[計算機科学向け 圏論の基礎]

デカルト閉圏の学習に向けて

命題論理とその解釈のための代数系の基礎知識

命題論理(古典, 直観主義),
順序集合,
半束, 束, Heyting 代数, Bool 代数

Akihiko Koga

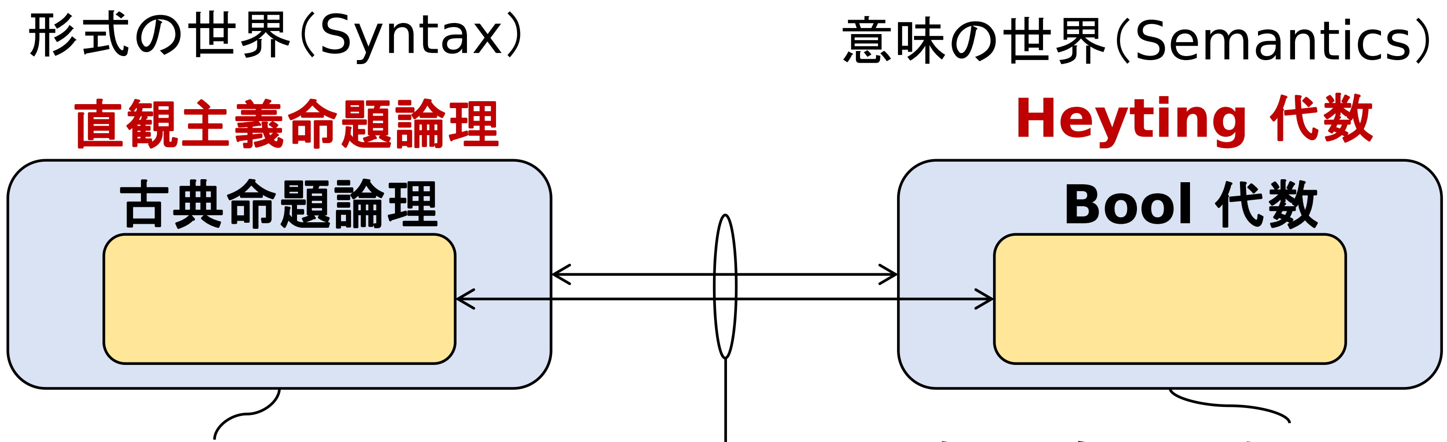
2023年 4月11日

この動画の趣旨

- 次の圏論のトピックスの予定
  - **デカルト閉圏(CCC: Cartesian Closed Category)**
    - **有限積**(terminal object と **A×B**)と**冪(A→B)**がある圏
    - その**計算機科学への応用**
      - **直観主義命題論理の意味論**
      - **単純 型付λ計算の意味論**
- 応用対象を知らないと，応用できるということが，**学習の動機**にならない
- この動画では，**(直観主義)命題論理の方に関する基本的なこと**を大急ぎで学習する
- **λ計算**については，**別に，すでに動画を作成している**ので，知らない人はそちらを見て**学習しておいて欲しい**

この動画の内容

- 登場するトピックスの関係の図



- <u>命題論理の形式的体系</u>
  - 命題論理を記述する言語
  - 証明の体系
  - 古典論理の完全性
  - 直観主義命題論理

- <u>命題論理の解釈のための代数</u>
  - 順序集合
  - 半束と束
  - Heyting 代数と Bool 代数

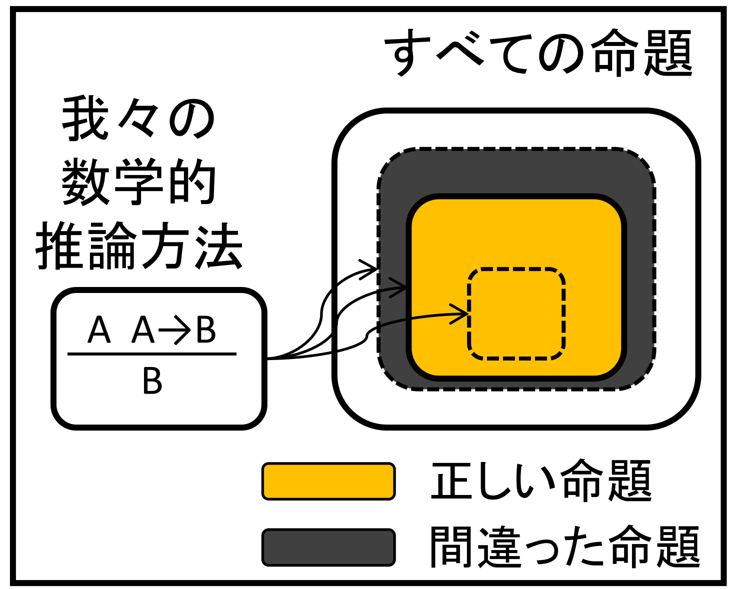
- <u>古典/直観主義論理の代数による解釈</u>

命題論理の形式的体系

- 数理論理学について
- 古典命題論理
  - 形式的体系
  - 意味論
  - 完全性定理
- 直観主義命題論理

数理論理学について

- 数理論理学のような形式的な手法を使った論理学は大昔からあったが，急速に発展してきたのは20世紀初めである
  - １9世紀終わりから，**カントールの集合論**が起こってきた
  - 集合論は**それまでの数学を集合という概念で再構築**し，さらに**無限集合に関する新しい領域を開く**など，**非常に実り豊かな発展**を見せた
  - しかし，次第に**ラッセルのパラドクス**などのパラドクスが起こることも分かってきた
  - 数学者の中には**次のような疑問**を持つ人たちがでてきた
    - 自分たちが行っている**数学的推論は本当に正しいのか**
    - さらに，**正しいことは**，我々が行っている**数学的推論だけですべて導くことができるのか**
- 我々の**数学的推論を厳密に定式化**し，これらを確認する**「数学基礎論」**が起こってきた
  - 第一歩は**論理を厳密に定式化**（＝**数理論理学**）



命題論理とはどんなものか

- 数学基礎論で構築された数理論理学の言明はどんなものか
  - 例： $\forall n(is\_number(n) \rightarrow \exists p(is\_number(p) \land (p > n) \land is\_prime(p)))$
  - **構成要素**
    - 個体変数 n, p, ...
    - 個体定数 0, 1, 2, π, e, ...
    - 関数記号 f, g, exp(･,･), ...
    - 述語記号 is_number(･), is_prime(･)
    - 論理記号 →, ∧, ∨, ¬
    - 限量記号 ∀, ∃
- **命題論理**は，このような言語の中の**論理記号の部分だけ**を取り出した部分．言い換えると，**述語 is_number(n) などを1つの命題としてつぶし**て内部が見えなく，また，**∀と∃を取り除いた**もの
  - **A → B ∧ C ∧ D**

    (n_is_number → p_is_number ∧ p_greater_than_n ∧ p_is_prime)
- **表現力**は**フルの数理論理学の言語**（（一階）述語論理）**より落ちるが，**論理的な推論の全体の中の**重要な一部分**を占めている　→　ここでは**命題論理を学習**する

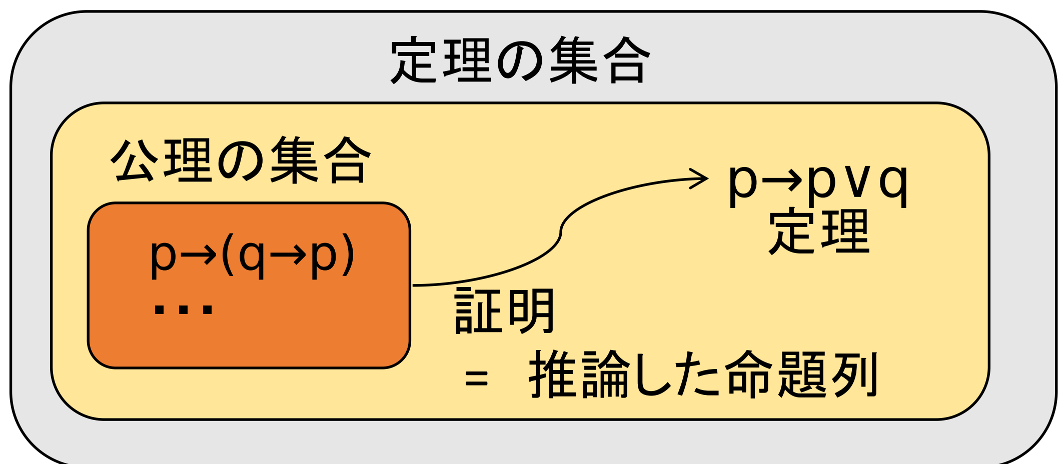
命題論理の定式化

- 言語の定式化
  - 構成要素
    - **命題変数** A, B, C, ..., p, q, r, ...
    - **命題定数** $\perp$ (false), $\top$ (true)
    - **論理記号** $\lor$, $\land$, $\to$, $\neg$
    - **その他の記号** (,)
  - 上の要素を使って**命題を構成する方法**が定められ，**命題の集合**が決定される
    - $A \lor B$,　$A \to A \lor B$,　$\neg A$,　$(A \to B) \to (C \lor \neg \perp)$,　...
- 推論体系の定式化
  - **公理** $A \to (B \to A)$, ...
  - **推論規則** $\dfrac{A \to B \quad A}{B}$　...
- 数学的な推論の定式化

証明 = **命題1 … 命題n**　**定理**

公理 または
先行する命題から推論規則で導出されたもの

すべての命題の集合



命題論理の定式化のまとめと補足

- 命題論理の体系の定式化
  - **言語の指定**
    - **構成要素：　命題変数**（p, q, ...），**命題定数**（⊥, ⊤），**論理記号**（∨, ∧, →, ¬），**その他の記号**（括弧 (,)）
    - **構成要素の組み立て方を決定することにより，命題の集合を決める**：Props
  - **体系の定式化**
    - **公理の集合**：A→(B→A), ...
    - **推論規則の集合**：$\dfrac{A \to B \quad A}{B}$ ...
  - **重要な概念**
    - **証明：　命題の列**で，各命題は**公理**か，**その命題より前の命題から推論規則で得られる命題**.
    - **定理：　証明の最後の命題**になる命題.
- 補足（記号 ⊢　entail と読む）
  - 関係 **Γ ⊢ P**　　（命題 P は**命題の集合 Γ** から証明可能）
    - Γ は命題の有限集合.　**Γ の命題を公理に追加して命題 P が証明可能**であること
    - ∅⊢P は，P が単に証明可能であることを意味する．これを **⊢ P** と書く

古典命題論理（Kleene の体系）

- 公理（Kleene の体系） ... 次の公理図式
  1. A→(B→A)
  2. (A→(B→C))→((A→B)→(A→C))
  3. A∧B → A
  4. A∧B → B
  5. A → (B → A∧B)
  6. A → A∨B
  7. B → A∨B
  8. (A→C) → ((B→C) → (A∨B →C))
  9. (A→B) → ((A→¬B) → ¬A)
  10. A → (¬A → B)
  11. A ∨ ¬A

1–2: **→** に関する公理

3–5: **∧** に関する公理

6–8: **∨** に関する公理

9–11: **¬** に関する公理
- ・背理法に相当
- ・矛盾からはなんでも導かれる
- ・排中律

- 推論規則
  - ■ モーダスポーネンス(MP)
    (Modus ponens 3段論法）

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

公理図式について

- 古典命題論理の Kleene の体系では11個の**公理図式**を上げた
- ここで，「**公理（axiom）**」と言わずに「**公理図式（axiom scheme）**」と言っていることに注意してほしい
- 例えば，公理図式の一番目は次の図式であった
  - **A→(B→A)**
- これは，**一個の公理を表しているのではなく**，**A や B に任意の命題を入れたもの**を表している
  - p→(q→p)　　A := p, B := q
  - r→(s→r)　　A := r, B := s
  - (a→b)→((c→(d→e))→(a→b))　　A := a→b, B := c→(d→e)

