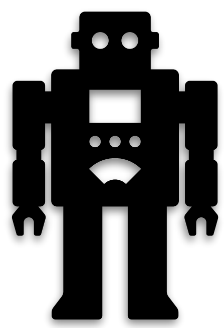
3

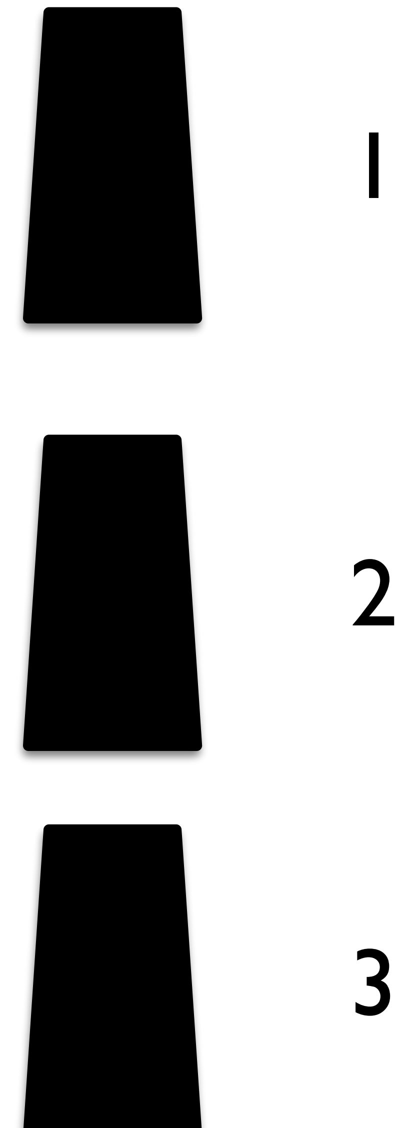
The ball is in the cup at location #1 or the ball is at location #3.

FALSE $\mathrm{Loc}(\mathrm{ball},1) \vee \mathrm{Loc}(\mathrm{ball},3)$

```
(or (Loc ball 1) (Loc ball 3))
```

