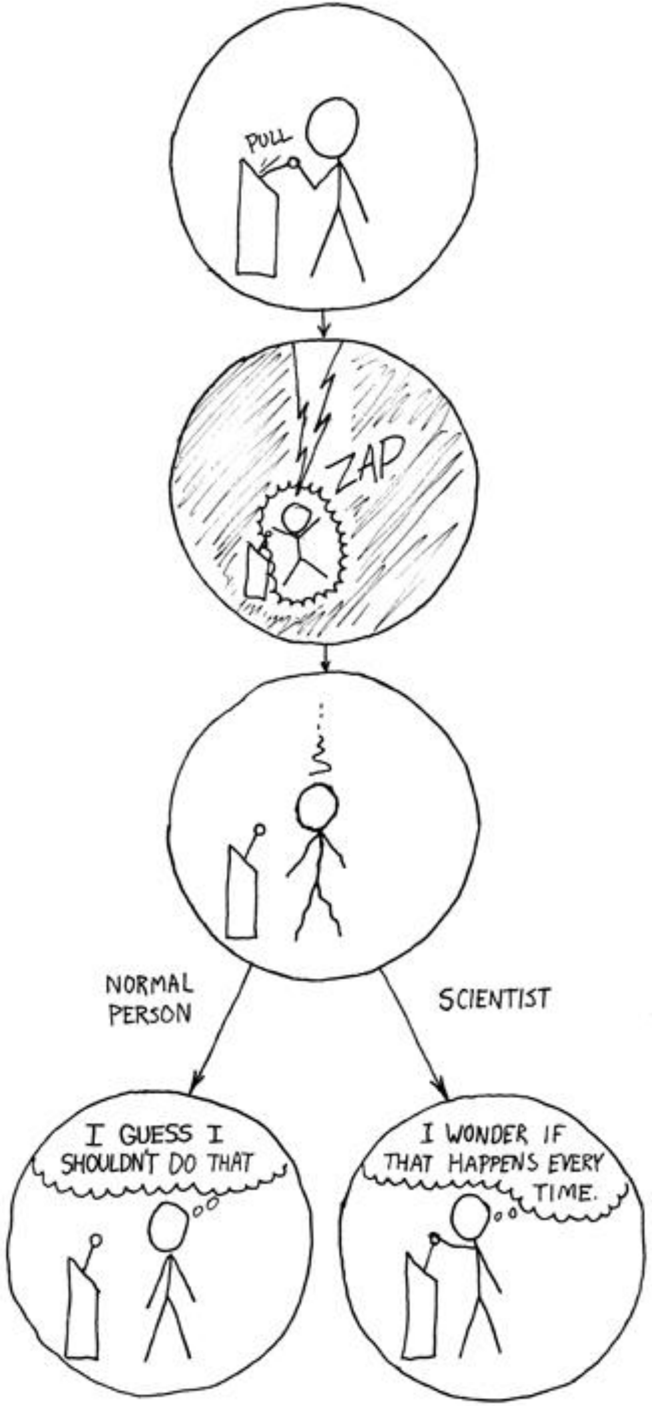
SCIENCE
PASSION
TECHNOLOGY

Logic and Computability

Predicate Logic
aka. First-Order Logic

Bettina Könighofer
bettina.koenighofer@iaik.tugraz.at

Stefan Pranger
stefan.pranger@iaik.tugraz.at



https://xkcd.com/2497/

Time Line - Topics

March → April → Mai → June

Propositional Logic

- Syntax & Semantic
- Decide Satisfiability (DPLL)
- **Data structures (BDDs)**
- Natural Deduction
  - Perform proofs
- **Equivalence Checking / Normal forms**
- Introduction to Z3

Predicate Logic

- Syntax & Semantic
- Natural deduction
  - Perform proofs
- **Satisfiability Modulo Theory** (SMT)
- Decide Satisfiability (DPLL(T))
- Solving Problems via Z3

Limitations of Propositional Logic

- Example: Model the following sentence in propositional logic:

  "Every person which is 18 years or older is eligible to vote."

- Solution: $\varphi \coloneqq p$
  - p … "Every person which is 18 years or older is eligible to vote."

- Prop. logic cannot express **quantified variables**
  - E.g.: "for all cats…"
  - "there exists a person…"

- We need a more powerful logic → Predicate Logic / First-Order Logic

Plan for Today

First Part – Predicate Logic

- Modelling Sentences
- Syntax
- Semantics, Models
  - Models
  - Satisfiability & Validity

Second Part – Z3

- Introduction to SMT solver Z3 – Part 2
- Focus on solving formulas in predicate logic

Learning Outcomes

After this lecture…

1. students can model declarative sentences with predicate logic.
2. students can explain the syntax and semantics of predicate logic.
3. students can explain what models in predicate logic are and what components they define.
4. for a given model, students can compute the semantics of a formula in predicate logic.

Formulas in Predicate Logic

$$\forall x \exists y . P(x, f(x, y))$$

Variables

- Variables over arbitrary **domains**
  - Integers, Reals, People…

Formulas in Predicate Logic

$$\forall x \exists y . P(x, f(x, y))$$

Predicates

- A predicate maps variables from an arbitrary domain to a **Boolean** value.
- Denoted by capital roman letters.
- Example: $P(x, y)$ … Returns true if x is smaller than y

Formulas in Predicate Logic

$$\forall x \exists y . P (x , f(x, y))$$

Functions

- A function maps variables from an arbitrary domain to a **value** in that **domain**
- Denoted by lowercase roman letters.
- Example: $f(x, y)$ …Returns the sum of x and y

Formulas in Predicate Logic

$$\forall x \exists y. P(x, f(x,y))$$

Universal Quantification

- $\forall x\, P(x)$ … $P(x)$ is true for **all possible values** of x in a particular domain
- $\forall x\, P(x)$ is read as “for all x $P(x)$”

Formulas in Predicate Logic

$$\forall x \exists y . P\left(x ,\ f(x,y) \right)$$

Existential Quantification

- $\exists x\ P(x)$ … $P(x)$ is true **for at least one value** of x in the domain.
- $\exists x\ P(x)$ is read as "there exists an x such that $P(x)$"

Modelling Sentences in Predicate Logic

- “Not all birds can fly”

Modelling Sentences in Predicate Logic

- "Not all birds can fly"
  - $A = \{\text{birds}\}$
  - Predicates:
    - Fly(x) … Returns true if x can fly
  - $\neg\forall \boldsymbol{x}.\,(\boldsymbol{Fly}(\boldsymbol{x}))$

Modelling Sentences in Predicate Logic

- "All integers are either even or odd."

Modelling Sentences in Predicate Logic

- "All integers are either even or odd."
  - $A = \mathbb{Z}$
  - Predicates:
    - Even(x) … Returns true if x is even
    - Odd(x) … Returns true if x is odd

$$\forall \boldsymbol{x}.\,(\boldsymbol{Even}(\boldsymbol{x}) \oplus \boldsymbol{Odd}(\boldsymbol{x}))$$

Modelling Sentences in Predicate Logic

- “Alice has no sister.”

Modelling Sentences in Predicate Logic

- “Alice has no sister.”

  - $A = \{\text{people}\}$
  - Predicates:
    - $Alice(x)$ … x is Alice
    - $Sister(x)$ … x has a sister

$$\forall \boldsymbol{x}(\boldsymbol{Alice}(\boldsymbol{x}) \rightarrow \neg \boldsymbol{Sister}(\boldsymbol{x}))$$

Modelling Sentences in Predicate Logic

- “Any person that wears a crown is either a king or a queen.”

Modelling Sentences in Predicate Logic

- "Any person that wears a crown is either a king or a queen."

  - $A = \{\text{people}\}$
  - Predicates:
    - $WearsCrown(x) \ldots x\ wears\ a\ crown$
    - $King(x) \ldots x\ is\ a\ king$
    - Queen(x) … x is a queen

      - $\forall \boldsymbol{x}\ (\boldsymbol{WearsCrown}(\boldsymbol{x}) \rightarrow (\boldsymbol{King}(\boldsymbol{x}) \vee \boldsymbol{Queen}(\boldsymbol{x}))$

Modelling Sentences in Predicate Logic

- "For any two integers it holds that their sum is smaller than their product."

$Note$:
+ and · are functions, and
< is a predicate.

Modelling Sentences in Predicate Logic

- "For any two integers it holds that their sum is smaller than their product."
  - $A = \mathbb{Z}$
  - x + y ... returns the sum of x and y
  - x · y ... returns the sum of x and y
  - x < y returns true if x is smaller than y

Note:
\+ and · are functions, and
< is a predicate.

$$\forall \boldsymbol{x} \forall \boldsymbol{y}((\boldsymbol{x} + \boldsymbol{y}) < (\boldsymbol{x} \cdot \boldsymbol{y}))$$

Modelling Sentences in Predicate Logic

- "Every even integer greater than 2 is equal to the sum of two prime numbers."

(Goldbach's Conjecture)

Modelling Sentences in Predicate Logic

- "Every even integer greater than 2 is equal to the sum of two prime numbers."

(Goldbach's Conjecture)

  - $A = \mathbb{Z}$
  - E(x) ... true if x is even
  - G(x) ... true if x is greater than 2
  - P(x) ... true if x is prime
  - x = y ... true if x is equal to y
  - x + y ... returns the sum of x and y

$$\forall x\ (E(x) \land G(x) \rightarrow \exists a, b(P(a) \land P(b) \land (x = a + b)))$$

Plan for Today

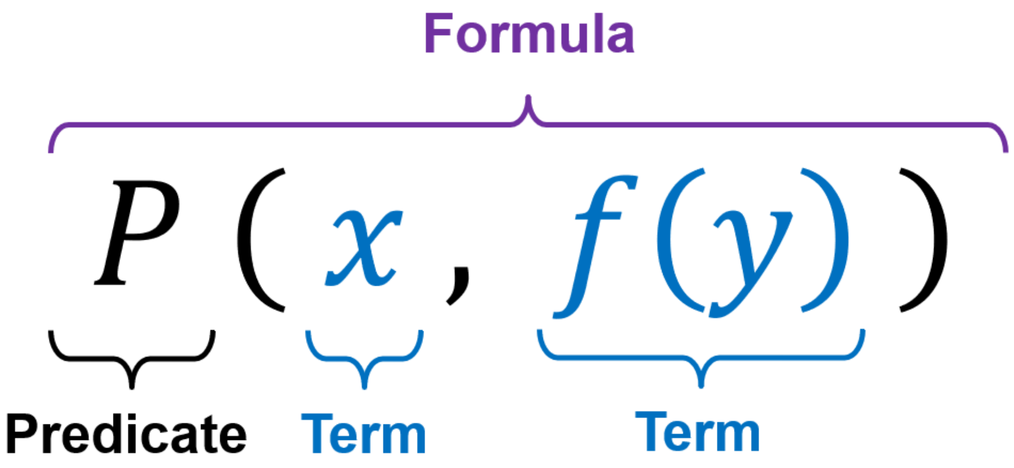
First Part – Predicate Logic

- Modelling Sentences
- **Syntax**
- Semantics, Models
  - Models
  - Satisfiability & Validity

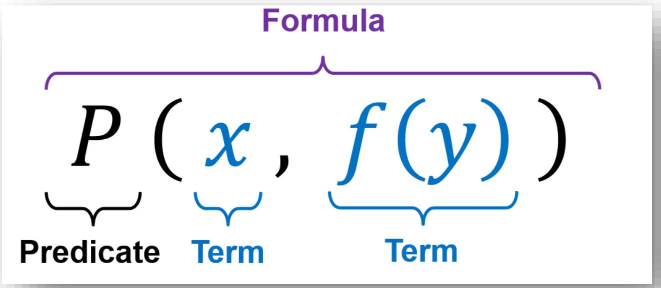
Syntax of Predicate Logic

Two types of *sorts:*

- **Terms**
  - Refer to **objects** of the **domain**:
    - *constants* represent individual objects, e.g., Alice, Bob, 5, 3, 3.45…
    - *variables* like x, y represent objects
    - *functions symbols* refer to objects like $x \cdot y, f(x)$ …
- **Formulas**
  - Have a truth value
  - E.g., $x \cdot y == 1$ is a formula