  ・・・
- 公理図式とは，いわば**テンプレート**である
- Kleene の体系では**公理図式は11個**だが，**公理自身は無限個**ある

A1→A2→・・・→An→B の直観的な意味

- 動機
  - 命題において**「→」が連なると意味が取りにくい**．例えば，
    - ➢ A→(B→A)
    - ➢ (A→B) → ((A→¬B) → ¬A)
  - 先に進む前に，これらの**直観的な意味**を説明しておく
  - これらはあくまで我々が**密かに心の中で思っておくだけのもの**で，形式的体系において，**これらを根拠に「証明」をしてはいけない**
- 「→」の直観的な意味
  - 古典論理において，A→B は ¬A∨B と等価である．ならば
    - ➢ **A→(B→(C→D))** = ¬A∨¬B∨¬C∨D = ¬(A ∧ B∧C) ∨D（ド・モルガン）
      = **A∧B∧C → D**
    - ➢ つまり，**最後の帰結以外の条件節はみな∧で結んで**，それらから**最後の命題に「→」がでている**のと同じである．
    - ➢ 従って，上の例はそれぞれ次のような直観的な意味を持つ
      - ➢ A→(B→A) は，**A∧B →A**
      - ➢ (A→B) → ((A→¬B) → ¬A) は，**(A→B) ∧ (A→¬B) → ¬A**

古典命題論理(Hilbert の体系)

- Hilbert の体系は，Kleene の体系と同等だが，公理図式が少なく，扱いやすい
- 公理 ... 次の公理図式
  - **A1. A→(B→A)**
  - **A2. (A→(B→C))→((A→B)→(A→C))**
  - **A3. (¬A→¬B)→(B→A)**

Kleeneの体系(11個)より
かなり少ない
(ここに着目することは，
Curry-Howard の対応の
参考にもなる)

- 推論規則
  - モーダスポーネンス(MP) $\dfrac{A \quad A\to B}{B}$
- 補足
  - A∨B と A∧B は次の略記とみなす
    - **A∨B ::= ¬A→B** (A→B = ¬A∨B)
    - **A∧B ::= ¬(A→¬B)**
- 証明の例

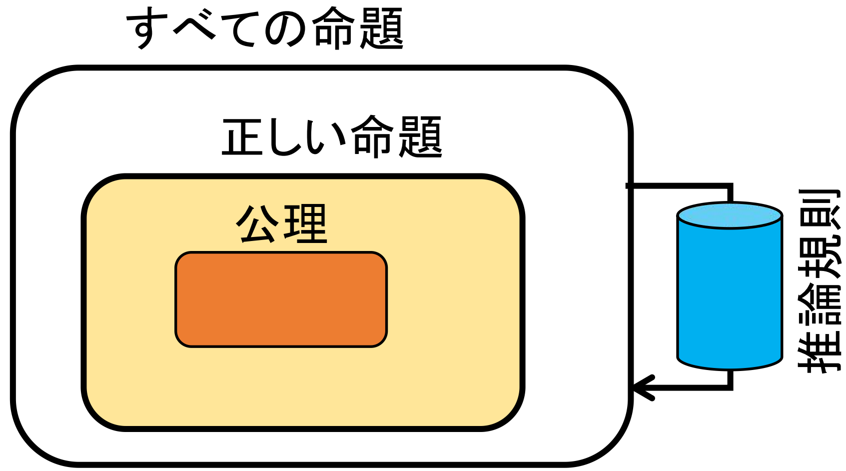
1. (p→(q→p)) → (s→((p→(q→p)) 公理図式 A1 (A := p→(q→r), B := s)
2. p→(q→p) 公理図式 A1 (A := p, B := q)
3. s→(p→(q→p) 1 と 2 から推論規則 MP を適用

記述体系の色々

- 命題論理を記述する形式的体系は色々なものがある
  - Kleene の体系
  - Hilbert の体系

    ・・・
  - Gentzen の体系
    - Natural Deduction
    - Sequent Calculus

    ・・・



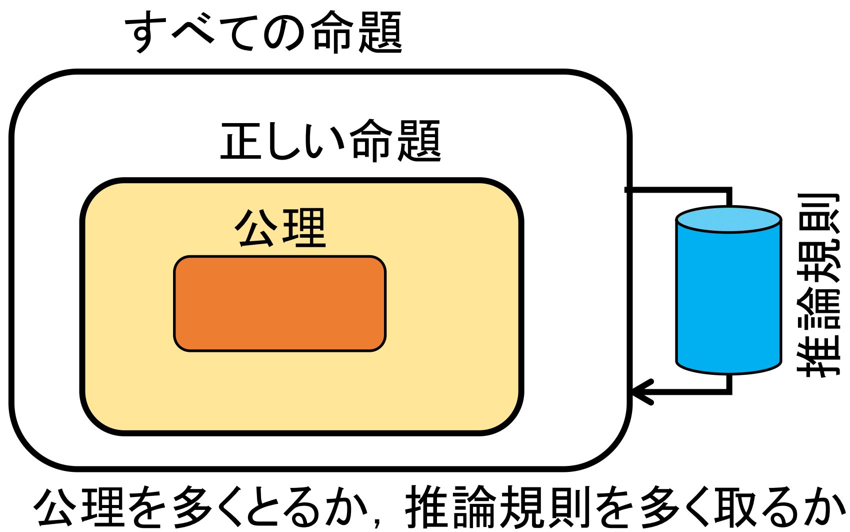
公理を多くとるか，推論規則を多く取るか

典型的な論理体系の比較（命題論理の場合）

体系	公理(図式)	推論規則	特徴
Hilbert の体系	中くらい（3個）	少ない（1個）	・推論規則が少ないので，体系に関する性質の証明は比較的楽 ・体系内の証明自身は難しい
Gentzen の体系（Sequent Calc.）	少ない（1個）	多い（10個程度）	・体系の証明は嵩張る ・体系内の証明は比較的楽

二通りの正しい命題の決め方

- **構文論的アプローチ**
  - これまでは，正しい命題を以下のように，予め正しいとする**公理の集合から推論規則を使って導出**する方法で決めるアプローチを説明した



- **意味論的アプローチ**
  - 伝統的には，もう一つの正さの決め方のアプローチがある
  - **命題 p に対して，真偽値 {0, 1} を計算**する方法を与えることである
  - こちらの方が，皆さんには馴染みが深いかもしれない

命題論理のもう一つの正しさの概念
意味論的アプローチ

- 命題 p∈Props に対して，真偽値 {0, 1} を割り当てる関数を定義する

  μ : Props→{0, 1}

- 定義
  - **命題変数の集合**を **Vars** とする.
  - 関数 **f : Vars→{0, 1}** が与えられたとき，これを命題の集合 Props まで拡張した関数 **μ_f : Props→{0, 1}** を次のように定義する
    - p∈Vars ならば，$\mu_f(p) = f(p)$
    - $\mu_f(p\to q) := \mu_f(p) \to \mu_f(q)$
    - $\mu_f(\neg p) := \neg\mu_f(p)$

p	q	p→q
0	0	1
0	1	1
1	0	0
1	1	1

p	¬p
0	1
1	0

  - Γ を命題の有限集合とする.

    $\mu_f(q)=1$ for **∀q∈Γ** となる **f** に対して **$\mu_f(p)=1$**となるとき，**Γ ⊨ p** と書く.

    （Γ ⊨ p を **p がΓ で valid である**ということもある）
  - Γ が∅のとき，**⊨ p** と書き，このような p を**恒真命題**という.

Hilbert の体系で導かれる命題は恒真か？

- 公理　**p→(q→p)**　　　　　　　　(¬p→¬q)→(q→p)

p	q	q→p	p→(q→p)
0	0	1	1
0	1	0	1
1	0	1	1
1	1	1	1

p	q	¬p	¬q	¬p→¬q	q→p	(¬p→¬q)→(q→p)
0	0	1	1	1	1	1
0	1	1	0	0	0	1
1	0	0	1	1	1	1
1	1	0	0	1	1	1

  - 省略するが，**(p→(q→r))→((p→q)→(p→r))** も恒真

- 推論規則

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

A と A→B が恒真なら，**B も恒真**になる

- 帰結：　**Hilbert の体系で証明できる命題は恒真**である（**健全性**）

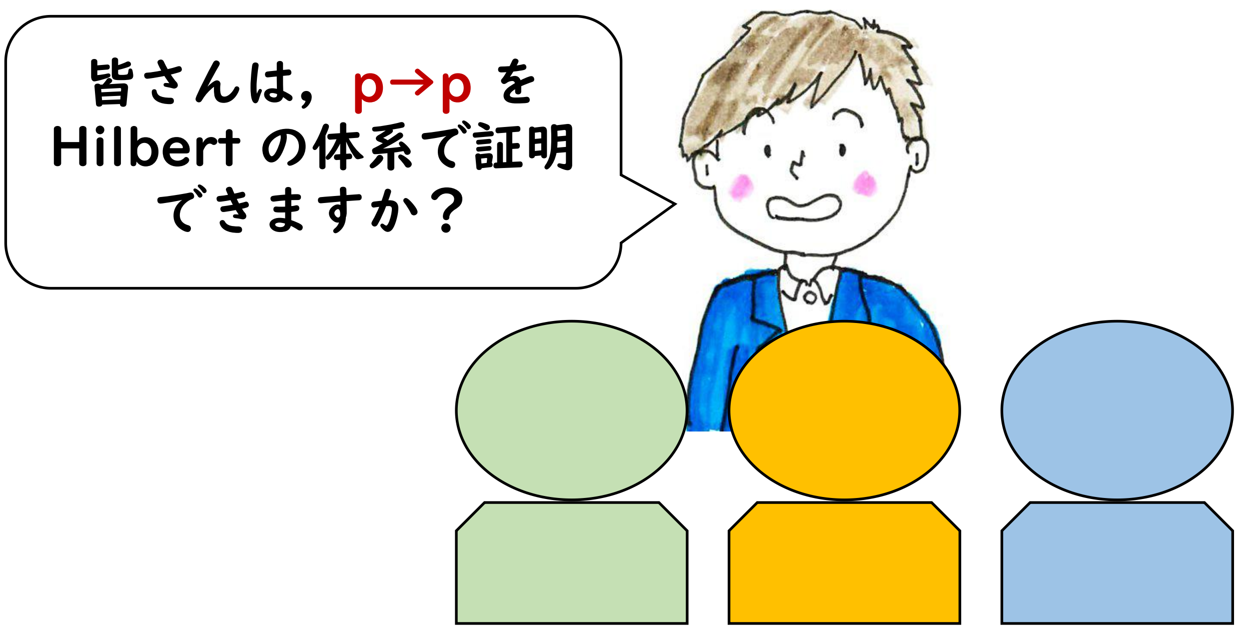
- 疑問：　逆に，**Hilbert の体系は恒真命題をすべて証明できる**か？

恒真の命題の例

- 公理に上げられていない命題で，**恒真な命題の例**
  - **p→p**

p	p	p→p
0	0	1
1	1	1

- Hilbert の体系でこれを証明できるか？



さあ，
どうでしょう？

Hilbert の体系での p→p の証明

- **p→p の証明**
  1. (p→((p→p)→p)→((p→(p→p))→(p→p))
     - 公理A2　(A→(B→C))→((A→B)→(A→C)) に於いて
       - A := p
       - B := p→p
       - C := p
  2. p→((p→p)→p)
     - 公理A1　(A→(B→A)) において
       - A := p
       - B := p→p
  3. (p→(p→p))→(p→p)
     - 1 と 2 から Modus ponens
  4. p→(p→p)
     - 公理A1　(A→(B→A)) において A := p, B := p
  5. p→p
     - 3 と 4 から Modus ponens

Hilbert の体系に関する重要なもの

- 一応，恒真命題の **p→p** は証明できたが，**すごく大変だった！**

- Hilbert の体系は，体系をコンパクトに設計したので，
  - **体系自身について**，たとえば，**正しさを確認するのには都合が良い**が，
  - 一方，**具体的な証明を行うことは非常にやりにくい**
  - これは，この体系で**恒真命題がすべて証明できるかどうかを調べるのも大変**であることを意味する

- Hilbert の体系で証明を容易にし，その証明能力を確認するのに重要なことがいくつかある
  - 演繹定理
  - 重要な定理群

次ページ

命題論理の章の残りの進行

- 完全性定理の証明に向けて
  - **演繹定理**
    - $\Gamma, p \vdash q$ ならば $\Gamma \vdash p \to q$
  - 命題論理の**完全性の証明に有効な定理**など
    - 正しい命題の前に任意の命題を置いて良い
    - 矛盾からはなんでも導かれる
    - 二重否定と元の命題は等価
    - ・・・
- 完全性定理の証明
- 直観主義論理の紹介
- 参考文献など