Blinky

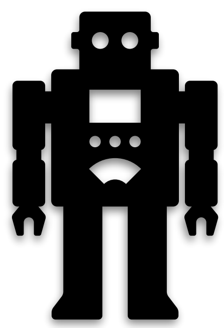


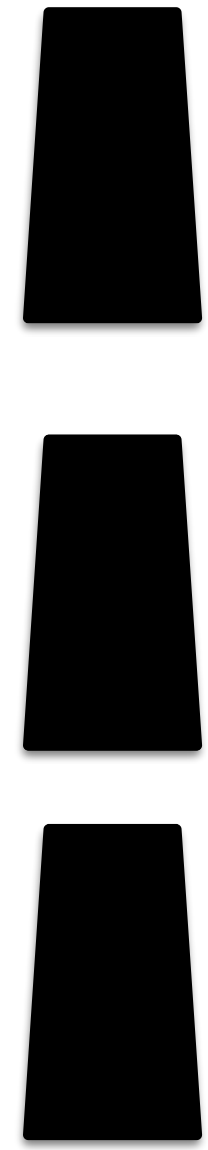


FALSE $Loc(ball,1) \vee Loc(ball,3)$

```
(or (Loc ball 1) (Loc ball 3))
```

Blinky





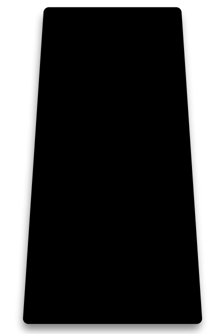
1

2

3

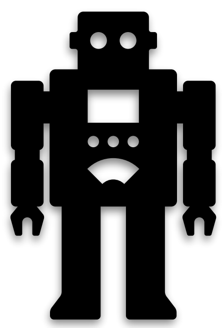
FALSE

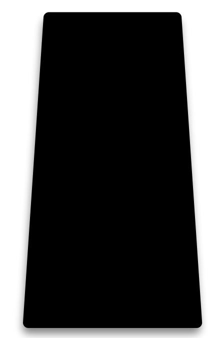
`(or (Loc ball 1) (Loc ball 3))`



