14-12-16

Propositional logic

Predicate logic.

Applications $\rightarrow$ logic programming.

<u>Propositional logic</u>

propositional symbols: $P, Q, R, \ldots$

logical connectives: $\wedge, \rightarrow, \leftrightarrow, \neg, \vee$

well-formed formulae

valuation: $\{T, F\}$ $\{0, 1\}$

to propositional symbols.

extend this assignment to all well-formed formulae via the truth-tables.

$\Delta \models \varphi$ iff $\Delta \vdash \varphi$

Δ: set, φ: formula; iff: if and only if.

Semantical — proof-theoretical

$\rightarrow$ <u>logical equivalence</u>

$\psi \equiv \varphi$ — $\psi \models \varphi$ and $\varphi \models \psi$

psi phi — $\psi \vdash \varphi$ and $\varphi \vdash \psi$.

Predicate logic

building blocks

variables: x, y, z
function symbols: f, g, h.
constants: a, b, c.

---

(1) ⇒ terms (can be constructed)

(2) prop. logical connectives: $\wedge, \vee, \Rightarrow, \neg, \leftrightarrow$.

(3) Quantifiers: $\forall, \exists$

(4) predicate symbols: P, Q, R.

D Domain. ⇒ non-empty set of elements.

tweety
wally
felix
odi

⇒ students, lecturers, dogs, birds, penguins.

bird(tweety) ⇒ T
bird(wally) ⇒ T
bird(felix) ⇒ F

$bird^I \subseteq D$ = {tweety, wally}

↑ only birds

$\forall x$ ⇒ universal
$\exists x$ ⇒ existential

$\forall x$ (bird(x)) ⇒ bird(tweety) ∧ bird(wally) ∧ bird(felix) ∧ bird(odi)
⇒ F

Existential

$\exists x \; bird(x)$

$\Rightarrow$ bird(tweety) $\vee$
bird (wally) $\vee$
bird (felix) $\vee$
bird (odi)

$\exists x \equiv \neg \forall x \neg$
$\forall x \equiv \neg \exists x \neg$.

"All birds fly"

$\forall x \, (\underline{bird(x)} \rightarrow fly(x))$.

$\downarrow \qquad \downarrow$

$bird^{I} \; \subset \; fly^{I}$

$\forall x \, (bird(x) \wedge fly(x))$.

$\hookrightarrow$ "everything is a bird <u>and</u> everything flies"

"penguins" $\Rightarrow$ penguin(x)
$\hookrightarrow$ x is a penguin.

"penguins are birds"

$\forall x \, (penguin(x) \rightarrow bird(x))$.

$\exists x\,(\text{bird}(x) \wedge \text{penguin}(x))$. $\underline{\underline{F1}}$

$\exists x\,(\text{bird}(x) \rightarrow \text{penguin}(x))$. $\underline{\underline{F2}}$

$D = \{a, b\}$.

(A)

bird = $\emptyset$

penguin = $\{a\}$

F1 = False .

F2 = True $\underline{x = a}$ bird(a) → penguin(a) .

(B) .

bird = $\{a\}$.

penguin = $\{b\}$.

F1 = false .

F2 = true $\underline{\underline{x = b}}$

(c) bird = $\{b\}$.

penguin = $\{a\}$.

F1 = FALSE .

F2 ⇒ true x = a witness .

(d) bird = $\{a\}$

penguin = $\{a, b\}$.

F1 = true $\underline{\underline{x = a}}$

F2 = true x = a or x = b .

Note: In the above

bird = $\{a\}$ means

$\left.\begin{array}{l} \text{bird}(a) = T \\ \text{bird}(b) = F \end{array}\right\}$ In interpretation (B)

bird = $\emptyset$ means

$\left.\begin{array}{l} \text{bird}(a) = F \\ \text{bird}(b) = F \end{array}\right\}$ In interpretation (A)

and so forth...

(the same applies to penguin)

NATURAL DEDUCTION

(1) (2) (3) ?

$P, \neg Q \to \neg P, Q \to R \vdash R$

Strategy

(A) Need "R"

(B) (3) You can get R if you have Q!

(C) (2) you can get Q if you have P!

$$\neg Q \to \neg P \equiv \neg\neg P \to \neg\neg Q \equiv P \to Q.$$

(d) We have P!

$$\dfrac{\dfrac{P^{\checkmark} \qquad \dfrac{\neg Q \to \neg P^{\checkmark}}{P \to Q \; ?}\ \text{NEED TO } \underline{\text{SHOW}}}{Q^{\checkmark}} \qquad Q \to R^{\checkmark}}{\underline{\underline{R}}}$$

How to show that $\neg Q \to \neg P$ gives me $P \to Q$?

$$\dfrac{B}{A \to B}\ \underline{\to I2}$$

$P \to Q$ Subcomputation box.

$$\boxed{\begin{array}{lll} P & \text{assume} & \underline{\underline{Q}} \\ & \checkmark\ ? & \\ Q & & \end{array}}$$

$$\neg Q \to \neg P.$$

$$\dfrac{\neg Q \to P}{Q}.$$

PROOF 1

$$\frac{B}{A \to B} \to I2$$

1. P . data
2. $\neg Q \to \neg P$. data
3. $Q \to R$ data .
4. $P \to Q$. subcomputation box

Q

4.1 P Assume
4.2 $\neg Q \to P$ 4.1, $\to I2$.
4.3 Q from 2, 4.2, $\neg E$

5. Q $\to E$, 1., 4.
6. R 5, 3, $\to E$ (modus ponens) .

PROOF 2

1. P data .
2. $\neg Q \to \neg P$ data .
3. $Q \to R$ data .
4. $\neg Q \to P$ from 1., $\to I2$.
5. Q from 2, 4, $\neg E$.
6. R from 3, 5, $\to E$

$$\neg(P \wedge \neg P) \equiv P \vee \neg P .$$

$\emptyset \vdash \neg(P \wedge \neg P)$

Sample proof for a tautology

1. $P \wedge \neg P \to P$ from subcomput.

1.1 $P \wedge \neg P$ Assume. $\underline{P}$

1.2 P 1.1, $\wedge E$

2. $P \wedge \neg P \to \neg P$ from subcomput.

2.1. $P \wedge \neg P$ $\underline{\neg P}$

2.2 $\neg P$

3. $\neg(P \wedge \neg P)$ 1, 2, $\neg I$