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B} \qquad A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

完全性定理の証明に有効な定理など

演繹定理（Deduction Theorem）

演繹定理

- 次の公理図式を持ち，

  A1　　$A \to (B \to A)$

  A2　　$(A \to (B \to C)) \to ((A \to B) \to (A \to C))$

  Modus ponens（MP）を推論規則として持つ証明システムは次の性質を持つ

  $\Gamma \vdash A \to B \quad \Leftrightarrow \quad \Gamma, A \vdash B$

- 注意
  - $\Gamma \vdash A \to B$ も $\Gamma, A \vdash B$ も殆ど違わないように思えるかもしれないが，**左で証明すべき式にある「→」が，右では無くなる**. これは**証明すべき式を簡単なものにする**のに効果を発揮する．　例：
    - $\vdash p \to p$　　$p \vdash p$ から演繹定理
    - $\vdash p \to ((p \to q) \to q)$　　$p, p \to q \vdash q$ から演繹定理
  - **「⇒」の証明**
    - $\Gamma, A \vdash A \to B$ だから $\Gamma, A \vdash A \to B$
    - $\Gamma, A \vdash A$
    - これら二つから MP を使って，$\Gamma, A \vdash B$ を得る

演繹定理: $\Gamma \vdash A \to B \Leftrightarrow \Gamma, A \vdash B$ の証明(続き)

- **「⇐」の証明**
  - P の証明の中で **P より前の命題数(証明の長さ)に関する帰納法**を用いる
  - **長さが 0 の場合**
    - この場合, Γ, A ⊢P となるためには次の場合しかない
      - **P が公理**, または, **P∈Γ**　　　どちらの場合も **Γ ⊢ P**
        - 公理 A1: P → (A→P) と P から MP で **Γ ⊢ (A→P)**
      - **P=A**
        - ⊢ P→P だから, A=P なら, ⊢ A→P であり, さらに **Γ ⊢ A→P**
  - **長さが k + 1 の場合(k ≥ 0)**
    - 証明の長さが k 以下では演繹定理が成立すると仮定する
    - **Γ, A ⊢P** の最後のステップで **X, X→P から MP で P を導いた**とする
    - **Γ, A ⊢ X** かつ **Γ, A ⊢ X→P** でそれらの**証明の長さは k 以下**である
    - 従って, 帰納法の仮定から, **Γ ⊢ A→X** かつ **Γ ⊢ A→(X→P)**
    - 公理 A2 から **(A→(X→P))→((A→X) → (A→P))** であるから, **MP を 2回**使って, **Γ ⊢ A→P** が導かれる.

□

古典命題論理の重要な(メタ)定理

- $\Gamma \vdash B$　ならば　$\Gamma \vdash A \to B$

B が証明できるなら，それに条件をつけても証明できる

- $A, \neg A \vdash B$

矛盾からはなんでも証明できる

- $\Gamma \vdash A$　ならば　$\Gamma \vdash \neg\neg A$

Aならばその二重否定も証明できる

- $\Gamma \vdash \neg\neg A$　ならば　$\Gamma \vdash A$

二重否定は除去できる(直観主義命題論理ではなりたたない)

- $\Gamma, A \vdash B$ かつ $\Gamma, \neg A \vdash B$　ならば　$\Gamma \vdash B$

A でも¬A でも証明できる命題 B は，A に関係なく証明できる

- $\Gamma, A \vdash B$ かつ $\Gamma, A \vdash \neg B$　ならば　$\Gamma \vdash \neg A$

Aを仮定して矛盾がでるなら A が間違っている

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B} \qquad A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

重要な(メタ)定理の証明(1)

- **$\Gamma \vdash B$ ならば $\Gamma \vdash A \to B$**

1.	$\vdash B \to (A \to B)$	公理A1
2.	$\Gamma \vdash A \to B$	1 と$\Gamma \vdash B$（仮定）から MP

- **$A, \neg A \vdash B$**

1.	$\vdash (\neg B \to \neg A) \to (A \to B)$	公理 A3
2.	$\neg B \to \neg A \vdash A \to B$	演繹定理
3.	$\neg A \vdash \neg B \to \neg A$	上の定理
4.	$\neg A \vdash A \to B$	2 と 3 から MP
5.	$A, \neg A \vdash B$	$\vdash$ の左の A と 4 から MP

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B} \qquad \boxed{A, \neg A \vdash B}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \quad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \quad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

■ これは演繹定理を使えば $\vdash$ **$A \to (\neg A \to B)$** という形にもなる

補題

補題1（3段論法2）：

$(A\to B)\to((B\to C)\to(A\to C))$ または

$A\to B, B\to C \vdash A\to C$

1. $A\to B, B\to C, A \vdash C$ MP を2回適用
2. $A\to B, B\to C \vdash A\to C$ 演繹定理
3. $A\to B \vdash (B\to C) \to (A\to C)$ 演繹定理
4. $\vdash (A\to B) \to ((B\to C) \to (A\to C))$ 演繹定理

補題2（2回対偶）：

$(\neg\neg A \to \neg\neg B) \to (A \to B)$

1. $(\neg\neg A\to\neg\neg B) \to (\neg B\to\neg A)$ 公理A3
2. $(\neg B\to\neg A) \to (A \to B)$ 公理A3
3. $(\neg\neg A \to \neg\neg B) \to (A \to B)$ 1 と 2 から補題1の3段論法2

重要な(メタ)定理の証明(2)

- **$\neg\neg A \vdash A$**

1. $(\neg\neg A \to \neg\neg B) \to (A \to B)$ 補題2(2回対偶)
2. $(\neg\neg\neg\neg A \to \neg\neg B) \to (\neg\neg A \to B)$ 補題2の A を ¬¬A にする
3. $\neg\neg B \to (\neg\neg\neg\neg A \to \neg\neg B)$ 公理 A1: $(A\to(B\to A))$
4. $\neg\neg B \to (\neg\neg A \to B)$ 1 と 2 と補題1(3段論法2)
5. $\neg\neg A \to (\neg\neg A \to A)$ 4 の B を A にする
6. $\neg\neg A \vdash A$ 5 に ⊢ の左の¬¬A で MP を2回

- **$A \vdash \neg\neg A$**

1. $(\neg\neg\neg A \to \neg A) \to (A \to \neg\neg A)$ 公理 A3
2. $\neg\neg\neg A \to \neg A$ $\neg\neg A\to A$
3. $A \to \neg\neg A$ 1 と 2 にMP
4. $A \vdash \neg\neg A$ 演繹定理

$$\frac{\Gamma \vdash B}{\Gamma \vdash A\to B} \qquad A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

重要な(メタ)定理の証明(4)

- **$\Gamma \vdash A \to B$ (仮定1)かつ $\Gamma \vdash \neg A \to B$ (仮定2)ならば $\Gamma \vdash B$**

1. $\Gamma \vdash \neg B \to \neg A$ 仮定1の対偶
2. $\Gamma \vdash \neg B \to A$ 以下の証明から
    1. $\Gamma \vdash \neg B \to \neg\neg A$ 仮定2の対偶
    2. $\neg\neg A \to A$ 二重否定の除去
3. **$\Gamma, \neg B \vdash A, \neg A$** 1と2
4. $A \to (\neg A \to C)$ 常に成り立つ定理
5. $\Gamma, \neg B \vdash C$ 3と4から2回 MP
6. $\Gamma \vdash \neg B \to C$ 5と演繹定理
7. $\Gamma \vdash \neg B \to \neg(A \to A)$ 6に $C := \neg(A \to A)$
8. $\Gamma \vdash (A \to A) \to B$ 7の対偶
9. $\Gamma \vdash B$ $(A \to A)$ が定理なので8とMP

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B} \qquad A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

- これは演繹定理と組み合わせると，次のようにも言える
    - **$\Gamma, A \vdash B$ かつ $\Gamma, \neg A \vdash B$ ならば $\Gamma \vdash B$**

重要な(メタ)定理の証明(5)

- **Γ, A ⊢ B かつ Γ, A ⊢ ¬B ならば Γ ⊢ ¬A**

1. Γ, A ⊢ B だから Γ ⊢ A→B
2. 対偶をとると Γ ⊢ ¬B→¬A
3. Γ, A →¬B だから Γ ⊢ A→¬B
4. 上の 3 と 2 から，2段階3段論法で Γ ⊢ A → ¬A
5. Γ,　A ⊢ ¬A
6. Γ, ¬A ⊢ ¬A
7. Γ ⊢ ¬A

5 と 6 と $\dfrac{\Gamma,\ A \vdash B \quad \Gamma,\ \neg A \vdash B}{\Gamma \vdash B}$ において B := ¬A

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B} \qquad A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

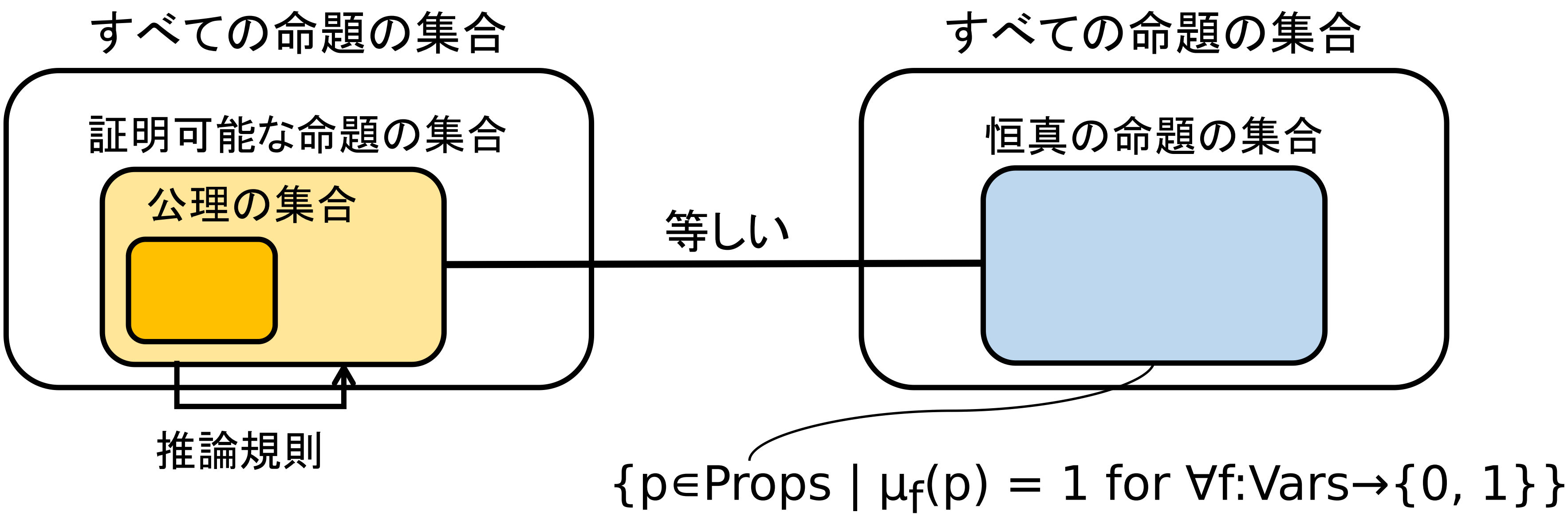
$$\frac{\Gamma, A \vdash B \quad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\boxed{\frac{\Gamma, A \vdash B \quad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}}$$