Syntax of Predicate Logic - Notation

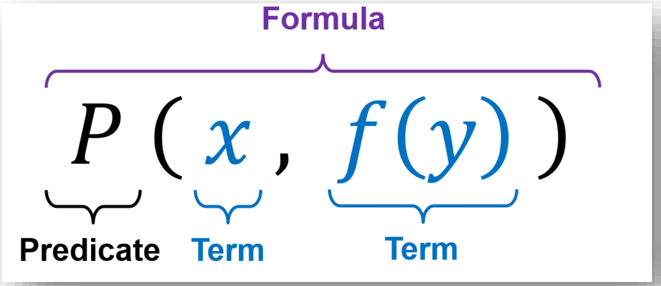


- Set of variable symbols $\mathbb{V}$
  - E.g., $x, y, z, \ldots$

- Set of function symbols $\mathbb{F}$
  - $f, g, h, \ldots$ (arity > 0)
  - constants (arity = 0)

- Set of predicate symbols $\mathbb{P}$
  - $P, Q, R, \ldots$ (arity > 0)
  - Prop. constants (arity = 0)

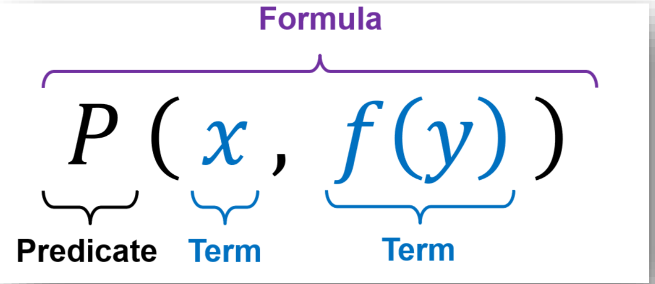
Syntax of Predicate Logic - Terms



- Recursive Definition
  - Any **variable** $\mathrm{v} \in \mathbb{V}$ is a term.
  - If $c \in \mathbb{F}$ is a **nullary function**, then c is a term.
  - Given terms $\boldsymbol{t_1}, \boldsymbol{t_2}, \ldots, \boldsymbol{t_n}$, and an n-ary function symbol $\boldsymbol{f} \in \mathbb{F}$, then

    $\boldsymbol{f}(\boldsymbol{t_1}, \boldsymbol{t_2}, \ldots, \boldsymbol{t_n})$ is a term.

Syntax of Predicate Logic - Formulas



Recursive Definition

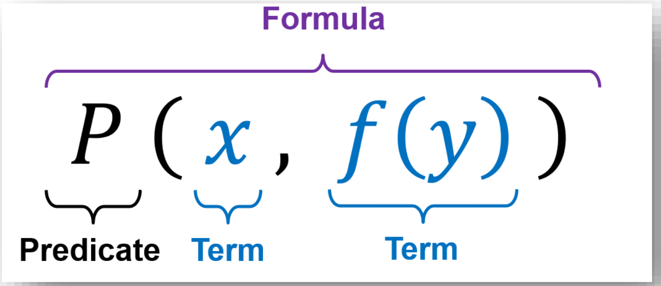
- Given terms $\boldsymbol{t_1, t_2, \dots, t_n}$, and an n-ary predicate symbol $\boldsymbol{P} \in \mathbb{P}$, then

  $$\boldsymbol{P(t_1, t_2, \dots, t_n)} \text{ is a formula.}$$

- If φ and ψ are formulas, then
  $\neg\varphi, (\varphi \land \psi), (\varphi \lor \psi)$, and $(\varphi \rightarrow \psi)$ are formulas.

- If φ is a formula and $x \in \mathbb{V}$ is a variable, then
  $(\forall x\ \phi)$ and $(\exists x\ \phi)$ are formulas.

Binding Priorities



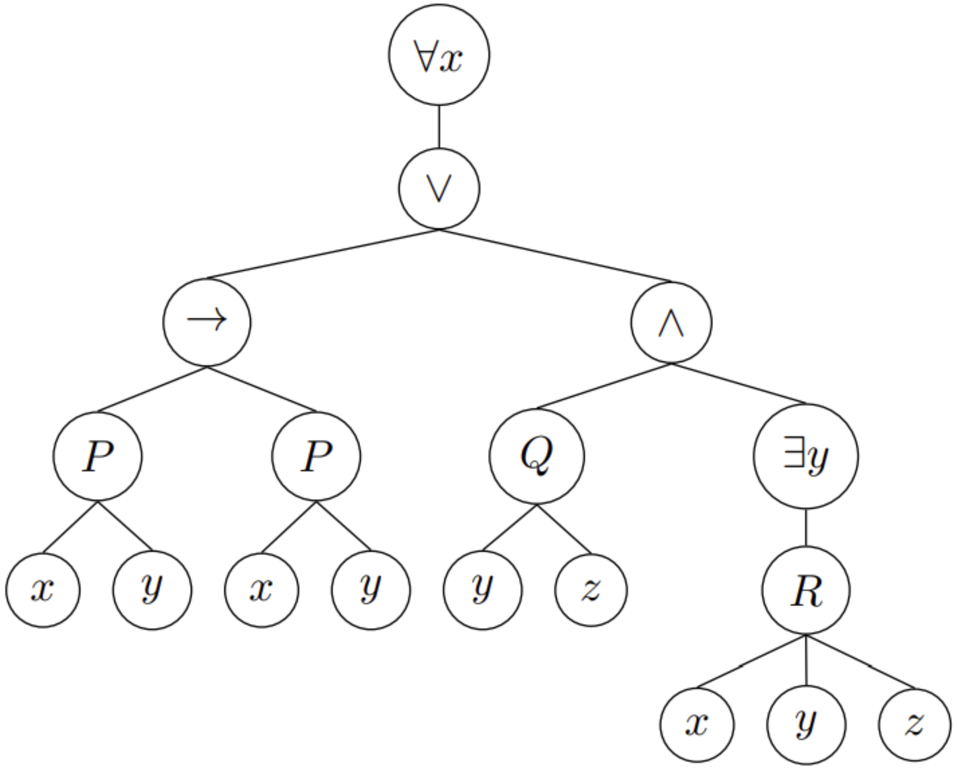
$\forall x$ and $\exists x$ are binding as strong as $\neg$

1. $\forall$, $\exists$, $\neg$
2. $\wedge$
3. $\vee$
4. $\rightarrow$
   - Right-associative

Syntax Tree

- Same as for formulas in prop. logic
- Additional sorts of nodes for quantifiers, functions, and predicates

- Example: Syntax tree for $\varphi \coloneqq \forall x((P(x, y) \rightarrow P(x, y)) \vee (Q(y, z) \wedge \exists y R(x, y, z)))$



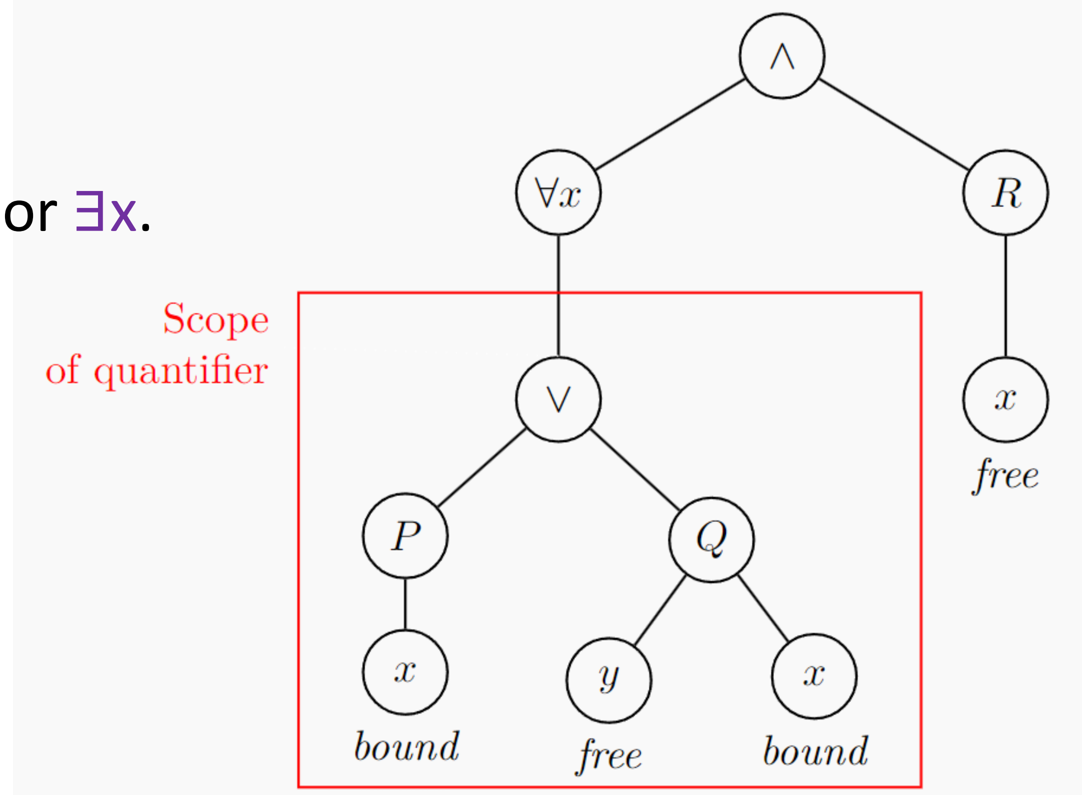
Free and Bound Variables

Scope of Quantifiers

- For a formula $\forall x\ \varphi$, it holds that φ is the scope of $\forall x$
- For a formula $\exists x\ \varphi$, it holds that φ is the scope of $\exists x$

Free and **bound** variables

- An instance of x in φ is called free if its node has no path upwards to any node labeled with ∀x or ∃x.
- Otherwise, the variable is called bound.



Example: Free and Bound Variables

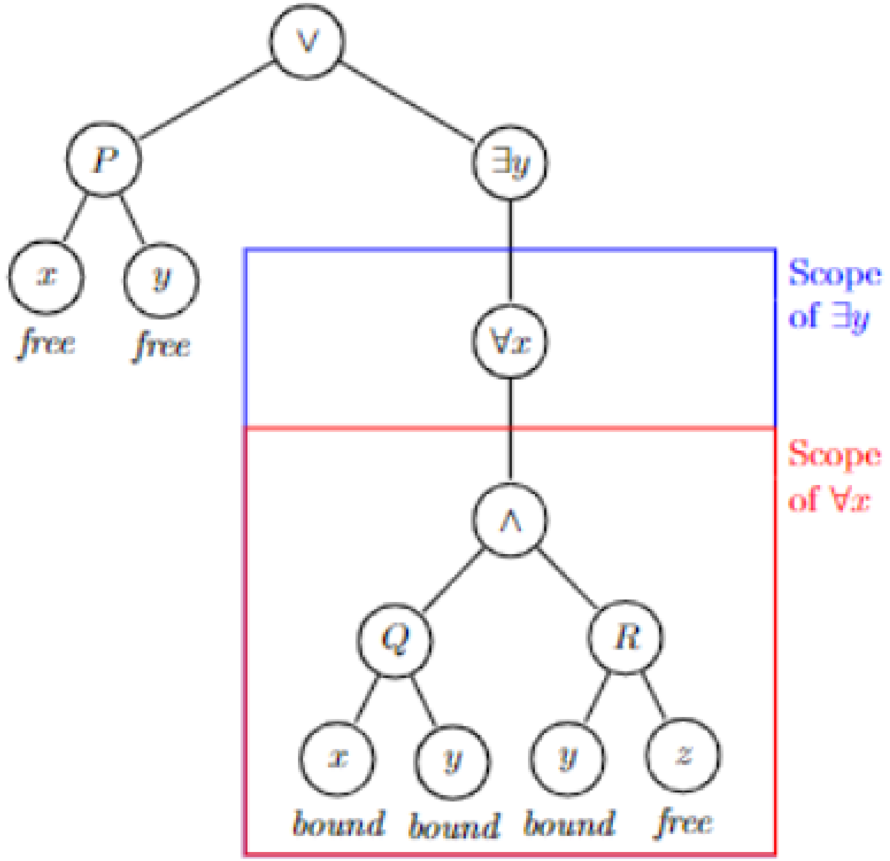
- Construct a *syntax tree* for φ. Determine the *scope* of its quantifiers. Indicate which variables are *free* and which are *bound*.