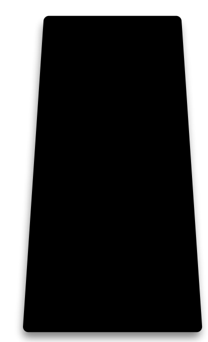
1

Blinky





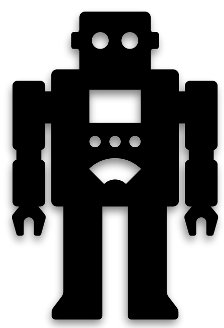
2

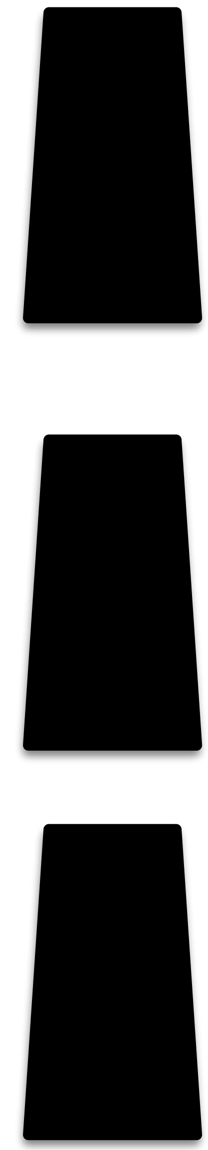


3

FALSE

Blinky

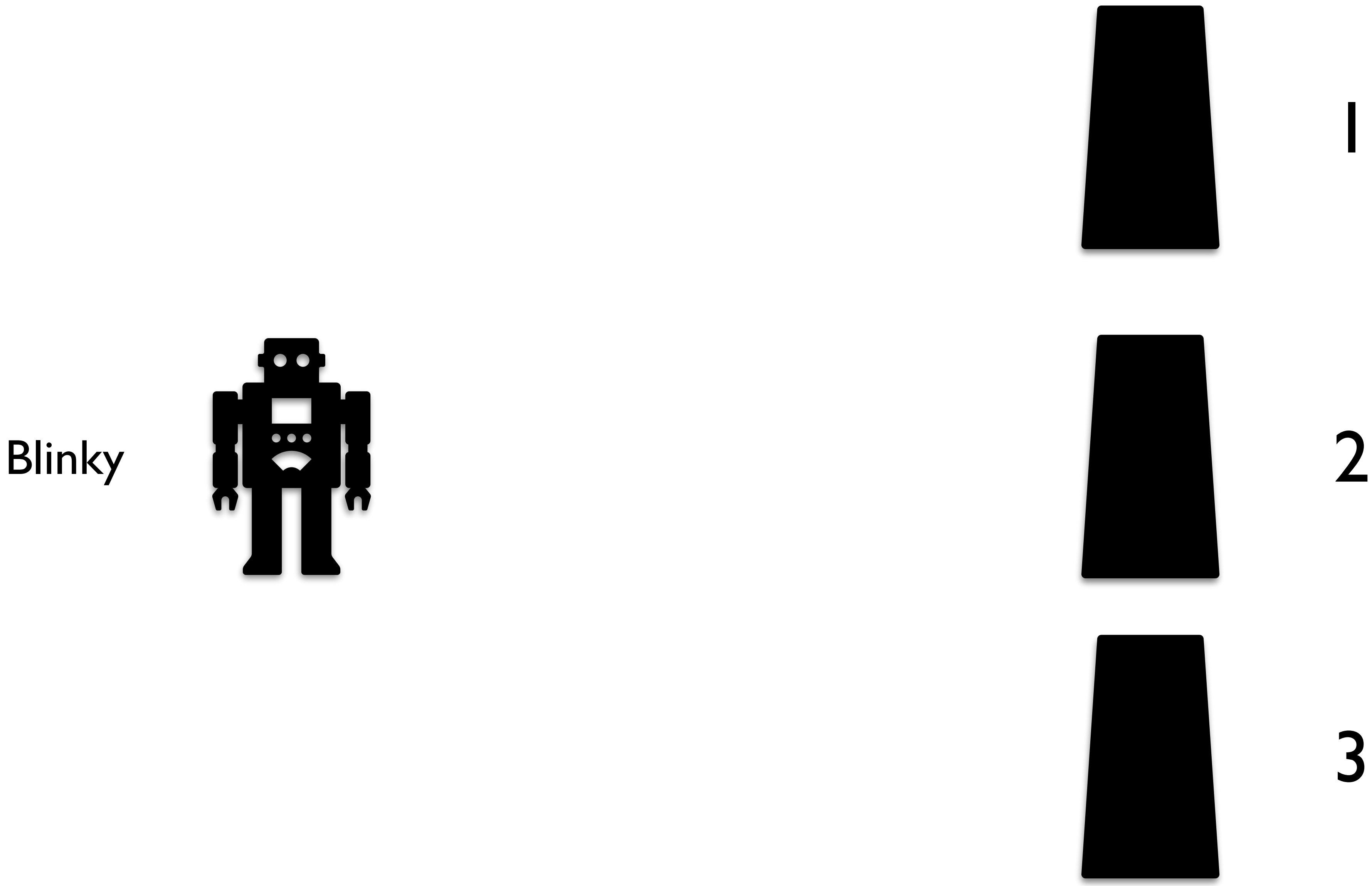




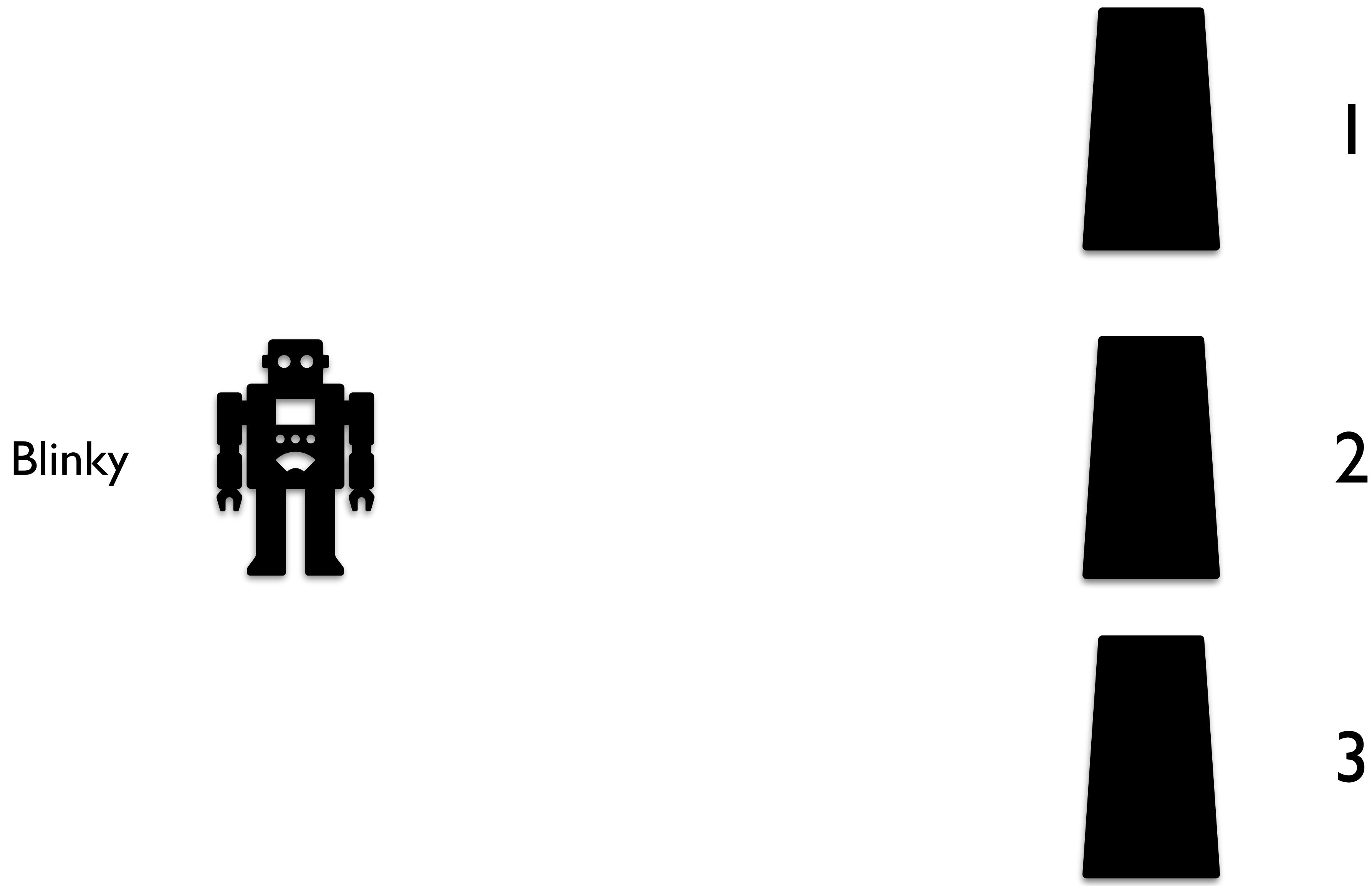
1

2

3

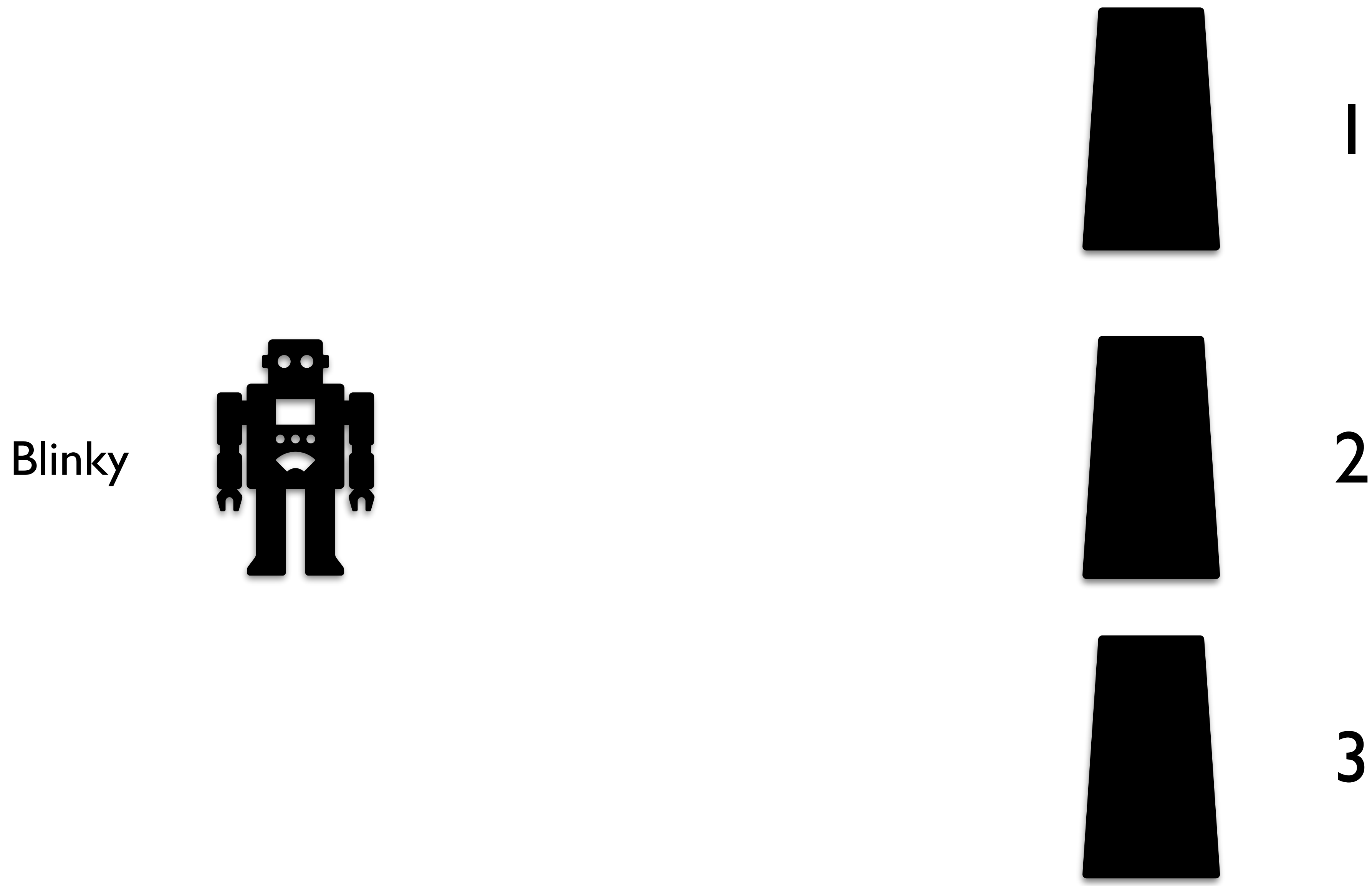

Blinky
1
2
3

Blinky believes that the ball is in the cup at location #1.



