

◎ 融合原理の続き
(例題)

$S = \{ \{P, Q, \neg R\}, \{P, \neg Q\}, \{\neg P\}, \{R\} \}$ に融合原理を適用する。

$\{P, Q, \neg R\}$ $\{P, \neg Q\}$

$\{P, R, P\} = \{P, R\}$ $\{\neg P\}$

$\{R\}$ $\{R\}$

$\square (= \perp)$
矛盾

融合結果

別表現

演えき融合

$C_1 = \{P, Q, \neg R\}$

$C_2 = \{P, \neg Q\}$

$C_3 = \{\neg P\}$

$C_4 = \{R\}$

$C_5 = \{P, \neg R\} \leftarrow C_1 \text{ と } C_2 \text{ の融合節}$

$C_6 = \{R\} \leftarrow C_3 \text{ と } C_5 \text{ の } \vee$

$C_7 = \{\perp\} \leftarrow C_4 \text{ と } C_6 \text{ の } \wedge$

S の反駁 (refutation)

(演習) 命題記号を定ギする。

- imaginary : $x = y \rightarrow z$ は架空の動物である。
- mortal : " 死ぬハ" 于運命に依。
- \neg mortal : 不死である。
- magical : " 魔法が使える
- horned : " 角がある。

(I) 与えられた各文を論理式に翻訳する。

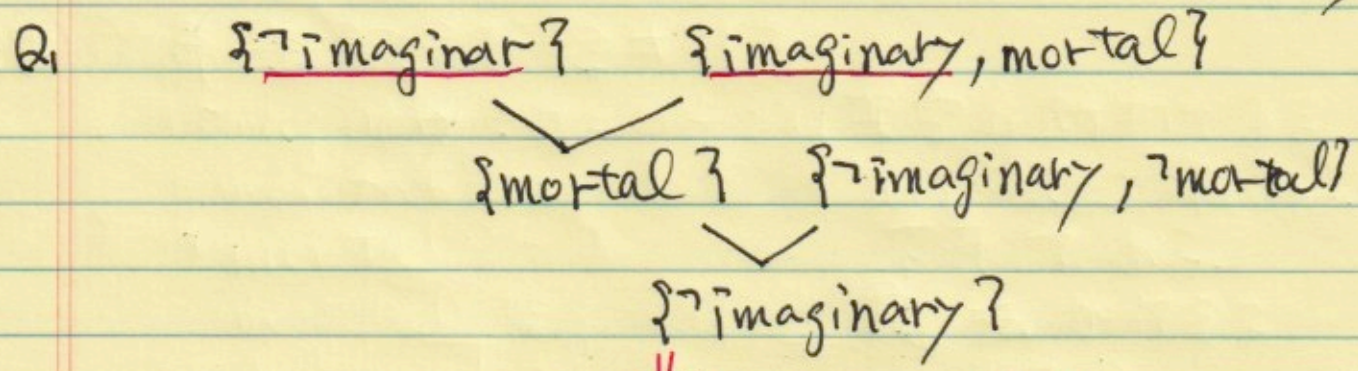
- ① imaginary $\rightarrow \neg$ mortal.
- ② \neg imaginary \rightarrow mortal
- ③ $(\neg$ mortal \vee mortal) \rightarrow horned
- ④ horned \rightarrow magical.

(II) (I) で得られた各論理式を連言標準形に変換

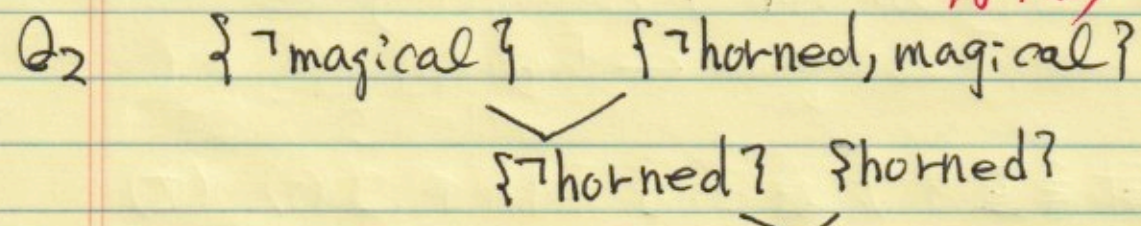
- ①' \neg imaginary $\vee \neg$ mortal
- ②' imaginary \vee mortal
- ③' \neg horned
- $= \neg$ horned
- $= F \vee$ horned $=$ horned
- ④' \neg horned \vee magical

(III) 命題集合に変換

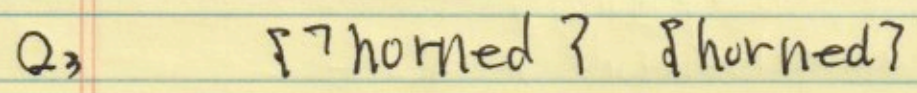
$$S = \{ \{ \neg \text{imaginary} \vee \neg \text{mortal} \}, \{ \text{imaginary} \vee \text{mortal} \}, \{ \text{horned} \}, \{ \neg \text{horned} \vee \text{magical} \} \}$$



二ニ[↓]からは矛盾は得られない。
 つまり反駁は得られない。
 よって imaginary は証明できた。
 (Sの論理的帰結ではない。)



□
 反駁が得られた。
 よって magical が証明された。



□ horned が証明された。

命題論理 } → 標準論理
述語論理 } (古典)

非標準論理 { 時相論理 temporal Logic
(模相論理) modal Logic

① 一階述語の統語論 (syntax)
0項 (term)

个体定項 : 1, 2, 3, ... (数) 1, 2, 1.9, ...
(定数) → PUVアベットの最初の5文字を使う
a, b, c, d, e
 $a_1, a_2, b_{11}, b_{12}, \dots$

个体変数 → x, y, z, w
(論理変数) x_1, x_2, y_{11}, \dots

関数項 → $f, g, h, \dots, f_1, g_{12}, \dots$
項は述語の引数 関数項の引数
項 $f(a_1)$
 $f(g(a, b))$

○ 述語 : P, Q, R, P_1, Q_2, \dots
引数である項の関係性を述べるもの。

(例) $P(a, f(b))$
述語項 関数項

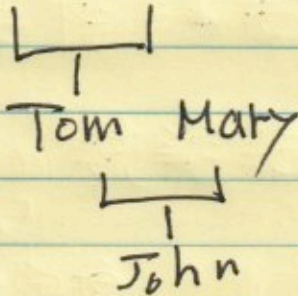
一階述語 (First Order Logic)
→ 述語は述語の引数に存在!!

$P(a, f(b))$
一階述語

P, Q : 述語
 $P(Q(a))$ 二階述語
振るふ!!

⑨ 関数と述語の違いは？

関数 Sam Cathy



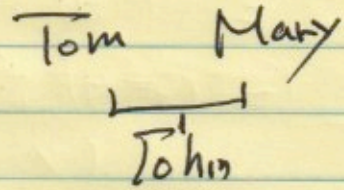
関数

father(John) = Tom

father(father(John)) = Sam

関数合成

述語



一階述語

father(Tom, John)

述語

「TomはJohnの父です。」