$$\varphi \coloneqq P(x, y) \vee \exists y \forall x (Q(x, y) \wedge R(y, z))$$

Example: Free and Bound Variables

- Construct a *syntax tree* for φ. Determine the *scope* of its quantifiers. Indicate which variables are *free* and which are *bound*.

$$\varphi \coloneqq P(x,y) \lor \exists y \forall x (Q(x,y) \land R(y,z))$$



Outline

- Modelling Sentences ✓
- Syntax ✓
- Semantics, Models
  - Models
  - Satisfiability & Validity

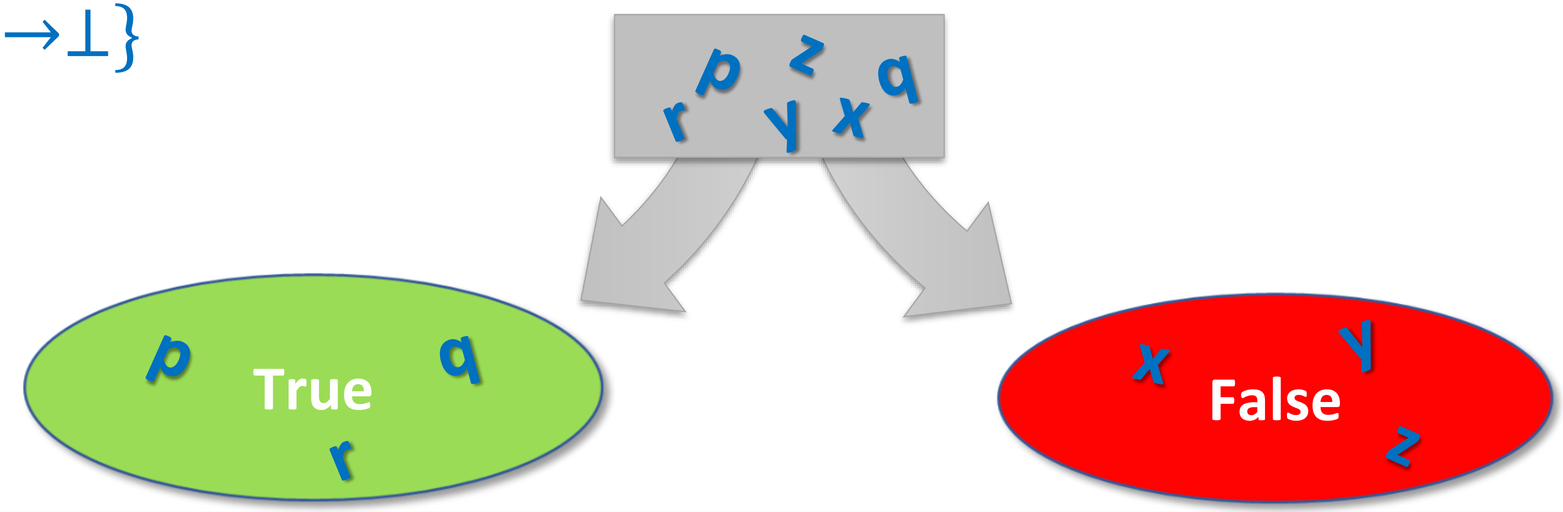
Recap: Model M for Formulas in Prop. Logic

We want to know if M satisfies φ (i.e., $M \vDash \varphi?$)

- What does M need to define?

Recap: Model M for Formulas in Prop. Logic

- Assignment: $\{Atomic\ propositions\} \mapsto \{\top, \bot\}$
- Example
  - $\varphi = (p \vee y \vee \neg r) \wedge (\neg x \vee \neg q \vee z)$
  - $\mathcal{M}: \{p \rightarrow \top, q \rightarrow \top, r \rightarrow \top,$
    $x \rightarrow \bot, y \rightarrow \bot, z \rightarrow \bot\}$



Recap: Model M for Formulas in Prop. Logic

- $\varphi^{\mathcal{M}}$ ...φ is evaluated under $\mathcal{M}$

- Satisfying Model: $\mathcal{M} \vDash \varphi$
  - $\mathcal{M}$ satisfies φ, or
  - φ evaluates to true under $\mathcal{M}$
  - Example
    - $\varphi = a \vee b$
    - $\mathcal{M}: \{a \rightarrow \top, b \rightarrow \bot\}$
    - $\mathcal{M} \vDash \varphi$ or $\varphi^{\mathcal{M}} = \top$

- Falsifying Model: $\mathcal{M} \nvDash \varphi$
  - $\mathcal{M}$ does not satisfies φ, or
  - φ evaluates to false under $\mathcal{M}$
  - Example
    - $\varphi = a \vee b$
    - $\mathcal{M}: \{a \rightarrow \bot, b \rightarrow \bot\}$
    - $\mathcal{M} \nvDash \varphi$ or $\varphi^{\mathcal{M}} = \bot$

Model M for Formulas in Predicate Logic

E.g., $\varphi \coloneqq S \land R(x) \land \forall x \exists y. P(x, f(x, y))$

We want to know if M satisfies φ (i.e., $M \vDash \varphi$?)

- What does M need to define?

- Models for predicate logic formulas need to define:
  - Domain of variables
  - Values for free variables
  - Values for nullary functions
  - Truth values for nullary predicates
  - Concrete instances for any function and predicate

Model M for Formulas in Predicate Logic

- Domain A
- For each nullary $f \in \mathbb{F}$: concrete element $f^M \in A$
- For each nullary $P \in \mathbb{P}$: true or false

- For each $f \in \mathbb{F}$ with arity $n > 0$: concrete function $f^M: A^n \to A$
  - Defined by e.g. function table

- For each $P \in \mathbb{P}$ with arity $n > 0$: concrete predicate $P^M \subseteq A^n$
  - Set of tuples which make P true

- For any free variable x : concrete value $x \to A$
  - Lookup table

Example: Models in Predicate Logic

- Give a model M for the following formula:

$$\varphi \coloneqq \exists x \forall y P(x, y)$$

- Model M:
  - $A = \{a, b\}$
  - $P^M \coloneqq \{(a, a), (a, b)\}$

Semantics of Predicate Logic

- We want to know if M satisfies φ
  - $M \vDash \varphi$?

Semantics of Predicate Logic

- We want to know if M satisfies φ
  - $M \vDash \varphi$?

- For φ of the form $P(t_1, t_2, \ldots, t_n)$
  - Interpret all terms $t_1, \ldots, t_n$ via M
  - Obtain $(a_1, a_2, \ldots, a_n)$ with $a_i \in A$
  - $M \vDash P(t_1, t_2, \ldots, t_n)$ iff $(a_1, a_2, \ldots, a_n) \in P^M$

Semantics of Predicate Logic

$[x \leftarrow a]$ means that x is mapped to a

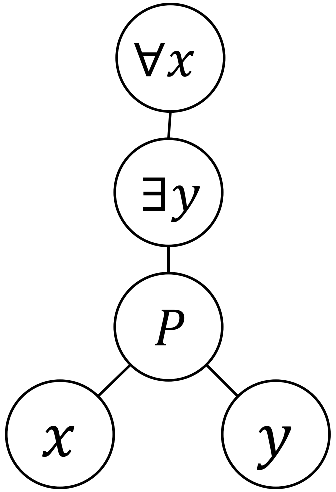
- For φ of the form $\forall x\ \psi$
  - $M \vDash \forall x\ \psi$ iff $M \vDash_{[x \leftarrow a]} \psi$, for **all** $a \in A$
- For φ of the form $\exists x\ \psi$
  - $M \vDash \exists x\ \psi$ iff $M \vDash_{[x \leftarrow a]} \psi$, for **at least one** $a \in A$
- For φ of the form $\neg\psi$, $\psi_1 \wedge \psi_2$, $\psi_1 \vee \psi_2$, $\psi_1 \rightarrow \psi_2$
  - Like in propositional logic

Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \varphi$?

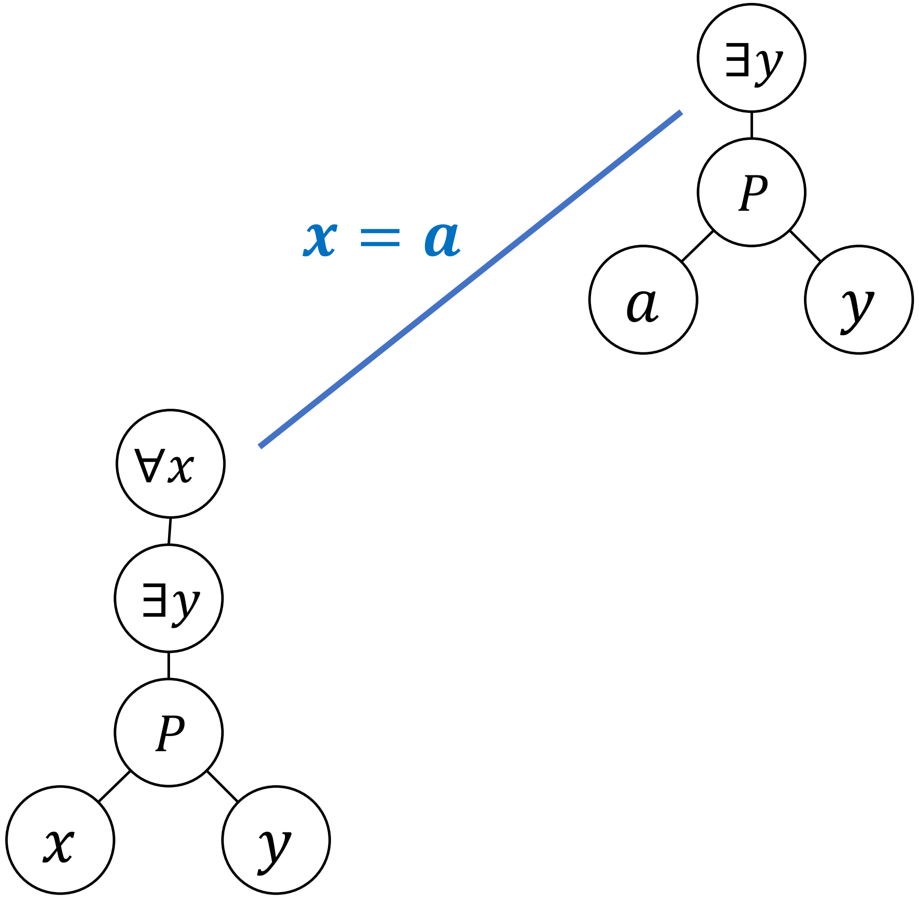
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \models \phi$?