Blinky believes that the ball is in the cup at location #1.

$$B(\text{blinky}, \text{Loc}(\text{ball}, 1))$$

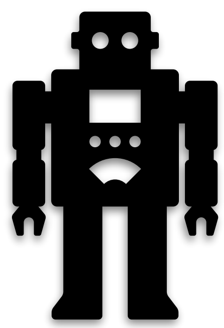


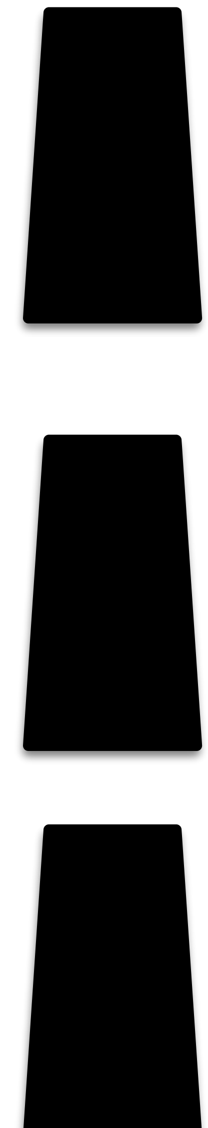
Blinky believes that the ball is in the cup at location #1.

$$B(blinky, Loc(ball, 1))$$

```
(Believes! t blinky (Loc ball 1))
```

Blinky





1

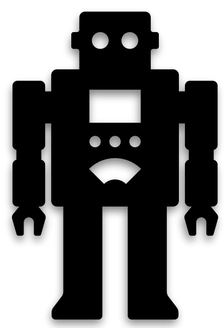
2

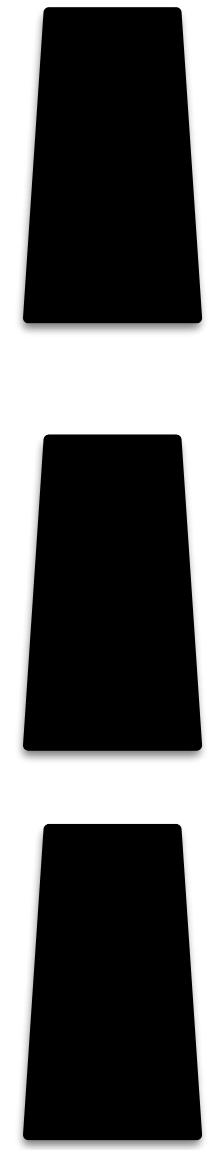
3

Blinky believes that the ball is in the cup at location #1.

??? $B(blinky, Loc(ball, 1))$

```
(Believes! t blinky (Loc ball 1))
```

Blinky 



1

2

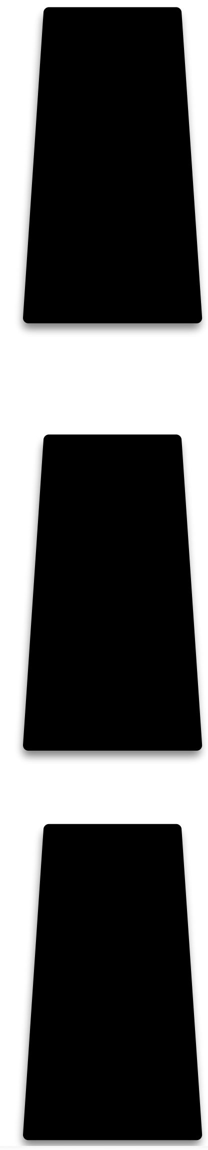
3

Blinky believes that the ball is in the cup at location #1.

???

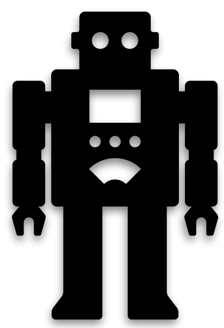
$$B(\text{blinky}, \text{Loc}(\text{ball}, 1))$$

```
(Believes! t blinky (Loc ball 1))
```