古典命題論理の完全性

<u>古典命題論理の完全性定理</u>

$$\vdash P \quad \Leftrightarrow \quad \models P$$



⇒ は，**公理が全部，恒真**であることと，**推論規則は，A と A→B が恒真なら，B も恒真になる**ことから明らか．

⇐ は少し難しい

古典命題論理の完全性の証明の概要

- **証明すべきこと:** $\models$ **P** ならば $\vdash$ **P**
  - **任意の割り当て f : vars(P) → {0, 1} に対して μ_f(P) = 1** ならば $\vdash$**P**
- **トライアル**
  - **命題 P の作り方に関する帰納法**で証明するのだろうなと予想できる
    - **P が命題変数の場合** . . . P は恒真にならない
    - **P = ¬Q の場合**
      - **P=¬変数** P は恒真にならない
      - **P = ¬¬Q の場合** **μ_f(Q)=1 for ∀f だから $\vdash$Q**
      - **P = ¬(Q→R) の場合** **μ_f(Q)=1 かつ μ_f(R)=0 for ∀f**
        $\vdash$Q かつ $\vdash$¬R
    - **P = Q→R の場合**
      - μ_f(Q→R) = μ_f(Q)→μ_f(R) =1 になる場合が，**μ_f(Q) とμ_f(R) の関係で決まる**ので，<u>**単純に Q と R の個別の証明に帰着できない**</u>
- **気付き**
  - **すべての命題変数に値が割り当てられていたら，**<u>**P = Q→R の場合でも，Q と R 個別に扱える**</u>

古典命題論理の完全性の証明の概要

- **$\models$P と仮定**する
- 次の**2つのステップ**に分けて証明する
  - **<u>ステップ1</u>**
    var(P) の各命題変数 X について,
    X か¬X のどちらかが命題変数の
    集合Γにあるとき, **$\Gamma \vdash$ P**
  - **<u>ステップ2</u>**
    var(P) の**各命題変数 X について,**
    X と¬X のすべての組み合わせΓ_i に
    ついて **$\Gamma_i \vdash$ P** ならば **$\vdash$P**

$\models$ P　ならば　$\underbrace{\ldots,\ Y,\ \ldots}_{\Gamma} \vdash$ P

X∈var(P) の**各 X について**
X か¬X が入っている

$\begin{pmatrix} \Gamma_1 \vdash P \\ \cdots \\ \cdots \\ \Gamma_n \vdash P \end{pmatrix}$ ならば　$\vdash$ P

各命題変数 X について, X と¬X のすべての組み合わせ

- 上の二つのステップから **$\models$P ならば $\vdash$P** が言える

古典命題論理の完全性の証明の概要 ステップ1

Γ が vars(P) の各変数 X について, **X か¬Xを含む**時, **Γ⊨ P ならば　Γ ⊢ P**

【証明】（P を構成する変数と論理結合子に関する帰納法による）

- **P が命題変数**のとき　　Γ ⊨ P なら, **P∈Γ**　であるから Γ ⊢ P
- **P=¬Q**のとき
  - **P=¬命題変数** のとき　　Γ ⊨ P なら, **P∈Γ** であるから Γ ⊢ P
  - **P=¬¬Q** のとき
    - **Γ ⊨ Q** となるから, 帰納法の仮定により **Γ ⊢Q で, Γ ⊢ ¬¬Q**
  - **P=¬(Q→R)** のとき
    - **Γ ⊨ Q かつ Γ ⊨ ¬R**. 帰納法の仮定により, **Γ ⊢ Q かつ Γ ⊢ ¬R**.
    - Γ ⊢ Q だから　　**Γ, Q→R ⊢ R**
    - Γ ⊢ ¬R だから　　**Γ, Q→R ⊢ ¬R**
    - したがって, **Γ ⊢ ¬(Q→R)**

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

- **P=Q→R** のとき
  - **Γ ⊨ R**のとき,　　帰納法の仮定から **Γ ⊢ R となり, Γ ⊢ Q→R**.
  - **Γ ⊨ Rでない**とき,　**Γ ⊨ ¬Q**. 帰納法の仮定から, **Γ ⊢¬Q. さらに, Γ ⊢¬R→¬Q**
  - 公理 (¬R→¬Q)→(Q→R) とΓ ⊢ ¬R→¬Q から, **Γ ⊢ Q→R**

□

古典命題論理の完全性の証明の概要 ステップ2

X∈vars(P) について，X と¬X の任意の組み合わせΓ で Γ ⊢ P ならば，⊢P

【例証】

- 厳密にはこれも数学的帰納法によるが，ここでは **命題変数が2個 {X, Y} のとき**を**例証**するにとどめる
- vars(P) = {X, Y} で**次の4つ**が成り立っているとする
  1. **X, Y ⊢ P**
  2. **X, ¬Y ⊢ P**
  3. **¬X, Y ⊢ P**
  4. **¬X, ¬Y ⊢ P**
- このとき，次のように⊢ P が言える
  5. **X ⊢ P** なぜなら **Γ, X ⊢ P** (上記1) かつ **Γ, ¬X ⊢ P** (上記2)
  6. **¬X⊢ P** 同様に，上記 **3 と 4 から**
  7. ⊢P 同様に，上記 **5 と 6 から**
- **変数が n 個の場合**も同様に，**各変数 X について，X と ¬X の二通り**，合計 **2^n 通りの Γ**について，**Γ ⊢ P** が示されていれば，Γに現れる**変数をひとつずつ少なくして行って**，**最後に⊢P** が得られることが分かるだろう

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \to B}$$

$$A, \neg A \vdash B$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \neg\neg A}$$

$$\frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

$$\frac{\Gamma, A \vdash B \qquad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$$

直観主義命題論理

排中律 $P \vee \neg P$ を無条件には認めない
（LEM: Law of the Excluded Middle）

(C) 2023, Akihiko Koga

直観主義とは

- **L. E. J. Brouwer** により主張された数学の証明に関する主義･主張 [**1907**]
  - 数学では，ある問題の**解が無いとして矛盾を導く**ことで，その**解があることを証明**することがある（**背理法**）
  - そのように**解を構成する方法を言わない**で，**「無ければ矛盾」**というだけで，その解の**存在を主張してよいのか？**

    よく引き合いに出される例： **「α^β が有理数となる無理数αとβが存在する」**
    - **$\sqrt{2}^{\sqrt{2}}$ が有理数なら**それが解
    - **$\sqrt{2}^{\sqrt{2}}$ が無理数なら** $(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2}\cdot\sqrt{2}} = \sqrt{2}^2 = 2$ が解*
  - 上記の背理法の基盤になっているのは，**排中律 $P \lor \neg P$** であるが，これを無条件に認めることはできない．
  - **排中律 $P \lor \neg P$** は，**Pが証明される場合か，あるいは，$\neg P$ が証明される場合**だけに認めるべきだ
- この主張を取り入れた記号論理の体系は，後に，**Heyting [1930]**, **Gentzen [1935]** and **Kleene [1952]**らにより定式化された
- 直観主義命題論理は，この論理の命題論理の部分である

* 実際は，**$\sqrt{2}^{\sqrt{2}}$ は無理数である**ことが証明されている

直観主義命題論理＊（Hilbert の体系）

- 命題定数
  - 「⊥」を加える（意図としては偽（false）を表す）
  - 略記法として次を加える
    - **¬A := A→⊥**

- 公理 ... 次の公理図式
  A1. A→(B→A)
  A2. (A→(B→C))→((A→B)→(A→C))
  ~~**A3. (¬A→¬B)→(B→A)**~~ （削除）
  A3'. ¬A→(A→B) （追加）

- 推論規則
  - モーダスポーネンス(MP) $\dfrac{A \quad A\to B}{B}$

＊この公理系は Wikipedia の Hilbert system (2023年4月6日 英語版) に基づく

直観主義命題論理(Hilbert の体系)補足

直観主義命題論理のまとめ	■「**⊥(偽)**」の追加と，略記法 **¬A** := **A→⊥**の追加 ■ 推論規則は**モーダスポーネンス** (MP)　**A→B, A** ⊢ **B** ■ 公理図式　A1.　**A→(B→A)** A2.　**(A→(B→C))→((A→B)→(A→C))** A3'.　**¬A→(A→B)**

- 今までの流れで，ここでは**「→」だけ使った直観主義命題論理の体系**を紹介した

- 「→」だけ使った命題について証明できる範囲を変えずに，「∨」と「∧」に関する公理を導入することはできる(**保守的拡張 conservative extension**)
  - ■ ¬A は A→⊥として導入されているので，「→，∨，∧，¬」が導入される
  - ■ **古典命題論理**では **A∨B := ¬A→B**, **A∧B := ¬(A→¬B)** と定義できたが，直観主義論理ではこのようには定義できない
  - ■ 次シートに「→，∨，∧，¬」全部を含んだ保守的拡張の例を示す

直観主義命題論理(Hilbert の体系)補足

- 「→, ∨, ∧, ¬」が定義された直観主義命題論理の例
  (Wikipedia intuitionistic logic 2023年4月6日 英語版 による)
  - $A \to (B \to A)$
  - $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$

  → に関する公理

  - $A \land B \to A$
  - $A \land B \to B$
  - $A \to (B \to A \land B)$

  ∧ に関する公理

  - $A \to A \lor B$
  - $B \to A \lor B$
  - $(A \to C) \to ((B \to C) \to (A \lor B \to C))$

  ∨ に関する公理

  - $\bot \to A$

  ⊥の定義.
  $\neg A := A \to \bot$とする

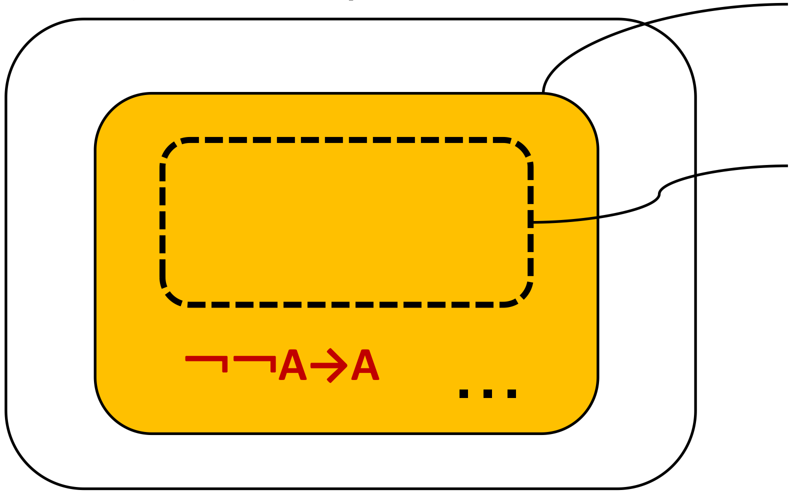
- 推論規則
  - モーダスポーネンス(MP)
    (Modus ponens 3段論法)

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

直観主義命題論理の意味論は？

- 直観主義命題論理は，本質的に**古典命題論理より弱い**

すべての命題の集合



古典命題論理で証明できる命題の集合
＝**恒真命題**の集合

直観主義命題論理で証明できる命題の集合

公理を弱めたため，**恒真命題すべては証明できない**．例えば，次の命題は証明できない

A ∨ ¬A

¬¬A→A　（A→¬¬A は証明できる）

- では，直観主義命題論理はどのように意味論を構成すれば良いか？
  - ■ 古典論理では2値の Bool 代数でよかった
  - ■ この答えが，Heyting 代数であり，これは第2部で説明する

参考：S. Awodey の Category Theory に載っている体系

- S. Awodey の Category Theory には，直観主義命題論理（IPC）として，次の体系が掲載されている
  - **記号**
    - **命題定数**（⊥, ⊤），**命題変数**（p, q, r, ...），**⊢**, **論理結合子**（→, ∧, ∨）

      （論理的な含意は→でなく⇒を使っているが，ここの記号に合わせた）
  - **公理**
    1. ⊢は反射的（p⊢p）であり，推移的（p⊢q かつ q⊢r ならp⊢r）である
    2. p⊢⊤
    3. ⊥⊢p
    4. p⊢q かつ p⊢r　⇔　p⊢q∧r
    5. p⊢r かつ q⊢r　⇔　p∨q⊢r
    6. p∧q→r　⇔　p⊢q→r
  - **推論規則**
    - Modus ponens:　p, p→q ⊢ q

(C) 2023, Akihiko Koga

数理論理学の学習のための参考文献

- **Wikipedia**
  - Propositional calculus
    - 日本語版もあるが現時点(2023年4月6日)では英語版の方が詳しい
  - Intuitionistic logic