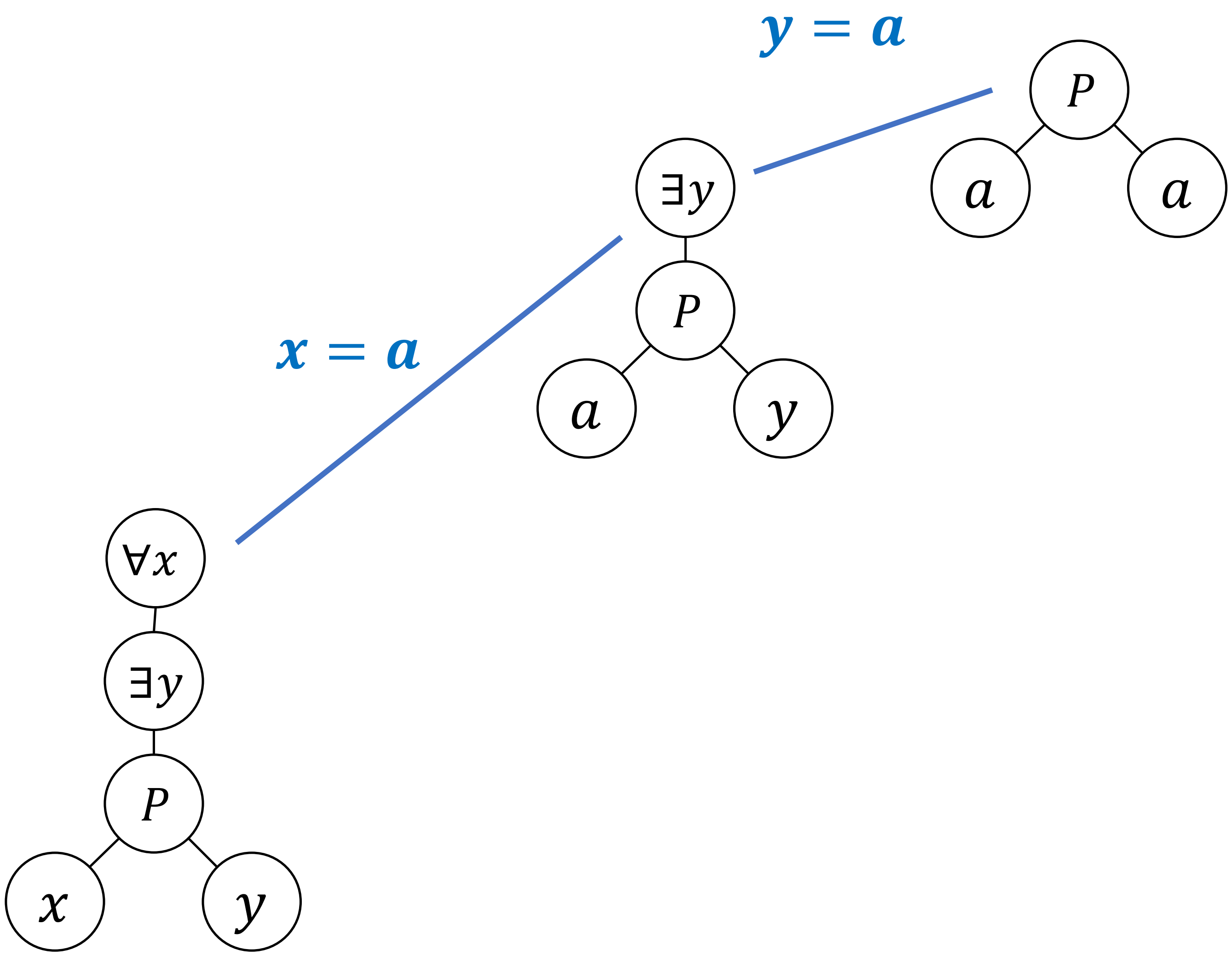
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



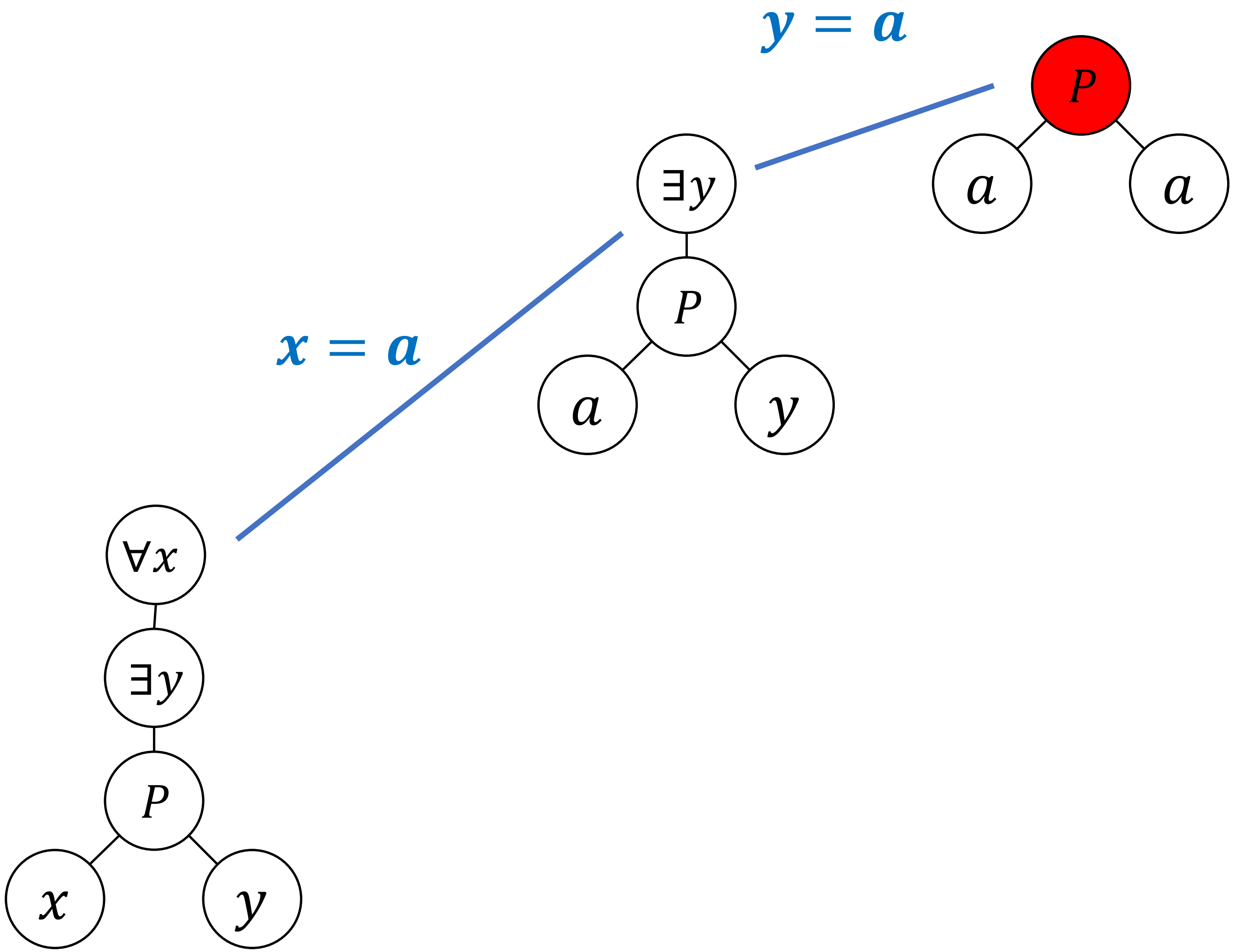
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



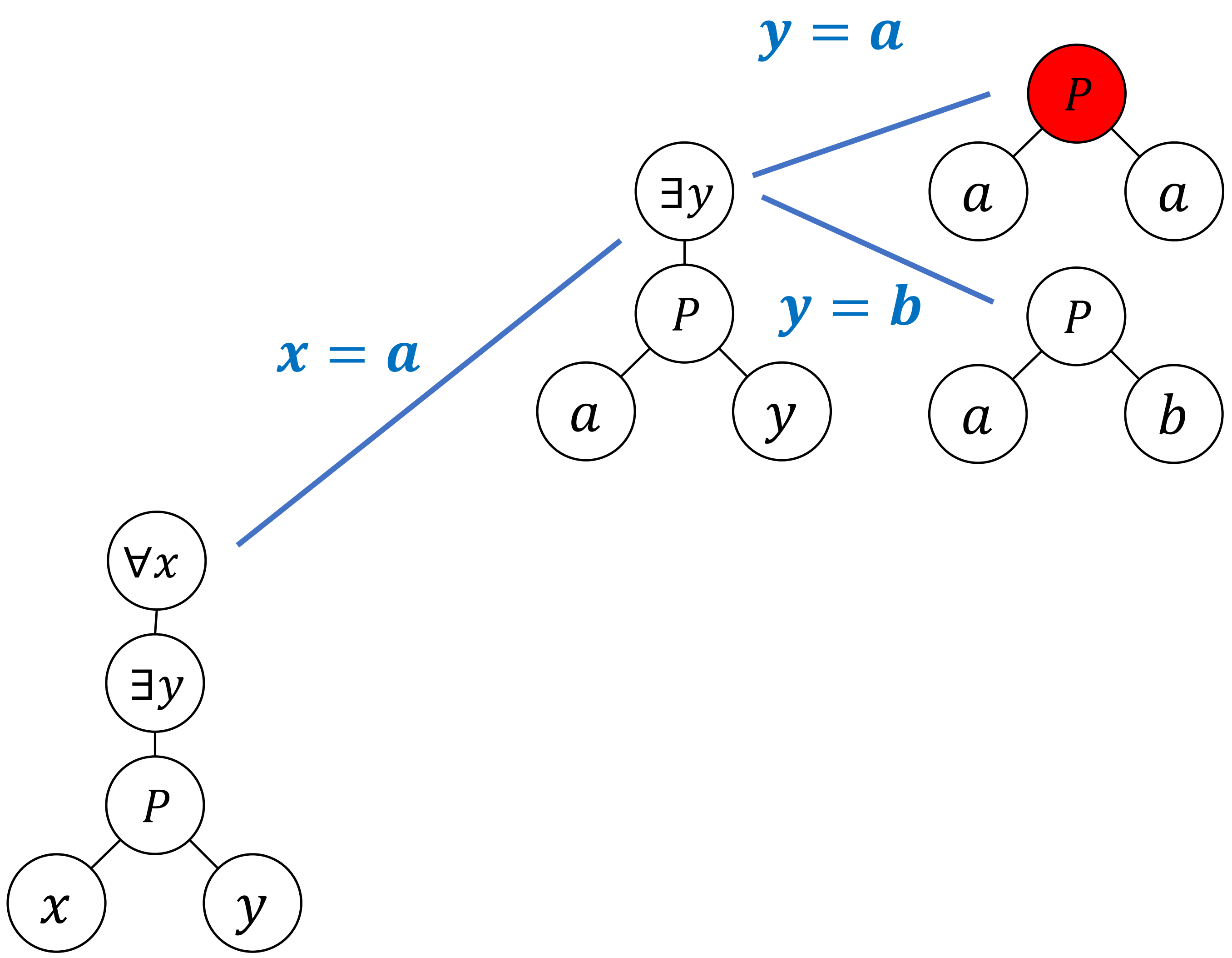
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