1

Blinky



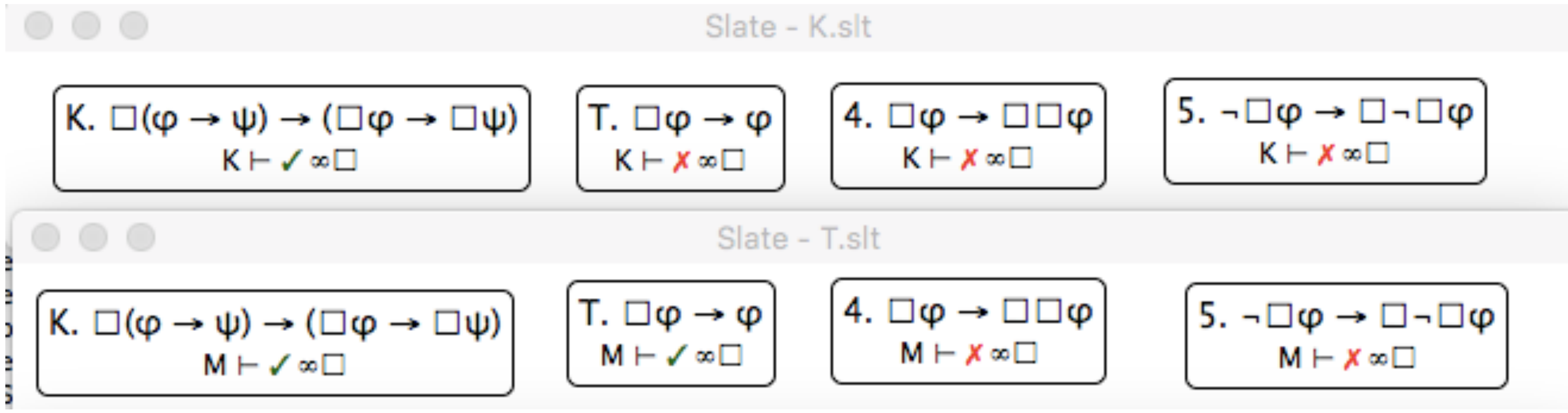
2

3

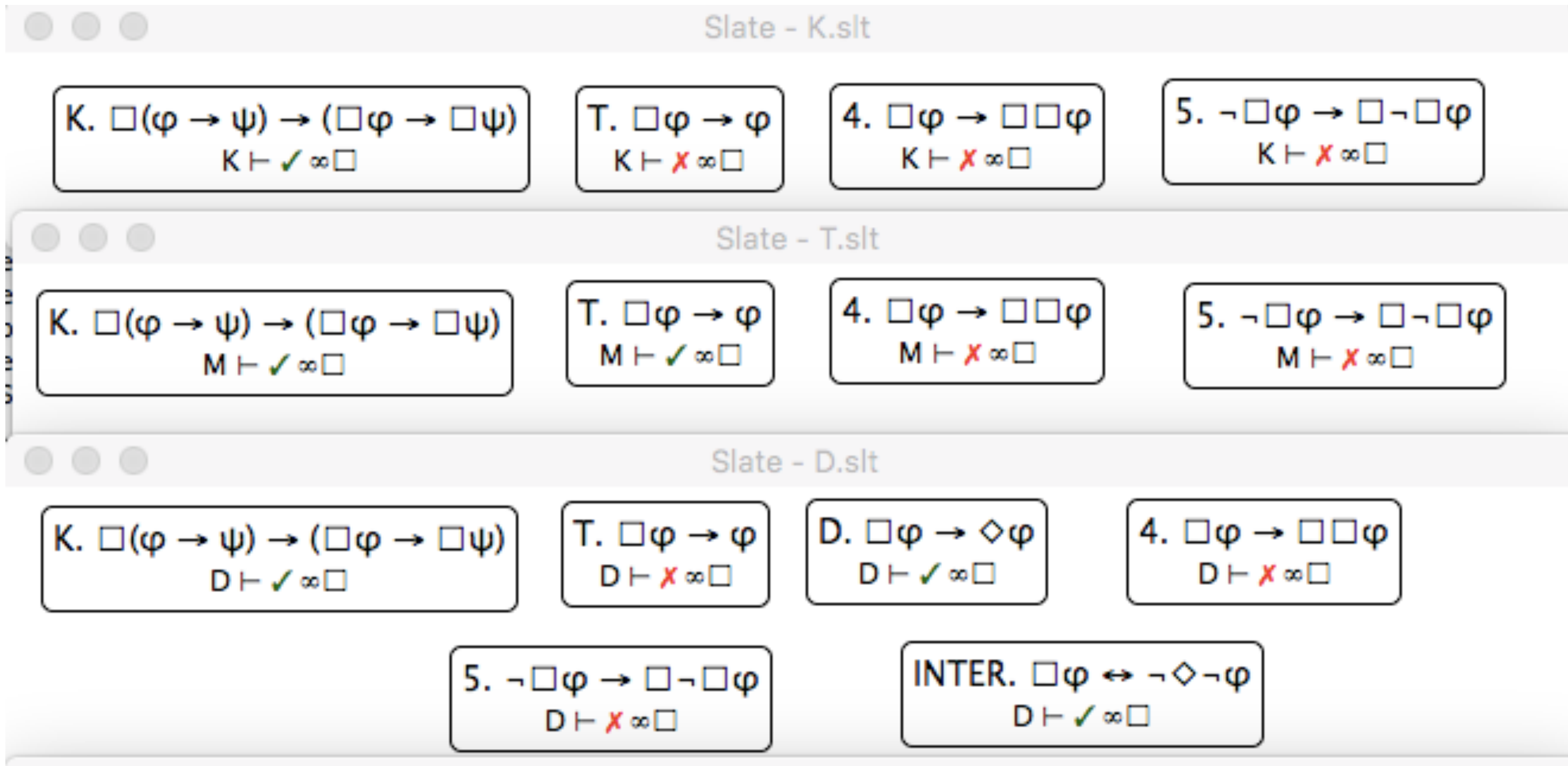
In extensional logics, what is denoted is conflated with meaning (the latter being naïvely compositional), and intensional attitudes like *believes*, *knows*, *hopes*, *fears*, etc cannot be represented and reasoned over smoothly (e.g. without fear of inconsistency rising up).