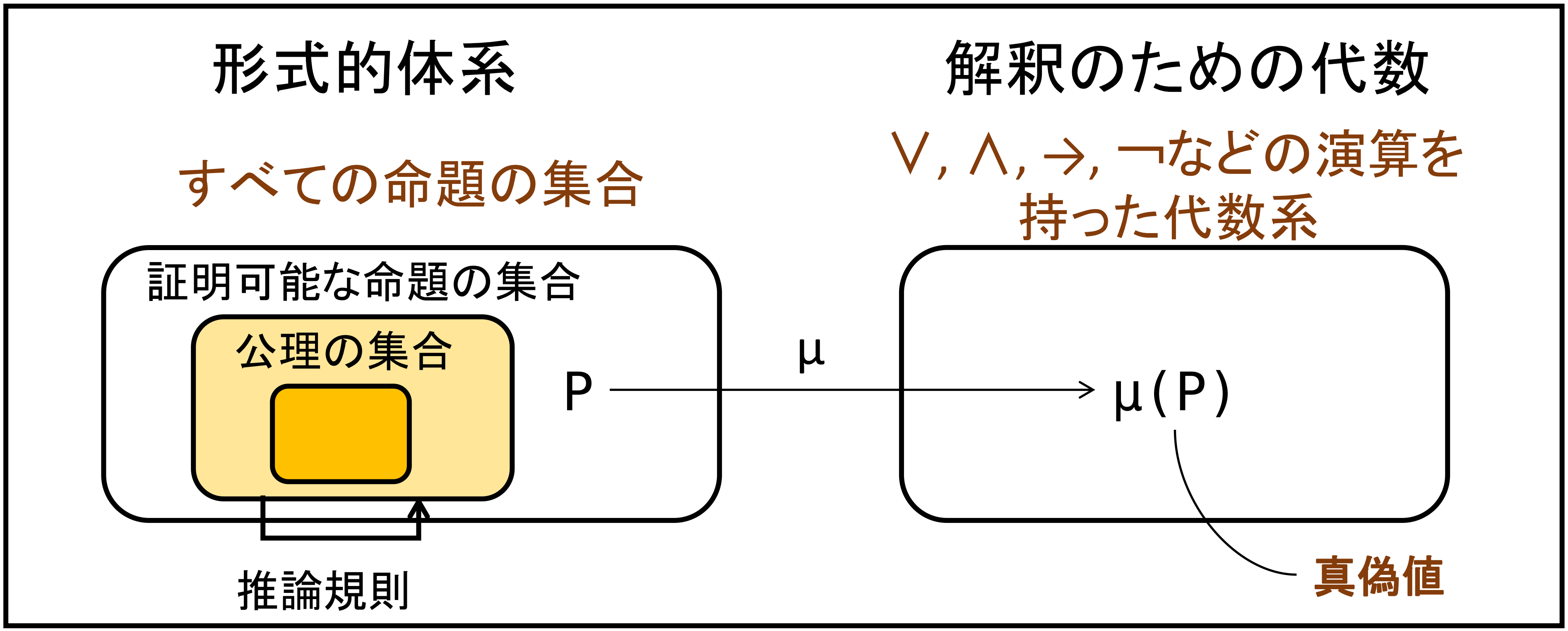
- **Stanford Encyclopedia of Philosophy**
  - Intuitionistic Logic

- Vilnis Detlovs & Karlis Podnieks: **Introduction to Mathematical Logic**, Edition 2021
  - 著作権は Creative Commons

- **私のホームページ**
  - http://www.cs-study.com/koga/cmath/cmath01.html
  - 計算機科学関連の数学の参考資料(1) -- ブール代数, イデアルとフィルター, ハイティング代数 など --

（一部分は, **「そのうち書きます」**になっているところもあります. ごめんなさい. ）

命題論理の解釈のための代数の知識



この章の内容

- この章の最終目標は，直観主義命題論理（IPC）のモデルとなる **Heyting 代数**と，その対比のための **Bool 代数**の理解である

- Heyting 代数だけ述べるという方法もあるが，公理の羅列だけではなかなか理解したという気にならない

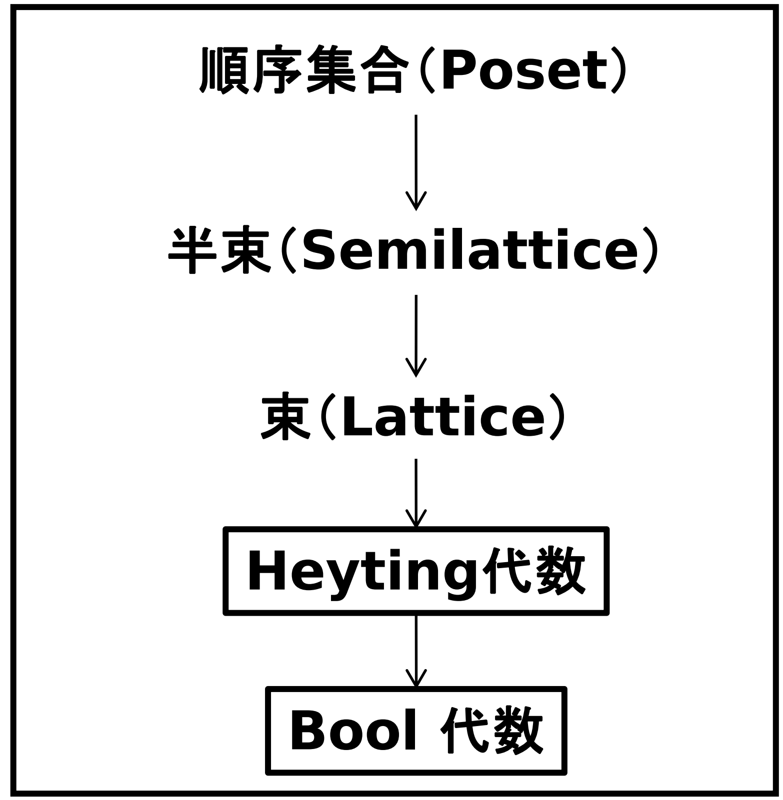
- そこで，ここでは右図のように

  順序集合から，**半束**，　**束**，

  Heyting代数，　**Bool 代数**

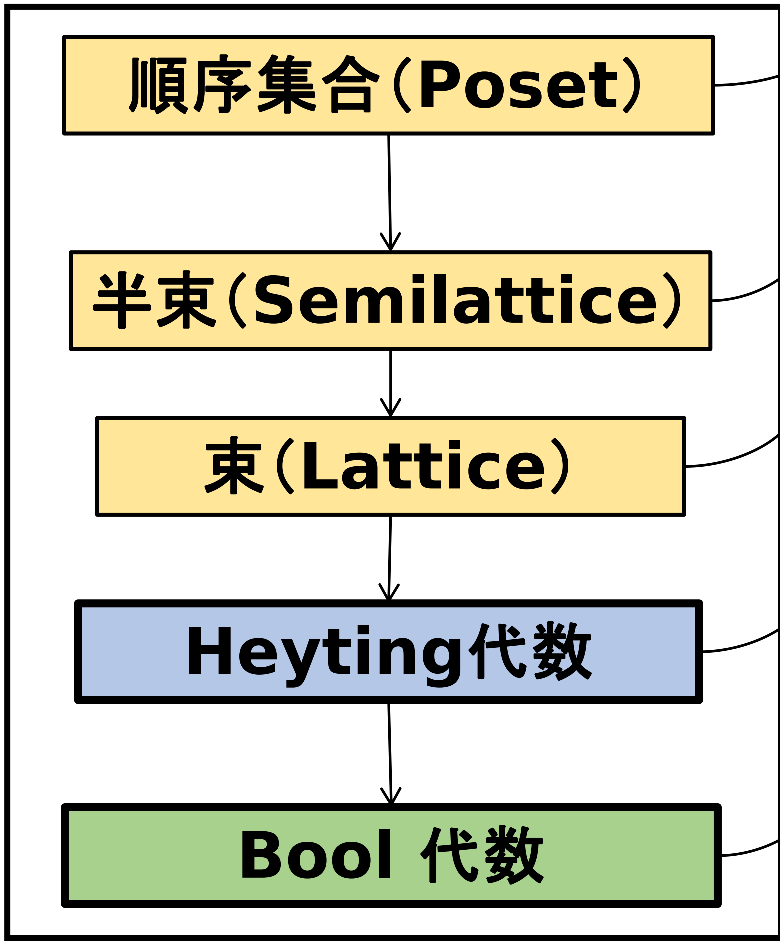
  と，関連する**一連の体系**を紹介して，

  それぞれの**位置づけを明確に**する



一連の体系の位置づけ

- ここで紹介する体系の関係の概要は次の通りである



ここで紹介する体系はみな**順序集合**である

ここから(**演算∨または∧**で)**代数**になる
(**代数**と順序集合が融合する)

演算∨と∧が両方入った代数になる

演算→が加わる
演算∨と∧は**分配法則**を満たす

最大元(⊤), **最小元(⊥)**がある
任意の元 x は**補元¬x**を持つ

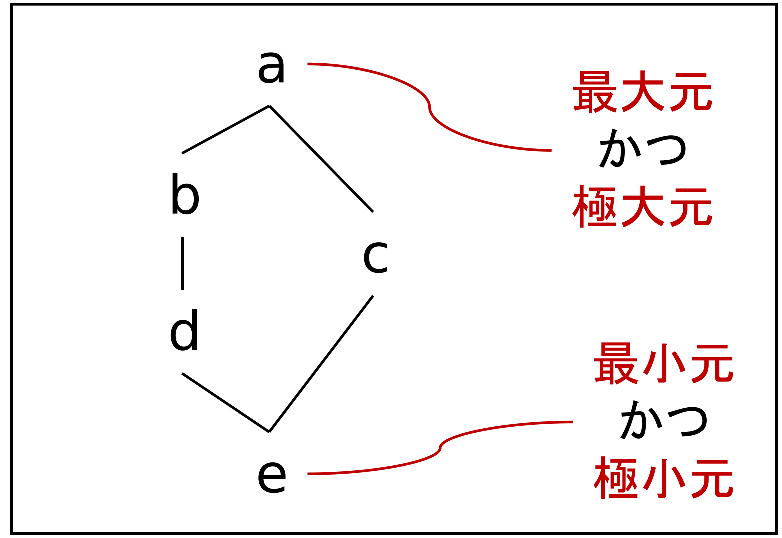
順序集合(Poset : Partially ordered set)

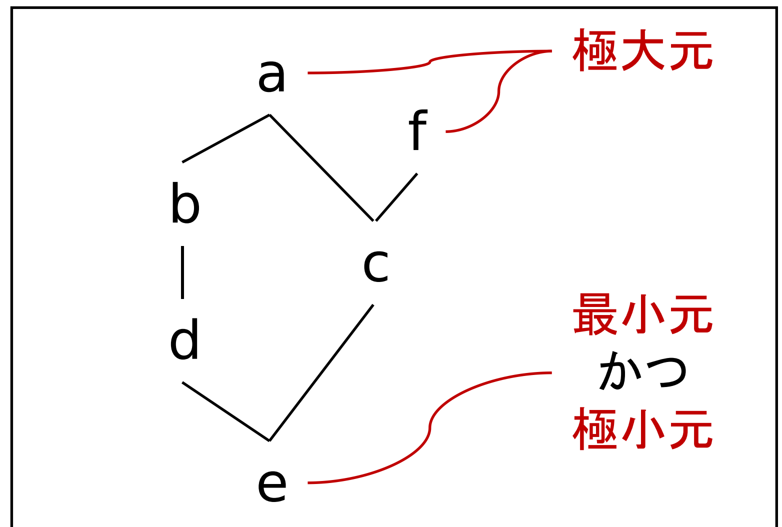
- 順序集合とは，1つの**関係≤**が入った集合のことである
- 定義
  - (P, ≤) が**順序集合(Partially ordered set, Poset)** であるとは，次の3つの性質が成り立つことである
    1. **p≤p** for ∀p∈P 反射律
    2. **p≤q** かつ **q≤r** ならば **p≤r** for ∀p, q, r∈P 推移律
    3. **p≤q** かつ **q≤p** ならば**p=q** for ∀p, q∈P 非対称律
  - 条件1と2だけが成立する場合は**前順序集合(preordered set)**という
  - 条件1, 2, 3 が成り立ち，さらに，次の条件が成り立つとき，(P, ≤) を**全順序集合(totally ordered set)**という
    - **∀p, q∈P について p≤q** または **q≤p**