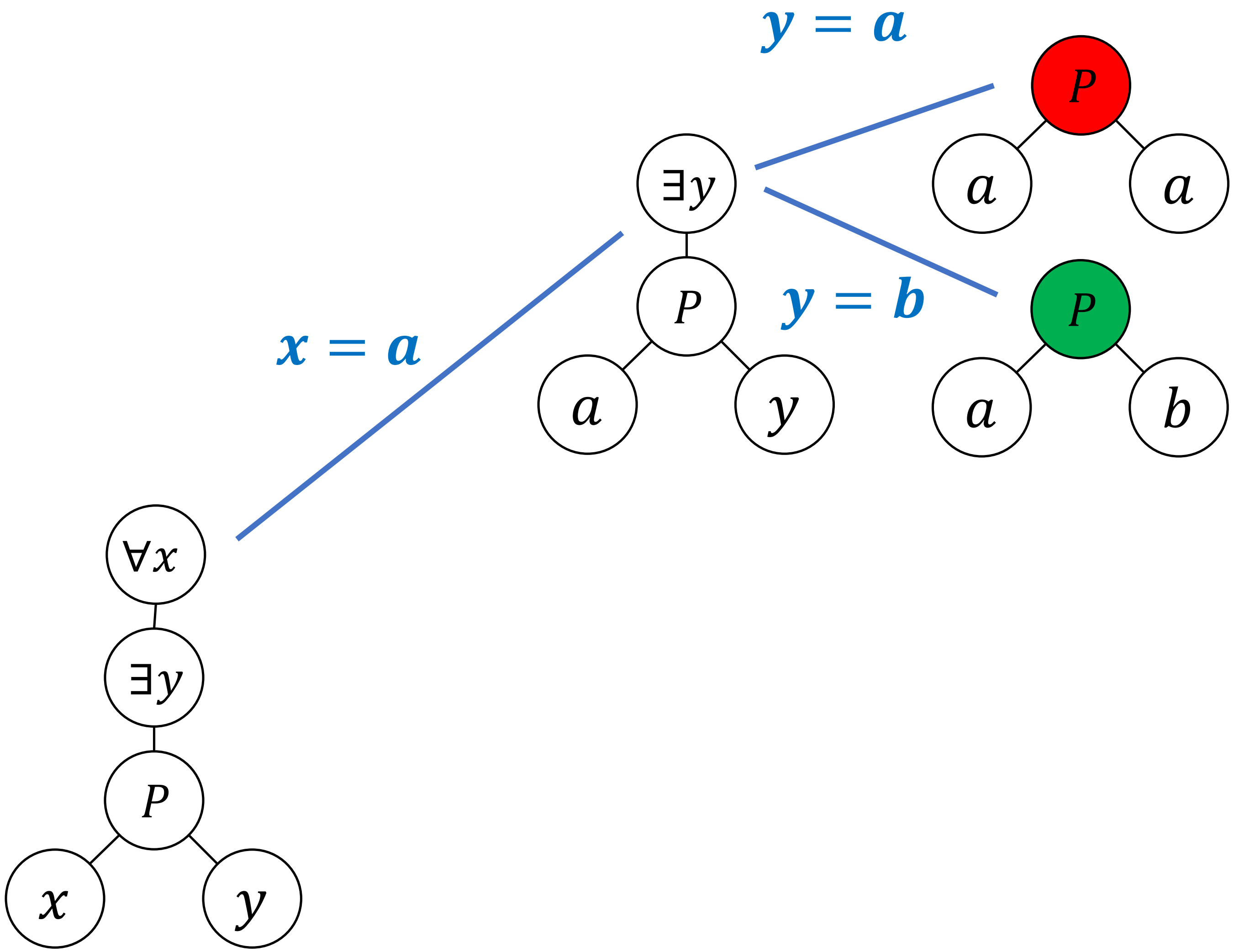
Evaluating a Model

- Given
  - $\phi = \forall x \exists y . P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \models \phi$?



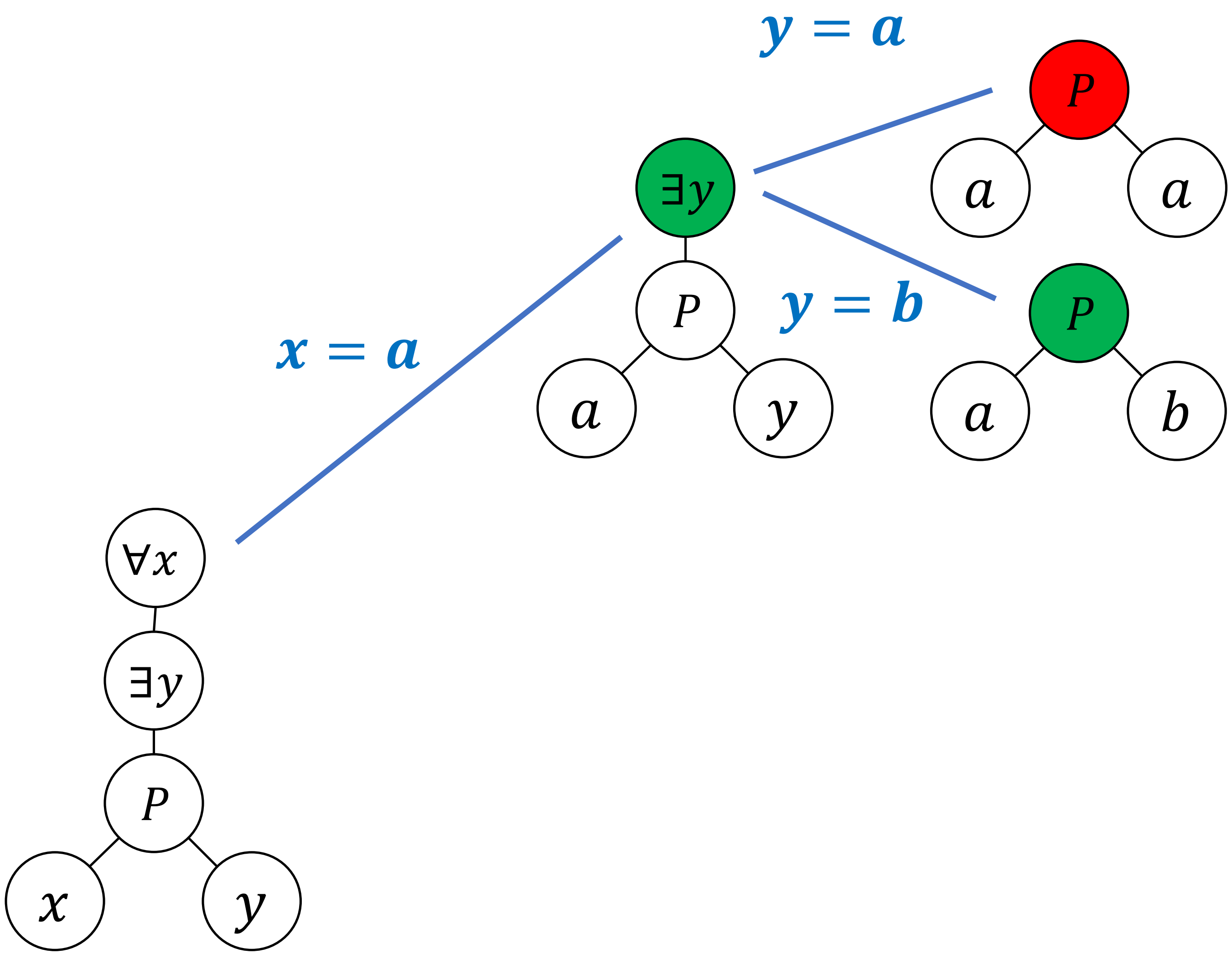
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



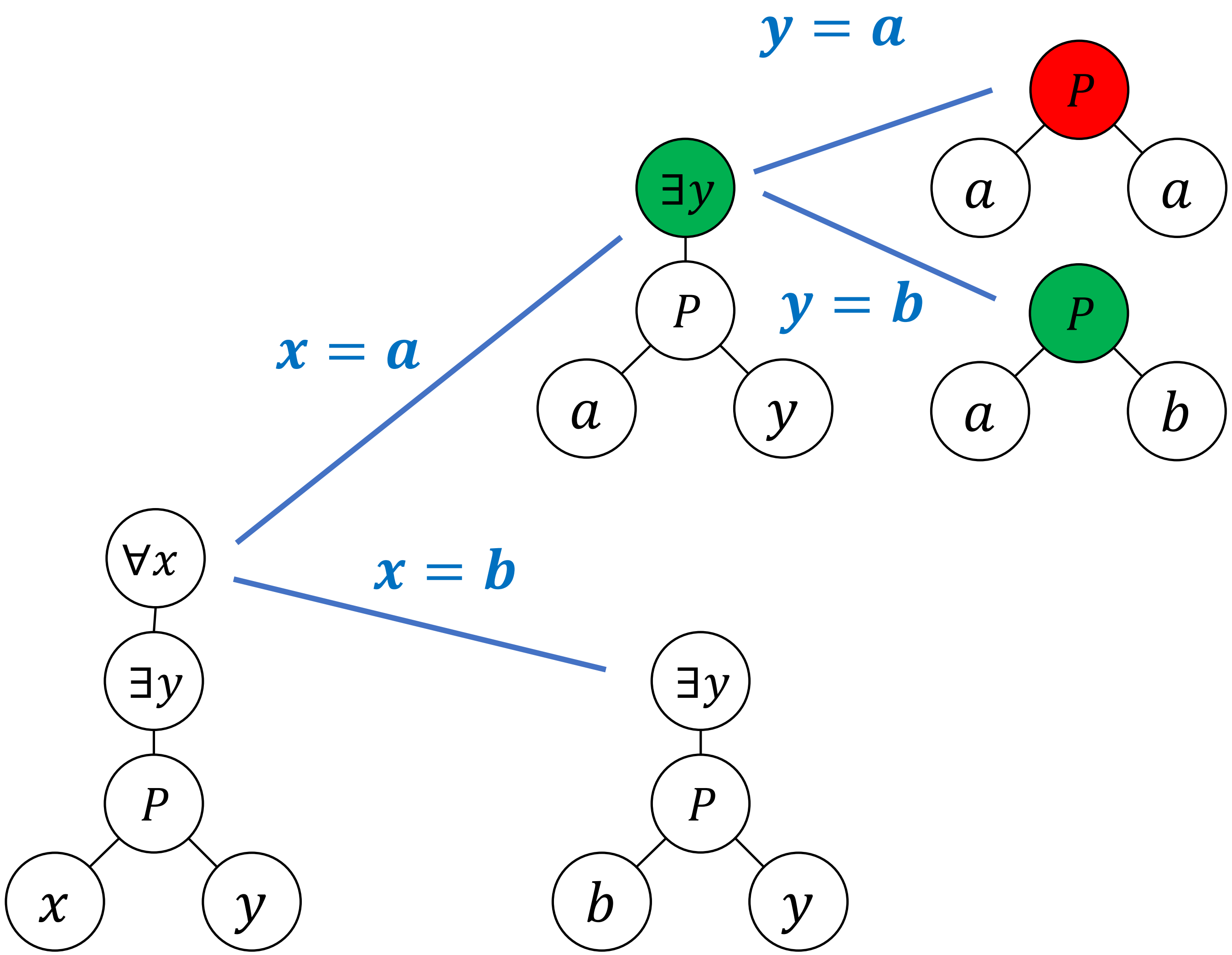
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