Review: Encapsulation

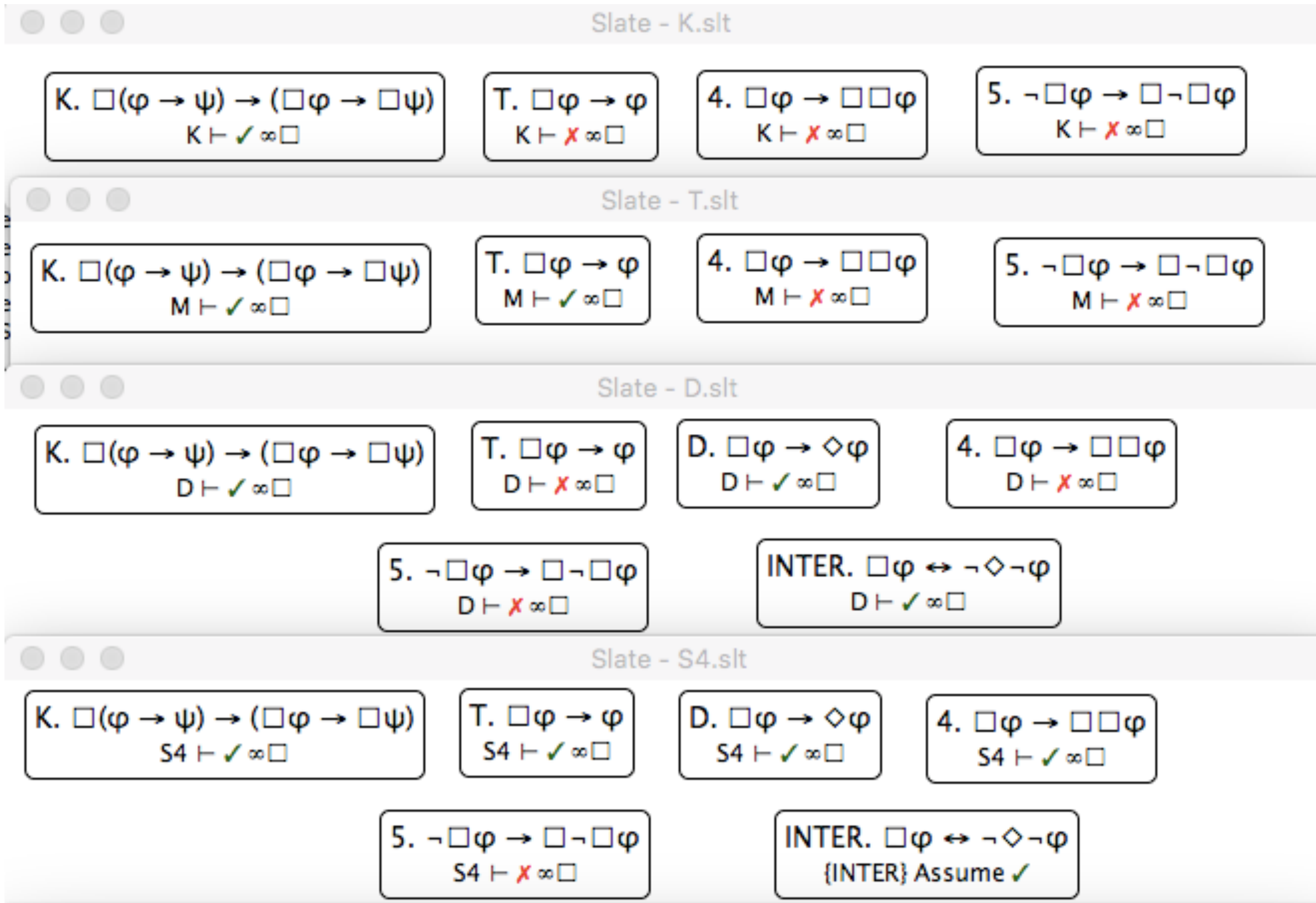
Review: Encapsulation



Review: Encapsulation



Review: Encapsulation



Review: Encapsulation

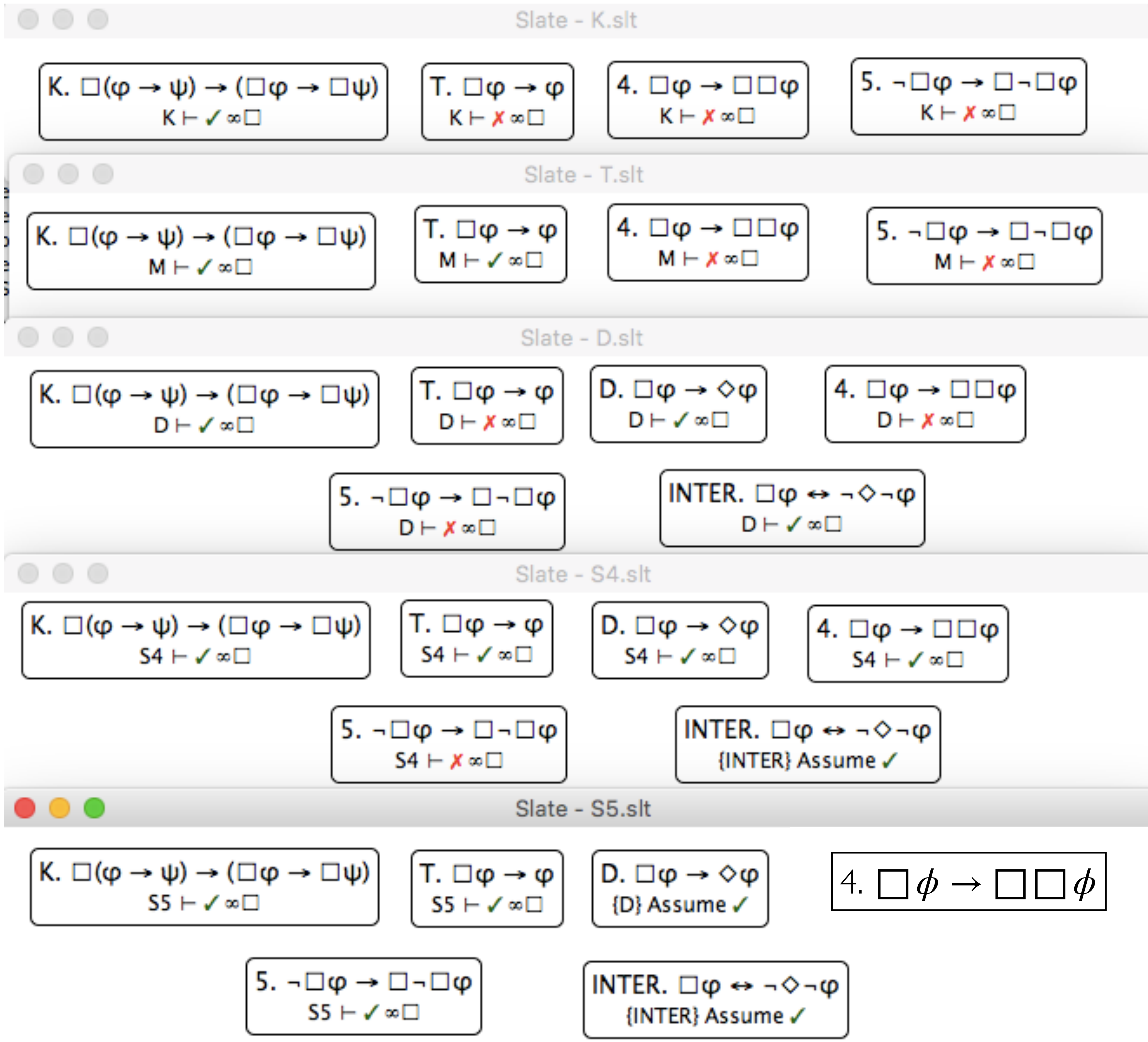
K

T

D

4 = S4

5 = S5



Review: Encapsulation

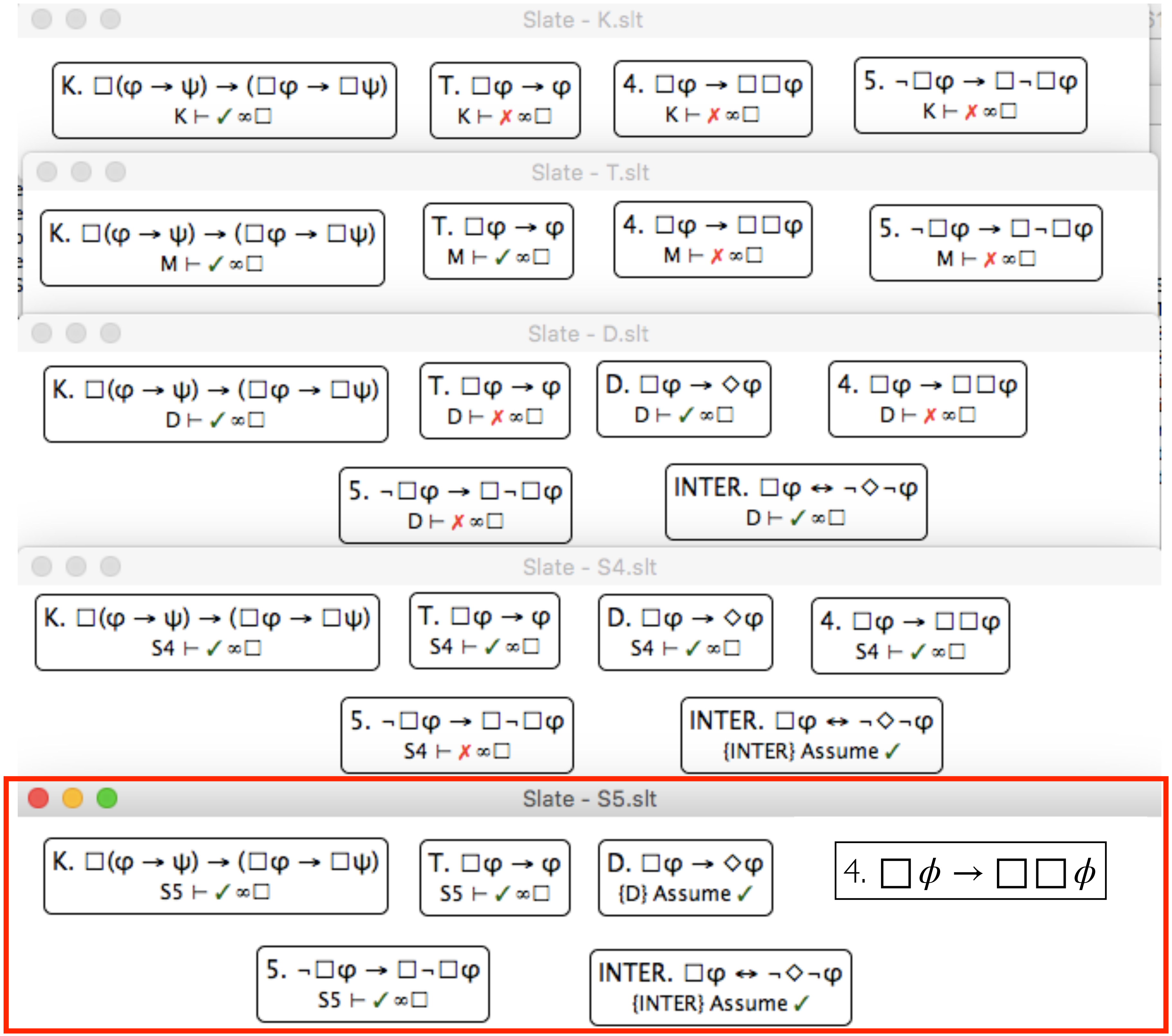
K

T

D

4 = S4

5 = S5



S5 ...

The Characteristic Axiom

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

The Characteristic Axiom

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

$$\neg\Box\psi \rightarrow \Box\neg\Box\psi$$

The Characteristic Axiom

$$\Diamond\phi \;\rightarrow\; \Box\Diamond\phi$$

$$\neg\Box\psi \;\rightarrow\; \Box\neg\Box\psi$$

The Characteristic Axiom

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

$$\neg\Box\psi \rightarrow \Box\neg\Box\psi$$

$$\neg\Box\neg\phi \rightarrow \Box\neg\Box\neg\phi$$

The Characteristic Axiom

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