    つまり，**P のすべての元が比較可能**であるということである．逆に言えば，単にPoset であれば，二つの元は必ずしも比較できなくとも構わない

順序集合上の諸定義

- **ヘッセ図**
  - 右のようなに大小を上下の線で表す図
- (P, ≤) を順序集合とする
  - **最大元（maximum）**
    - x∈P が∀y∈P y≤x を満たすとき，P の**最大元**であるという
  - **最小元（minimum）**
    - x∈P が∀y∈P x≤y を満たすとき，P の**最小元**であるという
  - **極大元（maximal）**
    - x∈P が，∀y∈P x≤y ならば x=y を満たすとき，P の**極大元**であるという
  - **極小元（minimal）**
    - x∈P が，∀y∈P y≤x ならば x=y を満たすとき，P の**極小元**であるという



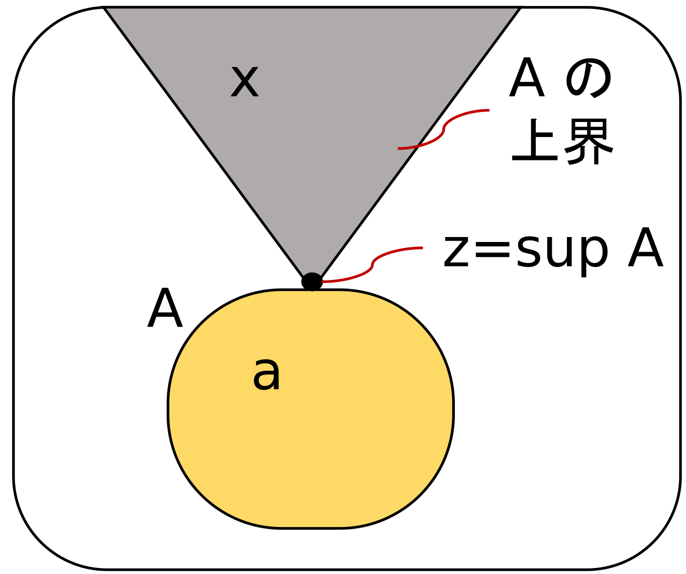


順序集合上の諸定義

(P, ≤) を poset, A⊆P とする.

- **上界(upper bound)**
  - x∈P が A の**上界(upper bound)**であるとは a≤x for ∀a∈A が成立することである
- **下界(lower bound)**
  - x∈P が A の**下界(lower bound)**であるとは x≤a for ∀a∈A が成立すること
- **上限(最小上界)(supremum, least upper bound)**
  - z∈P が A の**上限(supremum)**であるとは, z が A の上界の最小元であること. 上限は, ない場合もある. 記号で **sup A** と書く.
- **下限(最大下界)infimum, maximum lower bound)**
  - z∈P が A の**下限(infimum)**であるとは, z が A の下界の最大元であること. 上限は, ない場合もある. 記号で **inf A** と書く.

(P, ≤)

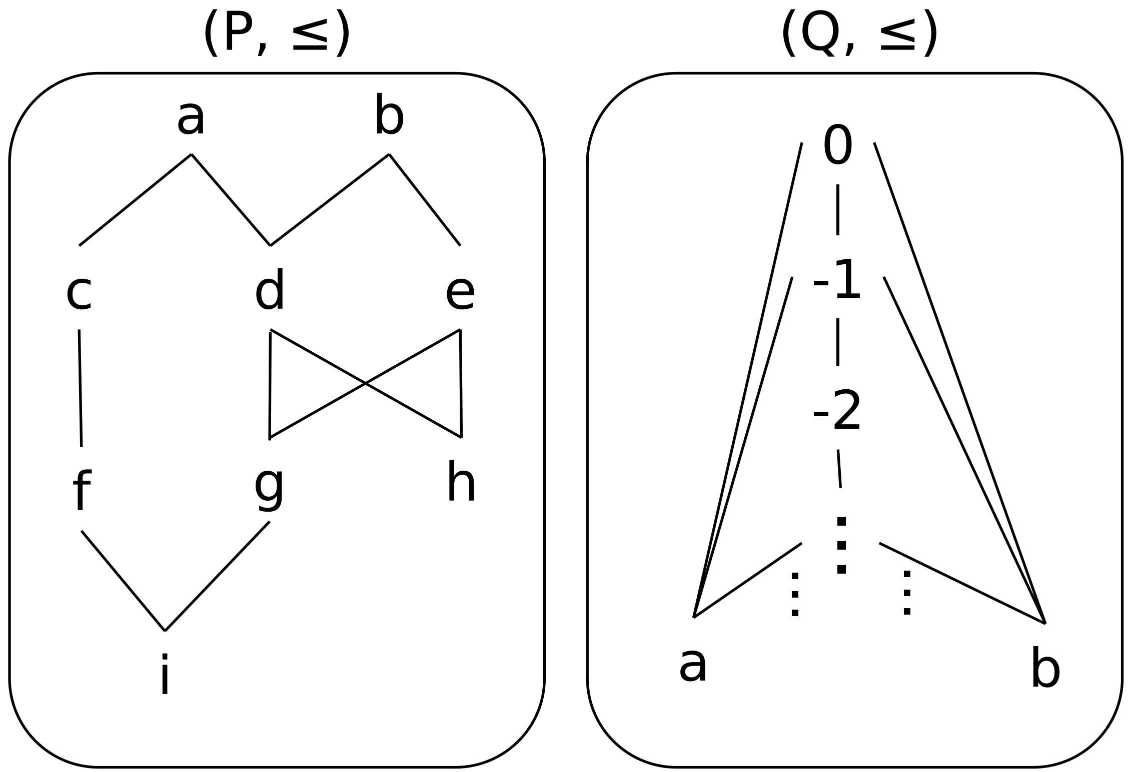


sup A は A の上界の最小元だから, Aに属するとは限らない.

しかし, A に最大元 max A があるときは,
　sup A = max A
になる

順序集合上の諸定義：上限，下限の例

- **上限**，**下限**は以下で重要な役割を演じるので，詳しく見て行こう

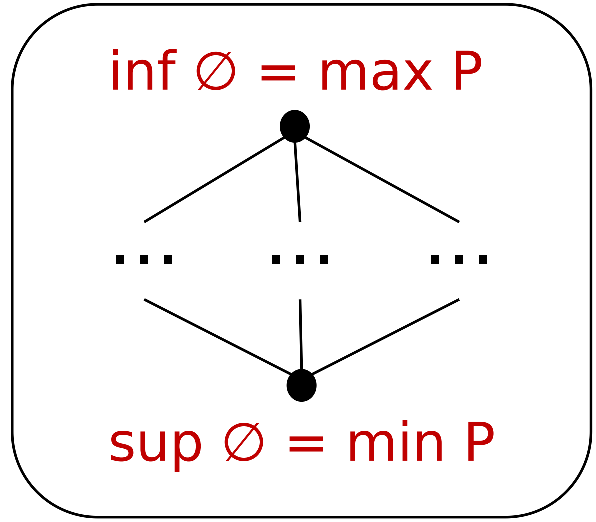


- (P, ≤) において
  - **sup {c, d}** = a
  - **sup {f, g}** = a
  - **sup {c, f}** = c
  - **sup {a, b}** は無い
    a と b 両方以上の元が存在しない
  - **sup {g, h}** は無い
    a と b 両方以上の元は存在するが，それらの最小元が存在しない
- (Q, ≤) において
  - **sup {a, b}** はない
    - a と b の上界に無限降下列がある（最小元がない）

順序集合上の諸定義：∅の上限，下限

- **∅の上限，下限**
  - (P, ≤) を poset とする
    - **sup ∅ = min P**
      ただし，min P が存在するとして
    - **inf ∅ = max P**
      ただし，max P が存在するとして

(P, ≤)



  - これは奇異に見えるかもしれないが，A⊆P の上限 sup A は，**P における A の上限**(sup A = sup A **in P**)である.
    - sup ∅
      = sup ∅ in P
      = min {x∈P | a≤x for ∀a∈A}
      = min {x∈P}

    となる

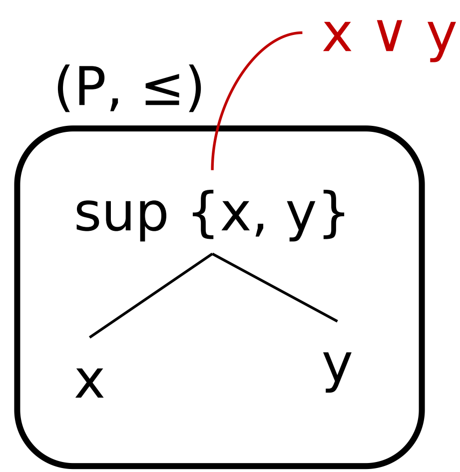
半束(Semilattice)

- 概要
  - 今までは半順序集合を扱ってきたが，半順序集合は**任意の2元に対して上限(下限)が存在する**と，**代数の性格**を持つようになる
  - 半束は，上限の存在を要求するか，下限の存在を要求するかで，**上半束**，**下半束**という2種類の概念が定義される．
  - どちらかを扱えば，もう一方は上下の方向を逆にするだけなので，**ここでは上半束だけを扱う**ことにする

- 定義
  - 順序集合 (P, ≤) が(上)**半束(semilattice)**であるとは，次の性質が成り立つことである．

    任意の2元 x, y∈P に対して，それらの

    上限 sup {x, y} が存在する



  - 注意
    - sup {x, y} は **x ∨ y** とも書き，x と y の**join** と言う
    - 下半束の場合は，x と y の下限 inf {x, y} の存在を要求するが，これは **x ∧ y** と書き，x と y の **meet** という

半束（Semilattice）の例

- 集合 S に対して**べき集合 ($\wp$(S), ⊆)**
  - X, Y∈$\wp$(S) に対して, **X ∨ Y** := **X ∪ Y** は X と Y の上限になる

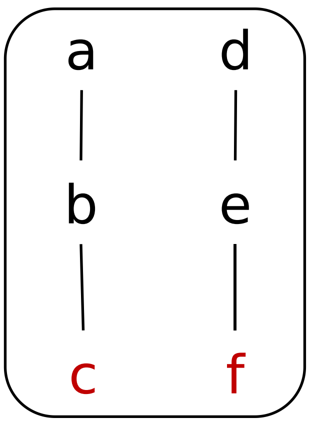
- **自然数の集合 (ℕ, ≤)**
  - **x ∨ y** := **max(x, y)**

- 上記2つは，下限も存在するので，後で定義する束にもなる

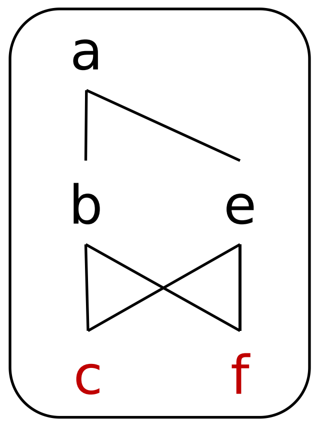
- 集合 S に対して次のように作った **S の部分集合の族**
  - A⊆$\wp$(S) があるとする.
  - **Q(A)** := **{X | X は, A の元の有限個の和集合になる}** とし，それに包含関係**⊆**で順序を入れる
  - このとき，X, Y∈Q(A) に対して, **X ∪ Y はQ(A) に入っている**ので，上限はあるが，一般に, **X ∩ Y が Q(A) に入っているとは限らない**ので，A の取り方によっては，下限はないこともある