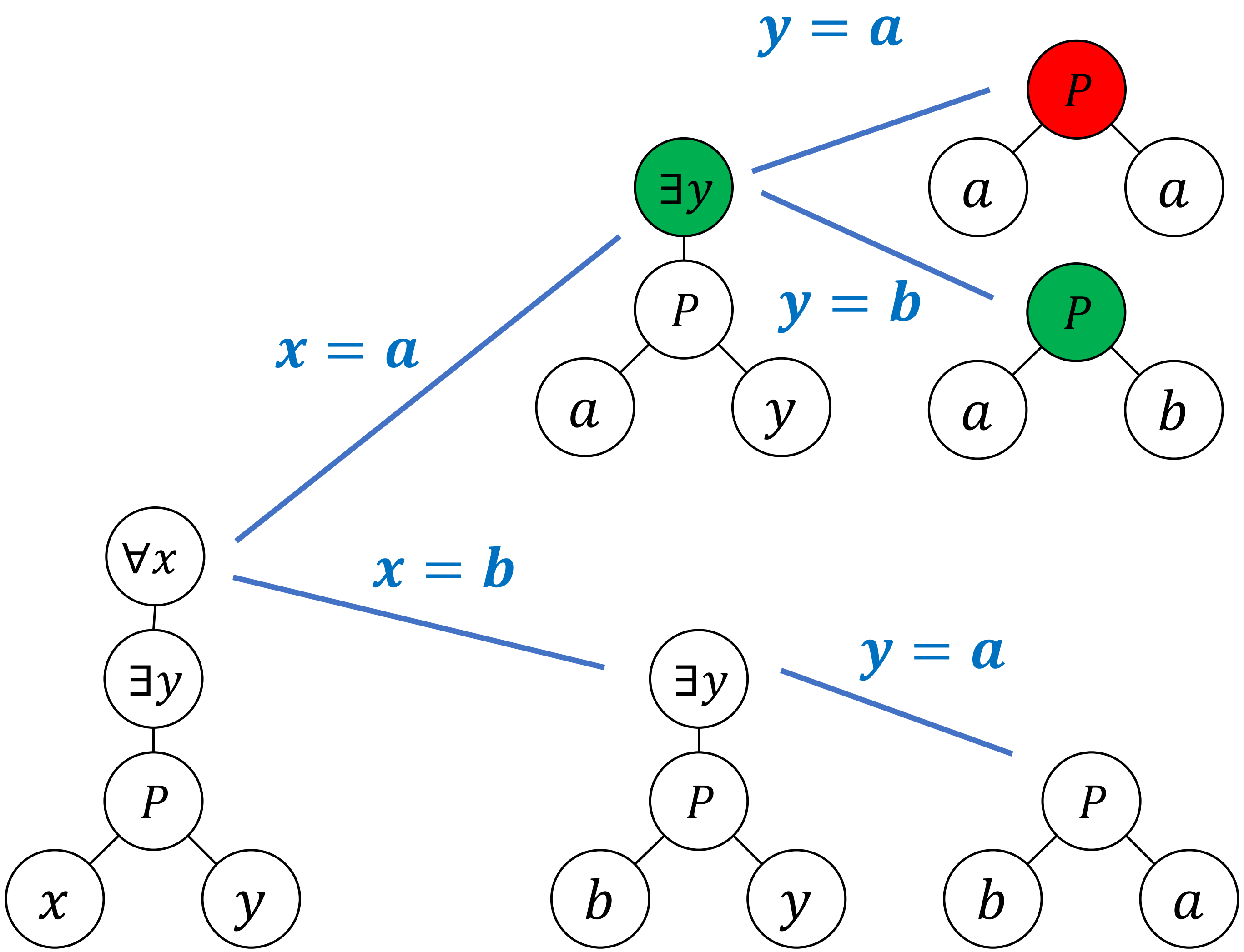
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



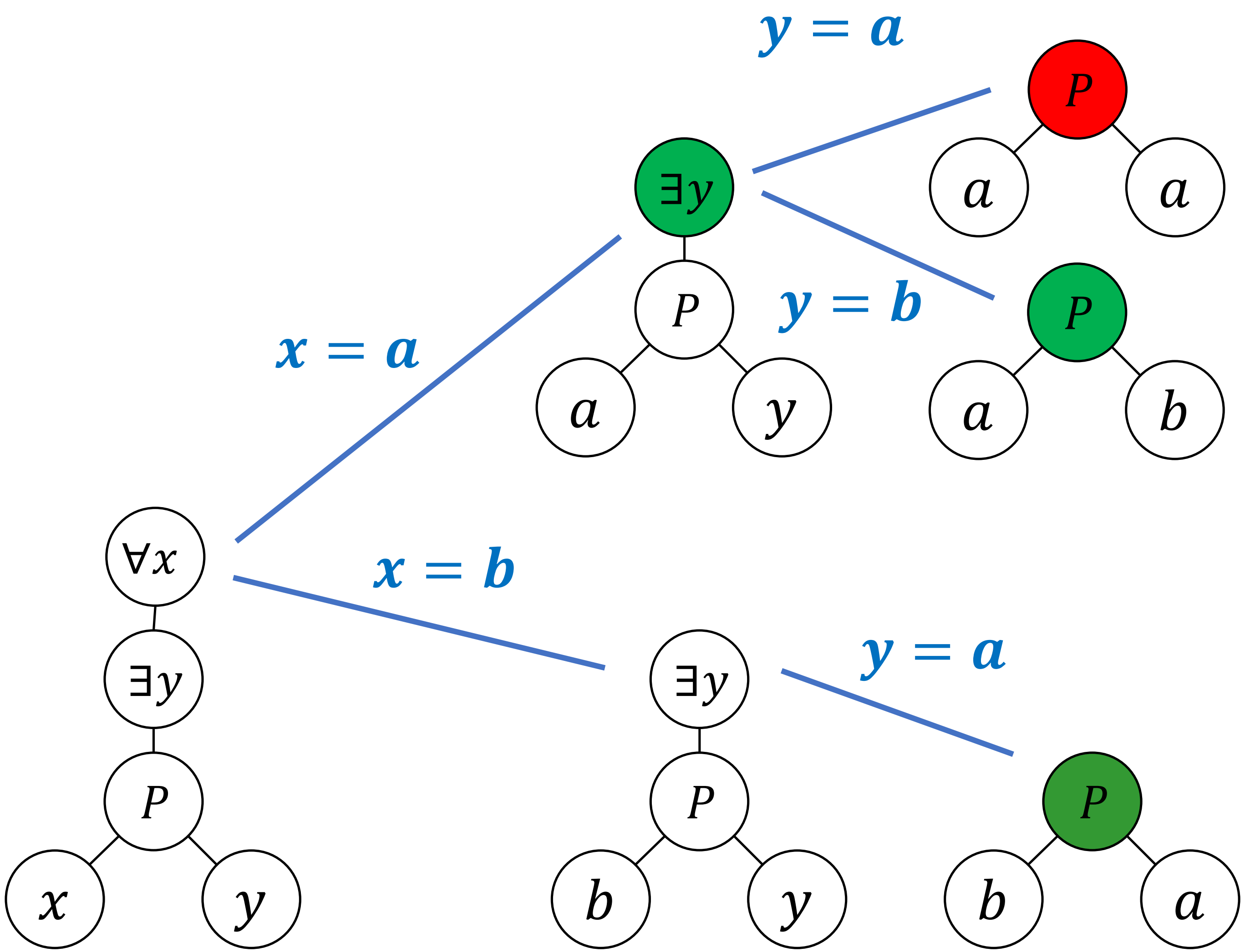
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



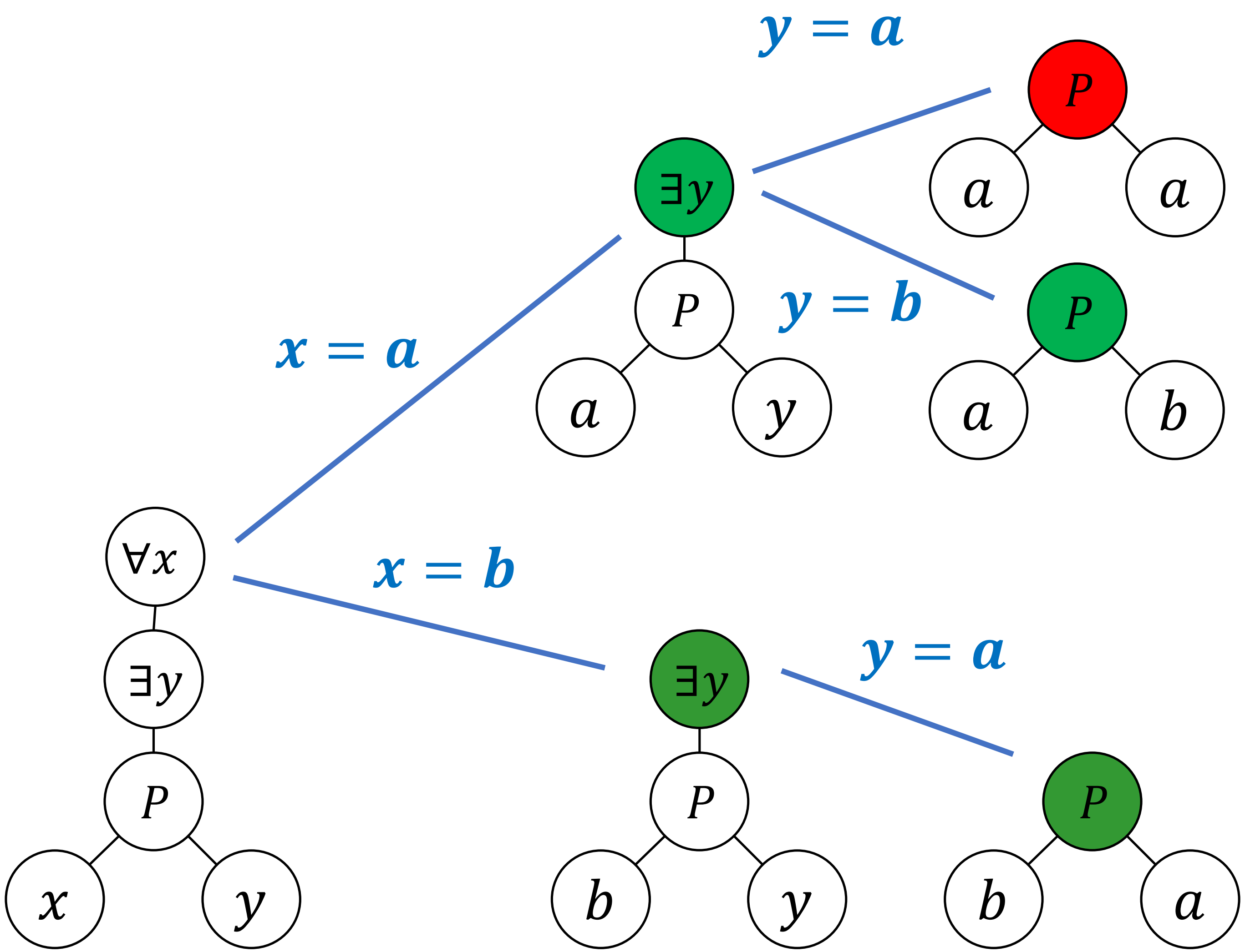
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



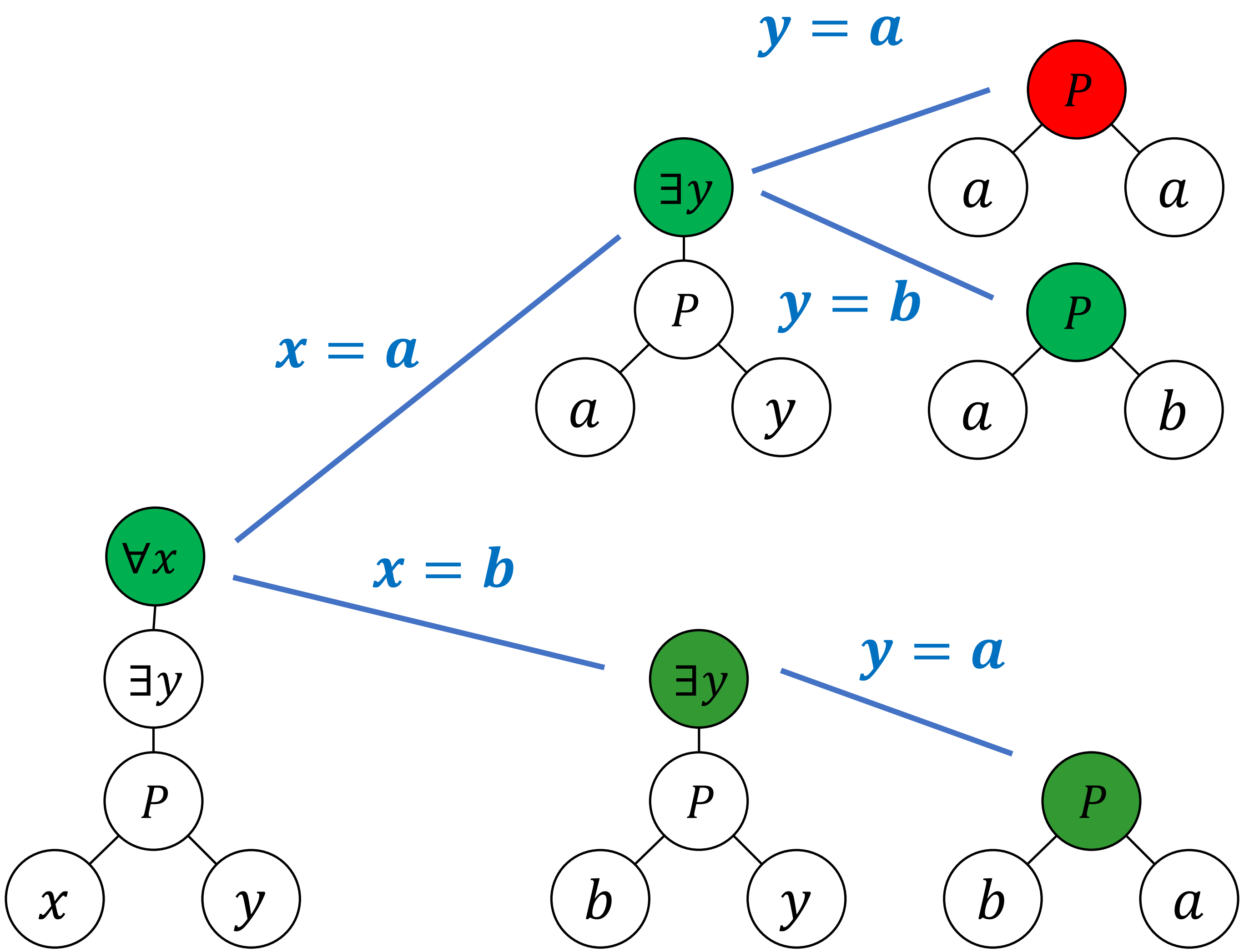
Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