$$\neg\Box\psi \rightarrow \Box\neg\Box\psi$$

$$\neg\Box\neg\phi \rightarrow \Box\neg\Box\neg\phi$$

The Characteristic Axiom

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

$$\neg\Box\psi \rightarrow \Box\neg\Box\psi$$

$$\neg\Box\neg\phi \rightarrow \Box\neg\Box\neg\phi$$

$$\Diamond\phi \rightarrow \Box\Diamond\phi$$

Nice S5 Reduction Lemmas

Nice S5 Reduction Lemmas

The Four Possible Pairs

$\Box\ \Box$

$\Diamond\ \Diamond$

$\Box\ \Diamond$

$\Diamond\ \Box$

Nice S5 Reduction Lemmas

<u>The Four Possible Pairs</u>

$\Box\,\Box$

$\Diamond\,\Diamond$

$\Box\,\Diamond$

$\Diamond\,\Box$

<u>The Four Reduction Principles</u>

$$\Box\phi \leftrightarrow \Box\Box\phi$$

$$\Diamond\phi \leftrightarrow \Diamond\Diamond\phi$$

$$\Box\phi \leftrightarrow \Diamond\Box\phi$$

$$\Diamond\phi \leftrightarrow \Box\Diamond\phi$$

(where $\phi \in \mathscr{L}_{pc}$)