半束（Semilattice）で無い例

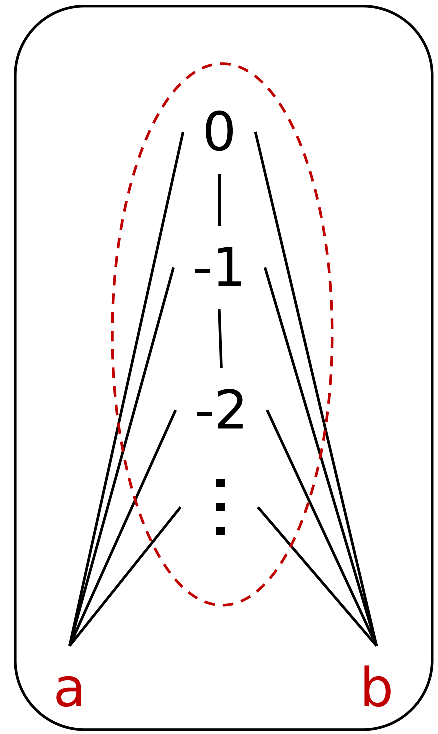
- 順序集合 (P, ≤) が**半束にならないため**には，ある x, y∈P に対して，その**上限 x∨y が無ければ**よい
- これには次のような場合がある
  - x と y の**共通の上界が存在しない**
  - x と y の**共通の上界は存在する**が，その**最小元がない**
    - x と y の共通の上界に**下限はあるが最小元がない**
    - x と y の共通の上界に**無限降下列がある**



c と f に共通の上界がない



c と f に共通の上界に最小元が無い



a と b に共通の上界に無限降下列がある

半束(Semilattice):代数としての性格

- 半束への二項演算の導入
  - $(P, \leq)$ を**半束**とする
  - このとき，x, y∈P に対して，**演算 ∨** を $x \vee y := \sup\{x, y\}$ と定義する
  - ∨は次の性質を持った演算である
    - $x \vee x = x$ 冪等律(idempotency)
    - $x \vee y = y \vee x$ 可換律(commutativity)
    - $(x \vee y) \vee z = x \vee (y \vee z)$ 結合律(associativity)

- **二項演算から半順序の定義**
  - **集合 Q** 上の演算**∨**が**冪等律**，**可換律**，**結合律**を満たすとする.
  - このとき．Q の上の**2項関係≤**を**次のように定める**と，**≤は半順序**になる
    - $x \leq y \quad \Leftrightarrow \quad x \vee y = y$
  - 証明
    - $x \vee x = x \quad \Rightarrow \quad x \leq x$
    - $x \leq y$ **かつ** $y \leq z$ ならば $x \vee y = y$ かつ $y \vee z = z$. したがって，
      $x \vee z = x \vee (y \vee z) = (x \vee y) \vee z = y \vee z = z \Rightarrow x \leq z$
    - $x \leq y$ **かつ** $y \leq x$ ならば $x \vee y = y$ かつ $x \vee y = x \Rightarrow x = y$

半束(Semilattice):代数としての定義

- 以上のことから，半束は次のように代数としても定義できる

- 定義
  - 演算∨が定義された集合 (S, ∨) が**半束**であるとは，次の性質が成り立つことである
    - $x \vee x = x$　　冪等律
    - $x \vee y = y \vee x$　　可換律
    - $(x \vee y) \vee z = x \vee (y \vee z)$　　結合律

(S, ≤) が**順序の意味で**(上)半束

∀x, y に対して
sup {x, y} が存在

$x \vee y := \sup \{x, y\}$ →

← $x \leq y \Leftrightarrow x \vee y = y$

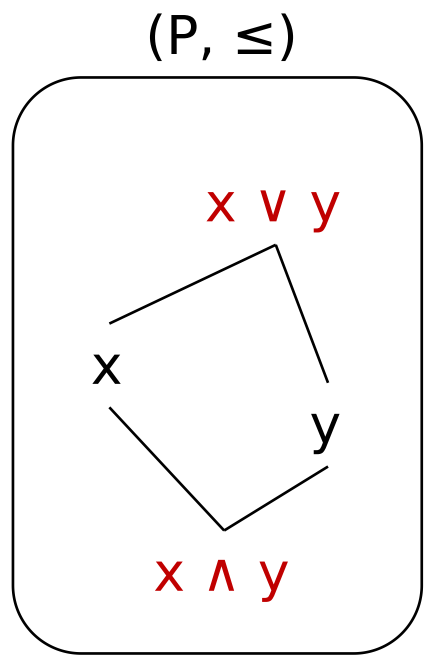
(S, ∨) が**代数の意味で**(上)半束

$x \vee x = x$　冪等律
$x \vee y = y \vee x$　可換律
$(x \vee y) \vee z = x \vee (y \vee z)$　結合律

(C) 2023, Akihiko Koga

束(Lattice)の定義(順序集合として)

- 定義
  - (P, ≤) が**束(lattice)**であるとは，次の性質が成立することである
    - 任意の2元 x, y ∈P に対して，**上限 x ∨ y** と **下限 x ∧ y** が存在する
  - (P, ≤) を束とするとき，P が**有界(bounded)**であるとは，P に最大元と最小元が存在することであるとする．最大元を **1** ，最小元を **0** で表すことが多い



- 参考 ... 圏とのかかわり
  - 順序集合(P, ≤) は **x≤y** のとき，**射 x→y が一本だけ**あると考えると，**圏になる**.
  - x, y∈P の
    - **x ∧ y** は，x と y の**積(product)**
    - **x ∨ y** は，x と y の**余積(coproduct)**にあたる.
  - したがって，**束は，圏としてみたときは，任意の2元に対して積と余積が存在する圏**である

束（Lattice）の定義（代数として）

- 束は代数的にも定義することができる
- 定義
  - 二つの二項演算∨と∧が定義された集合 (P, ∨, ∧)が束であるとは次の性質が成立することである
    - $x \vee x = x$　∨ の冪等律
    - $x \vee y = y \vee x$　∨ の可換律
    - $(x \vee y) \vee z = x \vee (y \vee z)$　∨ の結合律

$x \leq_1 y \Leftrightarrow x \vee y = y$ で **半順序**が定義できて，2元に対して**上限が存在**

- $x \wedge x = x$　∧ の冪等律
- $x \wedge y = y \wedge x$　∧ の可換律
- $(x \wedge y) \wedge z = x \wedge (y \wedge z)$　∧ の結合律

$x \leq_2 y \Leftrightarrow x \wedge y = x$ で **半順序**が定義できて．2元に対して**下限が存在**

- $x \wedge (x \vee y) = x$　吸収律
- $x \vee (x \wedge y) = x$　吸収律

$\leq_1$ と $\leq_2$ は1つの順序である

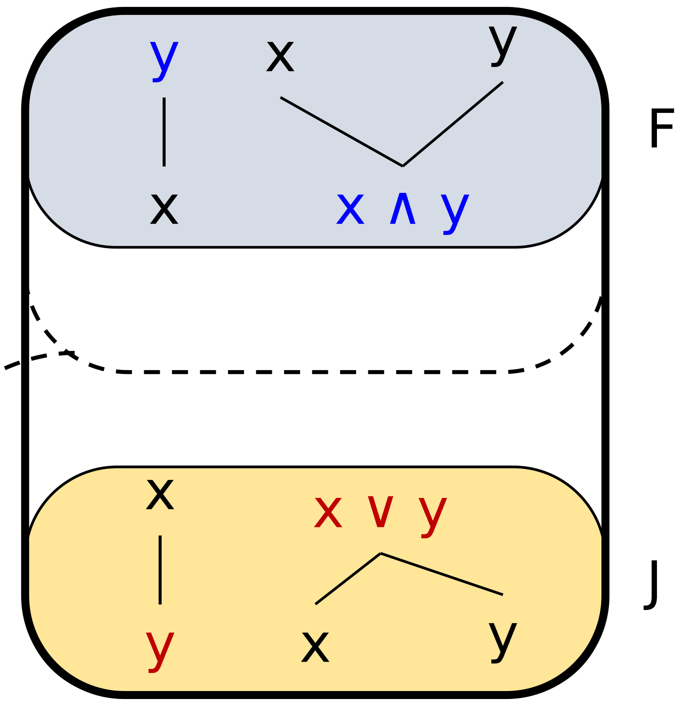
$x \wedge (x \vee y) = x \quad \Rightarrow \quad x \leq_1 y \Leftrightarrow x \vee y = y \Leftrightarrow x \wedge (x \vee y) = x \Leftrightarrow x \leq_2 y$

$x \leq_1 y \Leftrightarrow x \leq_2 y \quad \Rightarrow \quad x \leq_1 x \vee y \quad \Rightarrow \quad x \wedge (x \vee y) = x$

諸定義(イデアル，フィルター)

- 定義 イデアル，フィルター
  - (L, ∨, ∧) を束とする.
  - J⊆L が**イデアル(ideal)**であるとは次の条件が成り立つことである.
    - J≠∅, J≠L
    - x∈J かつ y∈L が y≤x ならば，y∈J
    - x, y∈J ならば x ∨ y ∈J
  - F⊆L が**フィルター(filter)**であるとは次の条件が成り立つことである.
    - F≠∅, F≠L
    - x∈F かつ y∈L が x≤y ならば，y∈F
    - x, y∈F ならば x ∧ y ∈F
  - **極大なフィルター**(何か新しい元を加えたら L 全体になってしまうフィルター)を**超フィルター(ultrafilter)**という.

(L, ∨, ∧)



束論の何が良いのか

- ここで立ち止まって
  - これまで，順序集合にいくつか条件を付けて，束と言う概念を定義してきた
  - ところで，**束は何か役に立つのだろうか？**

- 束論の**成り立ち**から
  - 代数の構造を解析するのに使える（**普遍代数**とともに発展）
- **圏**という立場から
  - 順序集合を圏として見たとき，**積（product）**と**和（coproduct）**が入っている
- **計算機科学**の人間にとって
  - 束，そのものはそれほど強くは出てこないかもしれない
    - **トップが無い場合が多い**
    - 束と似た概念がよく出てくるので，**隣接領域として束論**はかなり役に立つ
  - **形式的概念分析（FCA**: Formal Concept Analysis）は束論で論じられる
  - 束論で論じられる **Heyting 代数**と **Bool 代数** は重要である

計算機屋さんが束論を勉強するには

- 出版されている本
  - B. A. Davey & H. A. Priestley, **Introduction to Lattices and Order** (2nd Edition), 2002
- Lecture Note
  - J.B. Nation, **Notes on Lattice Theory**, course notes, revised 2017.
    - Wikipedia "Lattice (order)" の参考文献に，ハワイ大学の **J.B. Nation** 名誉教授のページのリンクがあり，そこに PDF がある．
    - 基本的には計算機科学用でなく，**数学系の教科書**だが，**計算機科学の学習者でも役に立つ**ことが多いので，これで勉強して，その他**必要に応じて計算機科学の知識を補えば**良いと思う
    - ただし，注意が2点ある（これらは下の**私のホームページに解説**がある）
      - **order ideal**, **order filter** は現在の多くの教科書と使い方が異なる
      - 完備束を定義するときに使っている **closure rules** という概念が少し分かりにくいかも（私は分かりにくかった）
- **私のホームページ**にも少しだが解説している

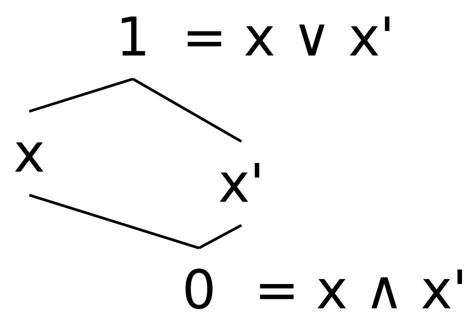
  http://www.cs-study.com/koga/lattice/index.html

分配束(Distributed Lattice)

- 定義
  - (L, ∨, ∧) を束とする. これが次の**分配法則**を満たすとき, **分配束**であるという.
    - $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ for x, y, z∈L　**分配法則1**
    - $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ for x, y, z∈L　**分配法則2**
- 補足
  - もともと, **束なら**, $x \wedge (y \vee z) \geq (x \wedge y) \vee (x \wedge z)$
    - $x \geq x \wedge y$ かつ $x \geq x \wedge z$ だから, $x \geq (x \wedge y) \vee (x \wedge z)$
    - $y \vee z \geq x \wedge y$ かつ $y \vee z \geq (x \wedge z)$ だから, $y \vee z \geq (x \wedge y) \vee (x \wedge z)$
    - 従って, $x \wedge (y \vee z) \geq (x \wedge y) \vee (x \wedge z)$
  - 従って, 逆の不等式もなりたつということが分配束の特徴
  - **二つの分配法則**を書いたが, **一方が成り立てば, もう一方は必然的になりたつ**
    - $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ とする(**分配法則1**)
    - 分配法則2の右辺 $= (x \vee y) \wedge (x \vee z) = ((x \vee y) \wedge x) \vee ((x \vee y) \wedge z)$
      $= x \vee (x \wedge z) \vee (y \wedge z) = (x \vee (x \wedge z)) \vee (y \wedge z) = x \vee (y \wedge z)$