Evaluating a Model

- Given
  - $\phi = \forall x \exists y. P(x, y)$
  - M:
    - $A = \{a, b\}$
    - $P^M = \{(a, b), (b, a)\}$
- $M \vDash \phi$?



Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \exists x\ \forall y\ (P(x,y) \rightarrow (Q(x,y) \vee R(x,y)))$$

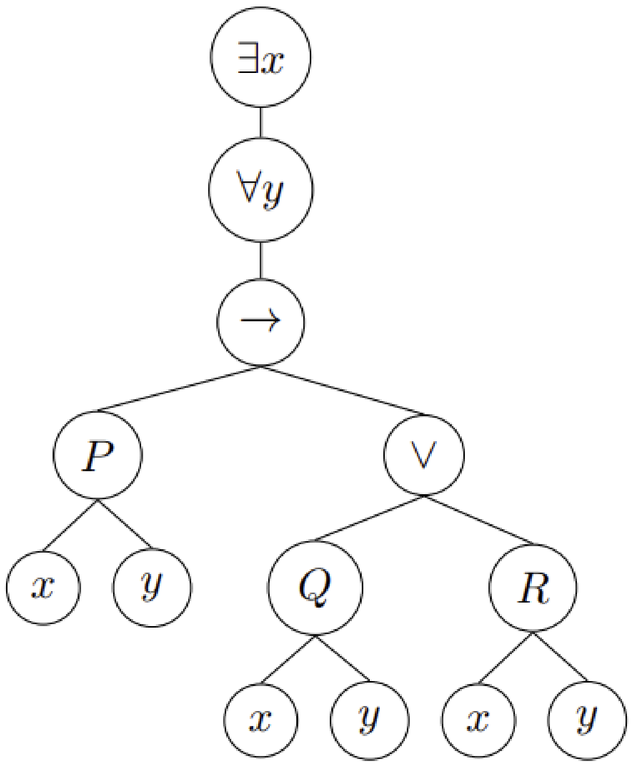
$$\mathcal{A} = \{a, b\}$$
$$P^{\mathcal{M}} = \{(a,a),(a,b)\}$$
$$Q^{\mathcal{M}} = \{(a,a),(b,a)\}$$
$$R^{\mathcal{M}} = \{(a,a),(b,b)\}$$

Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \exists x \, \forall y \, (P(x, y) \rightarrow (Q(x, y) \lor R(x, y)))$$

$$\mathcal{A} = \{a, b\}$$
$$P^{\mathcal{M}} = \{(a, a), (a, b)\}$$
$$Q^{\mathcal{M}} = \{(a, a), (b, a)\}$$
$$R^{\mathcal{M}} = \{(a, a), (b, b)\}$$

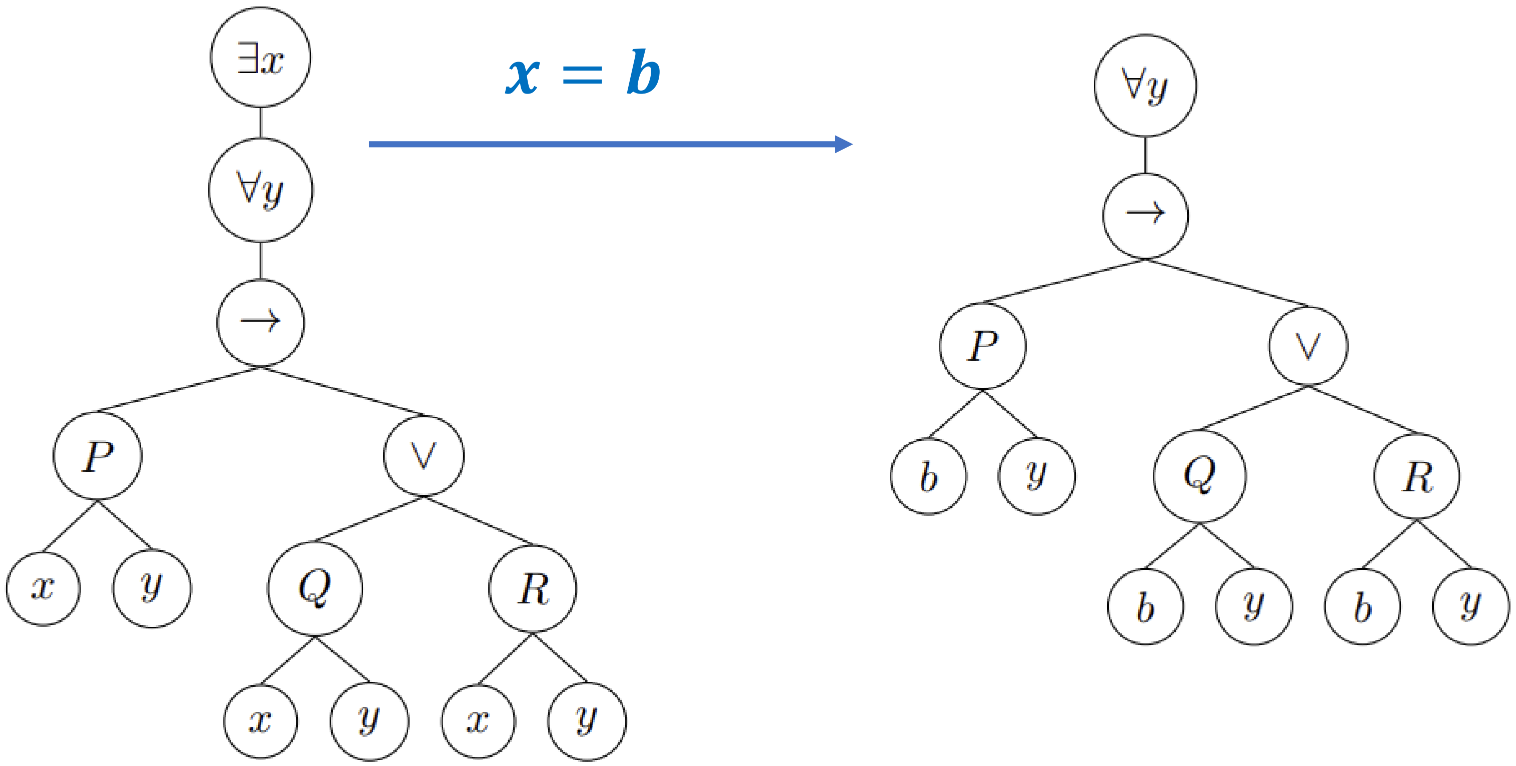


Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \exists x\ \forall y\ (P(x,y) \rightarrow (Q(x,y) \vee R(x,y)))$$

$$\mathcal{A} = \{a, b\}$$
$$P^{\mathcal{M}} = \{(a,a), (a,b)\}$$
$$Q^{\mathcal{M}} = \{(a,a), (b,a)\}$$
$$R^{\mathcal{M}} = \{(a,a), (b,b)\}$$

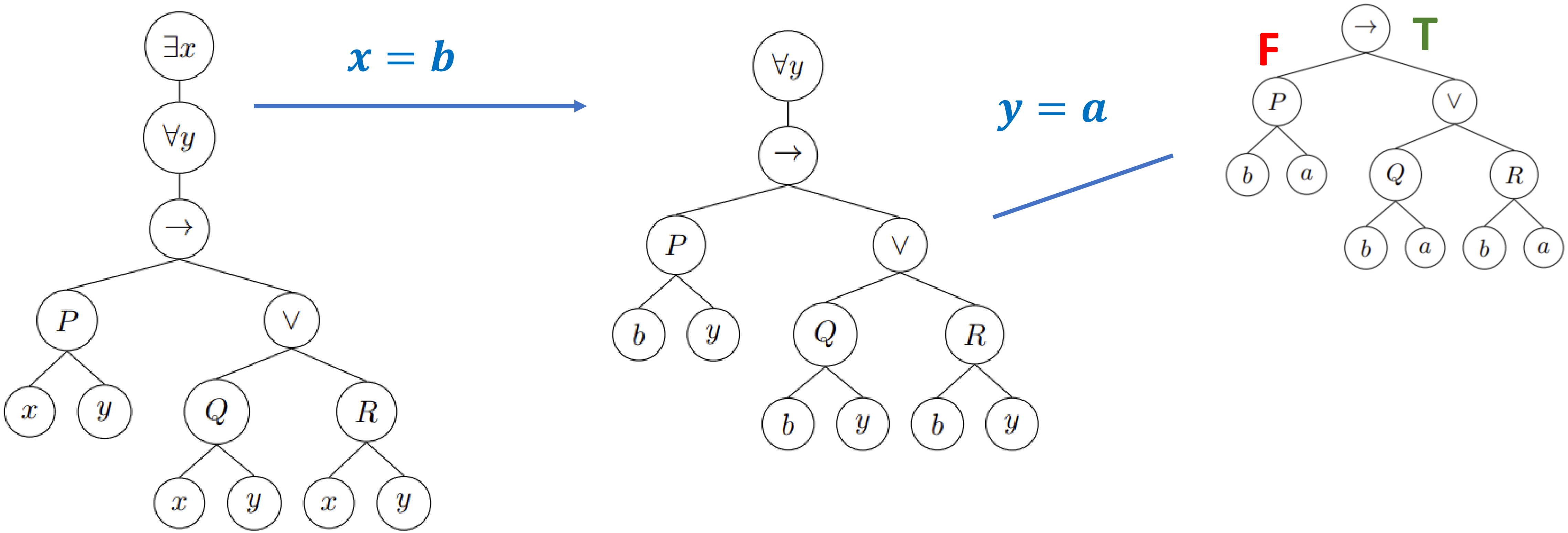


Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \exists x\ \forall y\ (P(x,y) \rightarrow (Q(x,y) \vee R(x,y)))$$

$$\mathcal{A} = \{a, b\}$$
$$P^{\mathcal{M}} = \{(a,a), (a,b)\}$$
$$Q^{\mathcal{M}} = \{(a,a), (b,a)\}$$
$$R^{\mathcal{M}} = \{(a,a), (b,b)\}$$