Nice S5 Reduction Lemmas

The Four Possible Pairs

$\Box\ \Box$
$\Diamond\ \Diamond$
$\Box\ \Diamond$
$\Diamond\ \Box$

The Four Reduction Principles

$$\Box\phi \leftrightarrow \Box\Box\phi$$

$$\Diamond\phi \leftrightarrow \Diamond\Diamond\phi$$

$$\Box\phi \leftrightarrow \Diamond\Box\phi$$

$$\Diamond\phi \leftrightarrow \Box\Diamond\phi$$

(where $\phi \in \mathscr{L}_{pc}$)

(verify in HS®)

Quantificational S5$_I$ ...

Quantificational $S5_1$ ...

Quantificational S5$_I$ ...

Easy peasy: Marry **PS5** + $\mathscr{L}_1$!

Quantificational S5$_1$ …

Easy peasy: Marry **PS5** + $\mathscr{L}_1$!

Theorem: $\forall x \Diamond R(x) \rightarrow \forall x \Box \Diamond R(x)$

Quantificational S5$_1$ …

Easy peasy: Marry **PS5** + $\mathscr{L}_1$!

Theorem: $\forall x \Diamond R(x) \to \forall x \Box \Diamond R(x)$

Theorem: $\Diamond \exists x R(x) \leftrightarrow \exists x \Diamond R(x)$

Quantificational S5$_1$ …

Easy peasy: Marry **PS5** + $\mathscr{L}_1$!

Theorem: $\forall x \Diamond R(x) \to \forall x \Box \Diamond R(x)$

Theorem: $\Diamond \exists x R(x) \leftrightarrow \exists x \Diamond R(x)$

$\vdots$

Quantificational S5$_1$ …

Easy peasy: Marry **PS5** + $\mathscr{L}_1$!

Theorem: $\forall x \Diamond R(x) \rightarrow \forall x \Box \Diamond R(x)$

(verify in HS®)

Theorem: $\Diamond \exists x R(x) \leftrightarrow \exists x \Diamond R(x)$

$\vdots$

The Notorious Barcans

The Notorious Barcans

Barcan Formula: $\vdash_{QS5_1} \Diamond \exists x \phi(x) \rightarrow \exists x \Diamond \phi(x)$

The Notorious Barcans

Barcan Formula: $\vdash_{QS5_1} \Diamond \exists x \phi(x) \rightarrow \exists x \Diamond \phi(x)$

Converse Barcan Formula: $\vdash_{QS5_1} \exists x \Diamond \phi(x) \rightarrow \Diamond \exists x \phi(x)$

The Notorious Barcans

Barcan Formula: $\vdash_{QS5_1} \Diamond \exists x \phi(x) \rightarrow \exists x \Diamond \phi(x)$

Converse Barcan Formula: $\vdash_{QS5_1} \exists x \Diamond \phi(x) \rightarrow \Diamond \exists x \phi(x)$

Minds & Machines (2017) 27:663–672
https://doi.org/10.1007/s11023-017-9454-1

CrossMark

An Argument for P=NP

Selmer Bringsjord[1,2]

Received: 8 November 2017 / Accepted: 9 November 2017 / Published online: 24 November 2017

Abstract I articulate a novel modal argument for **P=NP**.

Keywords **P=NP** · Modal logic · Digital physics

The Clay Mathematics Institute offers a $1 million prize for a solution to the **P=?NP** problem.[1] I look forward to receiving my award—but concede that the expected format of a solution is an *object-level* proof, not a meta-level argument like what I provide. On the other hand, certainly the winner needn't provide a *constructive* proof that **P=NP**.[2] Despite Gödel's recently discovered position on the

[1] See http://www.claymath.org/millennium. There are six other "millennium" problems; each of these is also associated with a $1M prize.

[2] As many readers know, the history of the problem is littered with failed attempts to provide non-constructive substantiation of the received view that **P**$\neq$**NP**.

I'm greatly indebted to Michael Zenzen for many valuable discussions about the **P=?NP** problem and physics (*simpliciter* and digital), and to Jim Fahey for discussions about such physics and mixed-mode dual-diamond operators in modal logic. The presentation of the core arguments herein to editions of Bringsjord's graduate seminar, *Logic & Artificial Intelligence*, and his guest lectures on **P=?NP** in *Formal Foundations of Cognitive Science* graduate seminars, sparked a number of helpful objections and suggestions, for which I'm grateful. I'm indebted as well to two anonymous referees for trenchant comments. Though the two arguments herein (the second of which seems to establish **P=NP**) are for weal or woe Bringsjord's, Joshua Taylor's astute objections catalyzed much thought and crucial refinements.

Selmer Bringsjord
Selmer.Bringsjord@gmail.com;
http://www.rpi.edu/~brings

1 Department of Cognitive Science, Rensselaer AI and Reasoning (RAIR) Lab, Rensselaer Polytechnic Institute (RPI), Troy, NY 12180, USA

2 Department of Computer Science, Rensselaer AI and Reasoning (RAIR) Lab, Rensselaer Polytechnic Institute (RPI), Troy, NY 12180, USA

Springer