

プログラミング言語特論レポート

情知 太郎 (171x222x)

平成 29 年 06 月 14 日

1 論理式の書き方

- $p(x) \wedge q(x)$

$\$p(x)\land q(x)\$$

- $p(x) \supset q(x)$

$\$p(x)\imp q(x)\$$

- $\forall x.p(x)$

$\$\forall\text{forall } x. p(x)\$$

- $\exists x.p(x)$

$\$\exists\text{exists } x. p(x)\$$

2 証明図の書き方

証明図を書くために, `proof.sty` を使います.

- 一般形

$$\begin{array}{l} \backslash[\\ \quad \backslash\text{infer}[規則 1]\{A_1, \ldots, A_n\}\{B_1,\ldots, B_m\} \\ \backslash] \end{array}$$

と書くと, 以下のように表示されます.

$$\frac{B_1, \dots, B_m}{A_1, \dots, A_n} \text{ 規則 1}$$

- $\wedge R$ 規則

```

\[
\infer[\land R]{\text{\texttt{\textbackslash CAL\{P\}drv G_1\land G_2}}{
  \text{\texttt{\textbackslash CAL\{P\}drv G_1} \& \text{\texttt{\textbackslash CAL\{P\}drv G_2}}
}
\]

```

と書くと，以下のように表示されます．

$$\frac{\mathcal{P} \longrightarrow G_1 \quad \mathcal{P} \longrightarrow G_2}{\mathcal{P} \longrightarrow G_1 \wedge G_2} \wedge R$$

- *decide* 規則

```

\[
\infer[decide]{\text{\texttt{\textbackslash CAL\{P\}drv A}}{
  \text{\texttt{\textbackslash CAL\{P\}stackrel{D}{\textbackslash drv} A}}
}
\]

```

と書くと，以下のように表示されます．

$$\frac{\mathcal{P} \xrightarrow{D} A}{\mathcal{P} \longrightarrow A} \textit{decide}$$

3 Prolog と証明図

- 以下は Prolog プログラムの例です．

```

% プログラム
app([], Y, Y).
app([X|Xs], Y, [X|Zs]) :- app(Xs, Y, Zs).

% ゴール
?- app([1,2], [3,4], W).
W = [1, 2, 3, 4]
Yes

```

- 以下は Prolog プログラムのトレース例です．

```

?- trace.
Yes
[trace] ?- app([1,2], [3,4], W).
Call: (7) app([1, 2], [3, 4], _G316) ?
Call: (8) app([2], [3, 4], _G379) ?
Call: (9) app([], [3, 4], _G382) ?
Exit: (9) app([], [3, 4], [3, 4]) ?

```

```

Exit: (8) app([2], [3, 4], [2, 3, 4]) ?
Exit: (7) app([1, 2], [3, 4], [1, 2, 3, 4]) ?
X = [1, 2, 3, 4]
Yes

```

- 以下は、プログラムとゴールを論理式で表した例です.

```

% プログラム
app([], Y, Y).
app([X|Xs], Y, [X|Zs]) :- app(Xs, Y, Zs).

```

↓

```

∀y.app([], y, y)
∀x, xs, y, zs.(app(xs, y, zs) ⊃ app([x|xs], y, [x|zs]))

```

```

% ゴール
?- app([1,2], [3,4], W).

```

↓

```

∃w.app([1,2], [3,4], w)

```

4 証明木