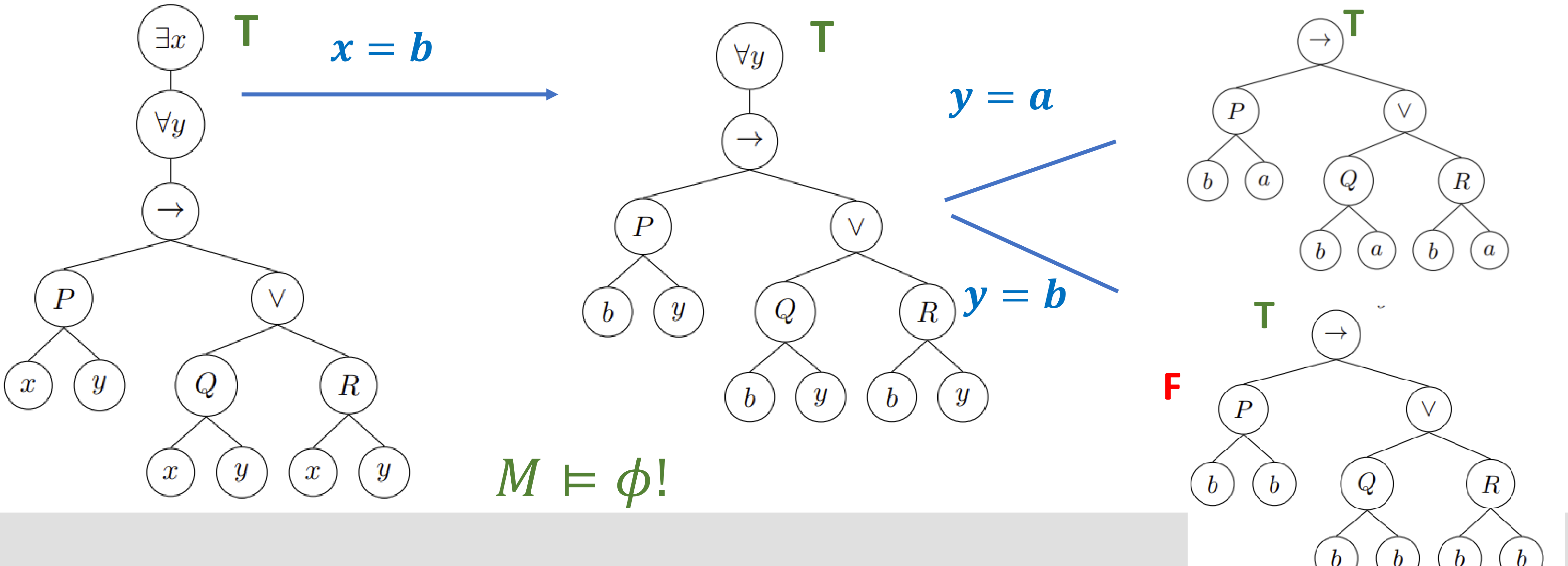


Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \exists x\, \forall y\, (P(x,y) \rightarrow (Q(x,y) \vee R(x,y)))$$

$$\begin{aligned} \mathcal{A} &= \{a, b\} \\ P^{\mathcal{M}} &= \{(a,a),(a,b)\} \\ Q^{\mathcal{M}} &= \{(a,a),(b,a)\} \\ R^{\mathcal{M}} &= \{(a,a),(b,b)\} \end{aligned}$$



Learning Outcomes

After this lecture…

1. students can model declarative sentences with predicate logic.
2. students can explain the syntax and semantics of predicate logic.
3. students can explain what models in predicate logic are and what components they define.
4. for a given model, students can compute the semantics of a formula in predicate logic.

Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \forall x \, \forall y \, ((P(x, f(y)) \land Q(y, z)) \rightarrow R(f(z)))$$

Model M:

Domain:

$$A = \{a, b\}$$

Definition of functions:

$$f^M(\mathrm{a}) = \mathrm{b}$$
$$f^M(\mathrm{b}) = \mathrm{a}$$

Definition of predicates:

$$P^M = \{(a, a), (a, b)\}$$
$$Q^M = \{(a, b)\}$$
$$R^M = \{b\}$$

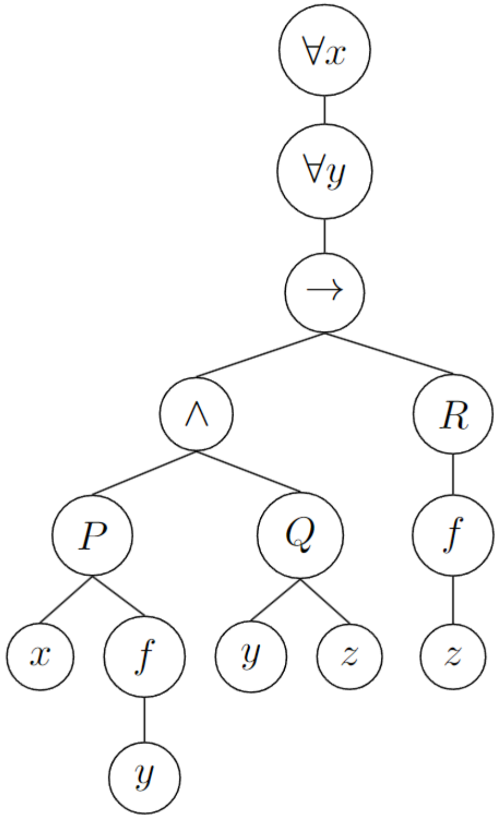
Lockup table for free variable:

$$z \rightarrow b$$

Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \forall x\, \forall y\, ((P(x, f(y)) \land Q(y, z)) \rightarrow R(f(z)))$$



Model M:

Domain:

$$A = \{a, b\}$$

Definition of functions:

$$f^M(\mathrm{a}) = \mathrm{b}$$
$$f^M(\mathrm{b}) = \mathrm{a}$$

Definition of predicates:

$$P^M = \{(a, a), (a, b)\}$$
$$Q^M = \{(a, b)\}$$
$$R^M = \{b\}$$

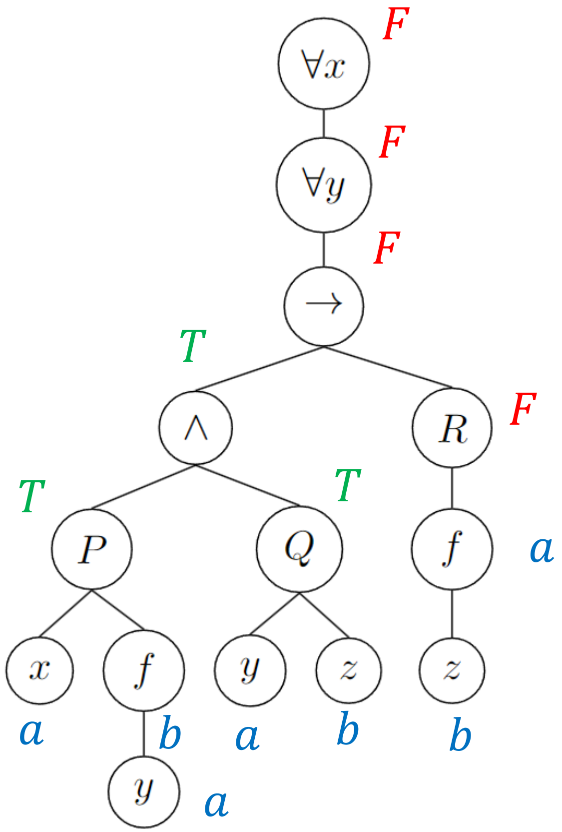
Lockup table for free variable:

$$z \rightarrow b$$

Evaluating a Model

Does the following model M satisfy the formula φ?

$$\varphi = \forall x\, \forall y\, ((P(x, f(y)) \land Q(y, z)) \to R(f(z)))$$



$M \not\models \varphi$

Model M:

Domain:

$$A = \{a, b\}$$

Definition of functions:

$$f^M(\mathrm{a}) = \mathrm{b}$$
$$f^M(\mathrm{b}) = \mathrm{a}$$

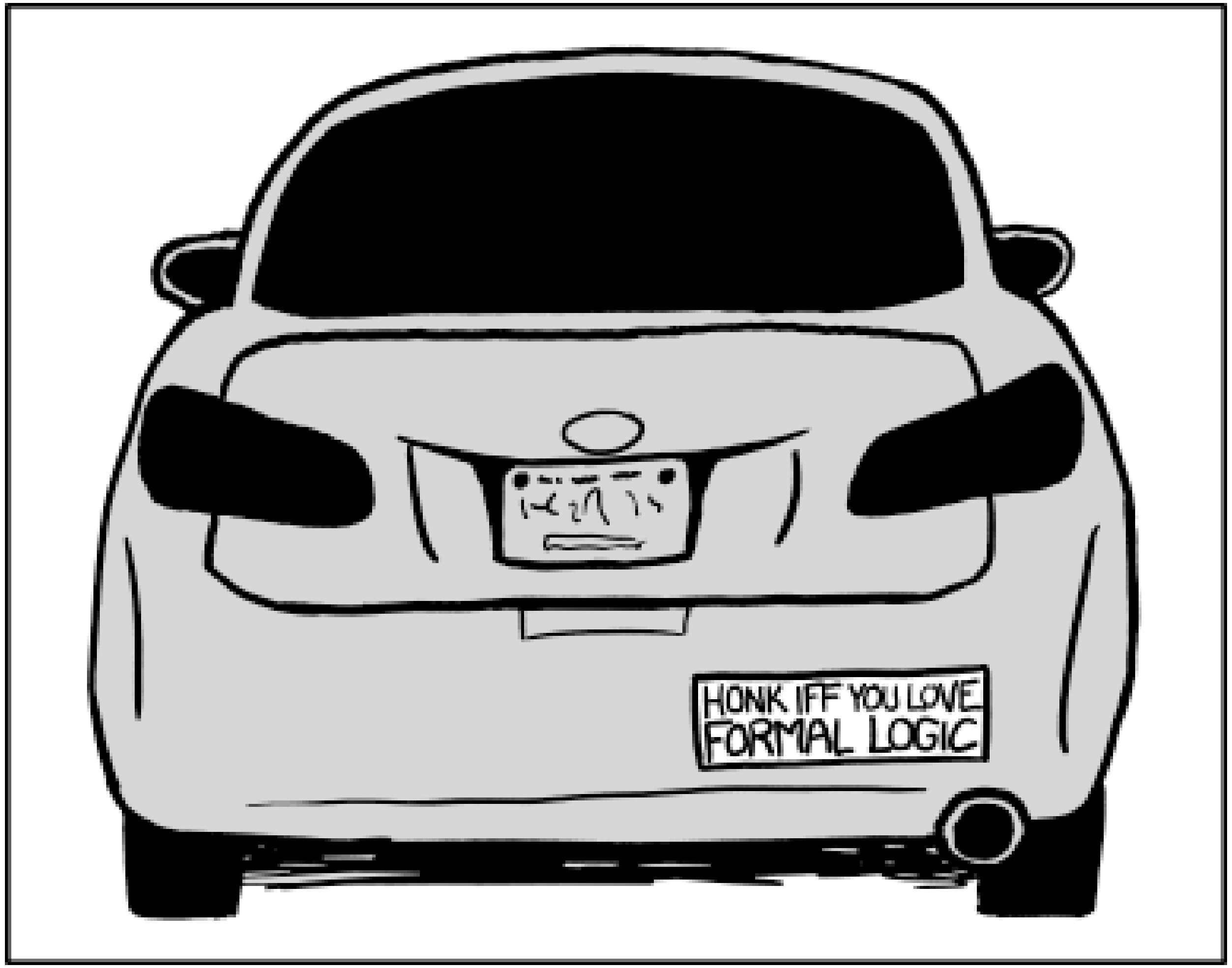
Definition of predicates:

$$P^M = \{(a, a), (a, b)\}$$
$$Q^M = \{(a, b)\}$$
$$R^M = \{b\}$$

Lockup table for free variable:

$$z \to b$$

Thank You



https://xkcd.com/1033/