補元 ￢x

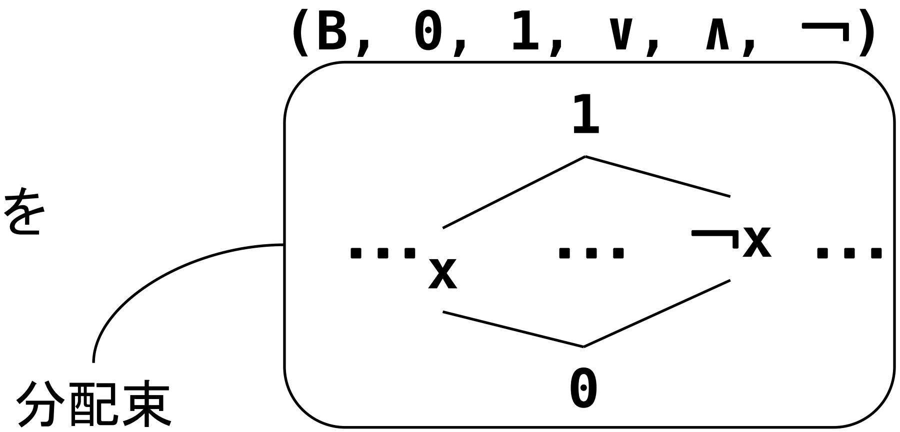
- 定義
  - (L, ∨, ∧) を，**最小元 0 と最大元 1 がある束**とする．x∈L に対して，次の2つの性質が成り立つ元 x'∈L があるとき，x' を x の**補元（complementary element）**という．
    - ➢ x ∨ x' = 1
    - ➢ x ∧ x' = 0



  - 元 x∈L に対して，補元はあるとは限らないし，また，ある場合も一意に決まるとは限らない
  - しかし，**L が分配束の場合は，存在すれば一意に決まる**
    - ➢ **x' と x'' を x の補元**とすると
      - ➢ **x' ∨ x''** = x' ∨ (x'' ∨ 0) = x' ∨ (x'' ∨ (x ∧ x'))
      = x' ∨ ((x'' ∧ x) ∨ (x'' ∧ x')) = x' ∨ (x'' ∧ x')
      = (x' ∨ x'') ∧ (x' ∨ x') = (x' ∨ x'') ∧ x' = **x'**
    - ➢ 故に **x''≤x'**．同様に **x'≤x'' も証明できる**ので，**x' = x''**．

Bool 代数(Boolean algebra)

- 定義
  - **最大元**, **最小元**が存在する**分配束 (B, ∨, ∧, 0, 1)** の各元に**補元が存在**するとき, B を **Bool 代数**という.
  - X の補元は一意に定まる.
    その補元を**¬X** で表すことにする.
  - 0, 1, 補元を明らかにして, Bool 代数を
    (B, 0, 1, ∨, ∧, ¬)
    のように表すことがある.



  - B1, B2 が Bool 代数のとき, 関数 f : B1 → B2 が最大, 最小, および演算を保つとき, f を**準同型写像**という.

Bool 代数の例

- **2** 二値 Bool 代数
  - **2** := {0, 1}
  - ∨, ∧, ¬ は右の表の通り

X	Y	X ∨ Y	X ∧ Y
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

X	¬X
0	1
1	0

- **℘(S)** for 集合 S
  - X, Y∈℘(S) に対して
    - X ∨ Y := X ∪ Y
    - X ∧ Y := X ∩ Y
    - ¬X := S - X

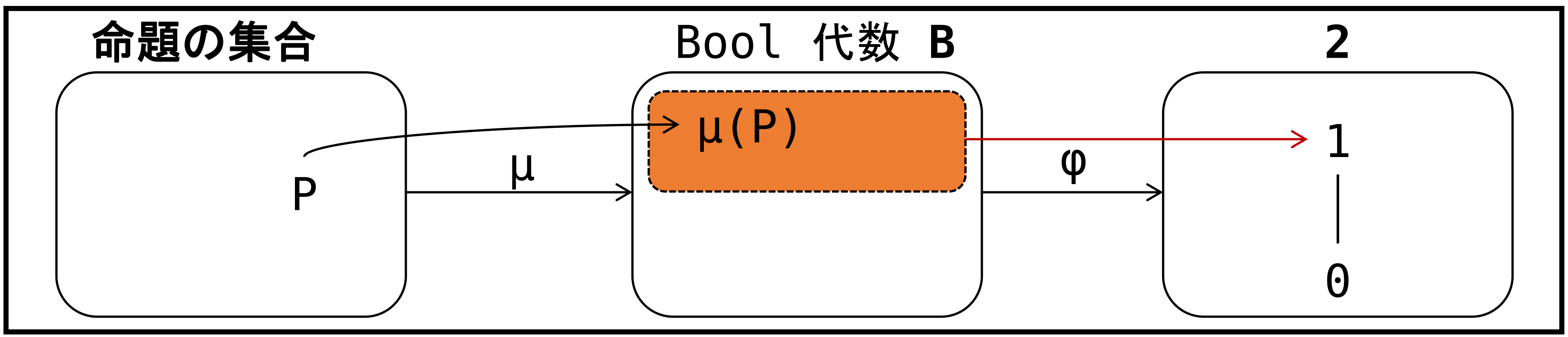
- 無限集合 S の**有限および余有限部分集合**の集合
  - B := **{X ⊆ S | |X| < ∞ }** ∪ **{X ⊆ S | |S - X| < ∞}**
  - ∨, ∧, ¬ については, ℘(S) と同じ

Bool 代数の性質

- 有限 Bool 代数の型
  - **有限 Bool 代数**は，何か集合 S の**$\wp(S)$ と同型**になる
  - 従って，要素数 |B| は必ず2のべき乗になる
- Ultrafilter
  - **φ** : B→{0, 1} が **Bool 代数の準同型**なら，**$\varphi^{-1}(1)$ は Ultrafilter** になる
    - **$\varphi^{-1}(1)$ がフィルターであること**
      - **$x \in \varphi^{-1}(1)$ かつ $x \leq y$** なら，$1=\varphi(x) \leq \varphi(y)=1$ で**$y \in \varphi^{-1}(1)$**
      - **$x, y \in \varphi^{-1}(1)$** なら $\varphi(x \wedge y)=\varphi(x) \wedge \varphi(y)=1 \wedge 1=1$ で，**$x \wedge y \in \varphi^{-1}(1)$**
    - **$\varphi^{-1}(1)$ の極大性**
      - $\varphi^{-1}(1)$ を真に含むフィルター F があったとする．**$x \in F - \varphi^{-1}(1)$** とすると，**$\varphi(x)=0$** である．
      - $x \vee \neg x = 1$ であるから，$1=\varphi(x \vee \neg x)=\varphi(x) \vee \varphi(\neg x) = 0 \vee \varphi(\neg x) = \varphi(\neg x)$ で，**$\varphi(\neg x) =1$**．したがって，**$\neg x \in \varphi^{-1}(1) \subseteq F$**
      - これは，**$x \wedge \neg x \in F$** であることを意味するが，$x \wedge \neg x = $ **$0 \in F$** なので，Fが上に閉じていることから，**F は B 全体**になってしまう

Ultrafilter について一言

- 以下では ultrafilter は出てこないが，以下の理由からコメントしておく
  - ■ ultrafilter は**圏論の学習**でも出てくる．例えば，**S. Awodey の圏論**でも，**自然変換の章**で，**Stone の同型定理**で出てくる.
  - ■ **計算機科学**の理論的な部分と**隣接する領域**で（or **数学基礎論**）で結構出てくる
    - ➢ **超準解析**で**無限小のような概念を作り出す**
    - ➢ **連続体仮説**の **ZFC からの独立性**を示
  - ■ 要は 2 = {0, 1} への準同型φがあったとき，$\boldsymbol{\varphi^{-1}(1)}$ である
    - ➢ Bool 代数 B をとったとき，**φの意味で最終的に真とみなし得る真偽値の集合**



Bool 代数の性質(2)

- 諸定理

$\neg\neg x = x$ 二重否定はもとのものに等しい

- $\neg x \vee x = 1$ かつ $\neg x \wedge x = 0$ だから，x は$\neg x$ の補元であり，Bool 代数では，補元は一意に決まるので，$\neg\neg x = x$

$\neg(x \vee y) = \neg x \wedge \neg y$ ド・モルガンの法則1

- $(x \vee y) \vee (\neg x \wedge \neg y) = (x \vee y \vee \neg x) \wedge (y \vee y \vee \neg x)$
  $= 1 \wedge 1 = 1$
- $(x \vee y) \wedge (\neg x \wedge \neg y) = (x \wedge \neg x \wedge \neg y) \vee (y \wedge \neg x \wedge \neg y)$
  $= 0 \vee 0 = 0$

したがって，$(\neg x \wedge \neg y)$ は $x \vee y$ の補元である

$\neg(x \wedge y) = \neg x \vee \neg y$ ド・モルガンの法則2

- $\neg(x \wedge y) = \neg(\neg(\neg x) \wedge \neg(\neg y))$
  $= \neg(\neg((\neg x) \wedge (\neg y)) = (\neg x) \wedge (\neg y)$

Bool 代数の性質(3)

- 古典命題論理の公理に対応する式が成立する（ただし, **$x \to y := \neg x \lor y$** と定義）

■ $x \to (y \to x) = 1$ for $\forall x, y \in B$

➢ $x \to (y \to x) = \neg x \lor (\neg y \lor x) = \neg y \lor (x \lor \neg x) = \neg y \lor 1 = 1$

■ $(x \to (y \to z)) \to ((x \to y) \to (x \to z)) = 1$ for $\forall x, y, z \in B$

➢ $(x \to (y \to z)) \to ((x \to y) \to (x \to z)) = \neg (x \to (y \to z)) \lor ((x \to y) \to (x \to z))$

$= \neg(\neg x \lor (\neg y \lor z)) \lor (\neg(\neg x \lor y) \lor (\neg x \lor z))$

$= \neg\neg x \land (\neg\neg y \land \neg z) \lor (\neg\neg x \land \neg y) \lor \neg x \lor z$

$= (x \land y \land \neg z) \lor (x \land \neg y) \lor \neg x \lor z$

$= (x \land y \land \neg z) \lor (x \lor \neg x \lor z) \land (\neg y \lor \neg x \lor z)$

$= (x \land y \land \neg z) \lor (\neg y \lor \neg x \lor z)$

$= (x \lor \neg y \lor \neg x \lor z) \land (x \lor \neg y \lor \neg x \lor z) \land (x \lor \neg y \lor \neg x \lor z)$

$= 1 \land 1 \land 1 = 1$

■ $(\neg x \to \neg y) \to (y \to x) = 1$ for $\forall x, y \in B$

➢ $(\neg x \to \neg y) \to (y \to x) = \neg(\neg\neg x \lor \neg y) \lor (\neg y \lor x)$

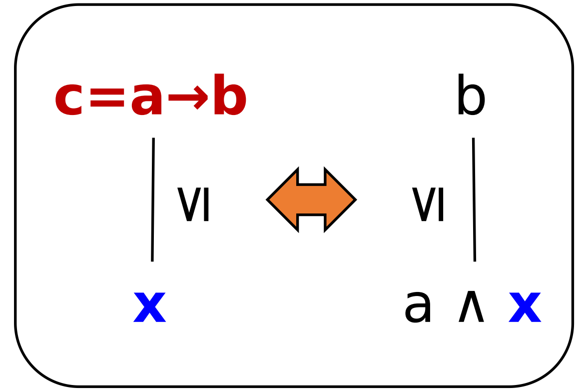
$= (\neg x \land y) \lor (\neg y \lor x)$

$= (\neg x \lor \neg y \lor x) \land (y \lor \neg y \lor x) = 1 \land 1 = 1$

Heyting 代数（Heyting algebra）

- 定義
  - ■ 有界束（最小元と最大元がある束）(H, ∨, ∧, 0, 1) が **ハイティング代数（Heyting algebra）**であるとは H が次の条件を満たす時にいう．
    - ➢ 任意の a, b∈H に対して，ある **c**∈H があって，
      - ➢ **(a ∧ x)≤ b** ⇔ **x≤c　∀x∈H**

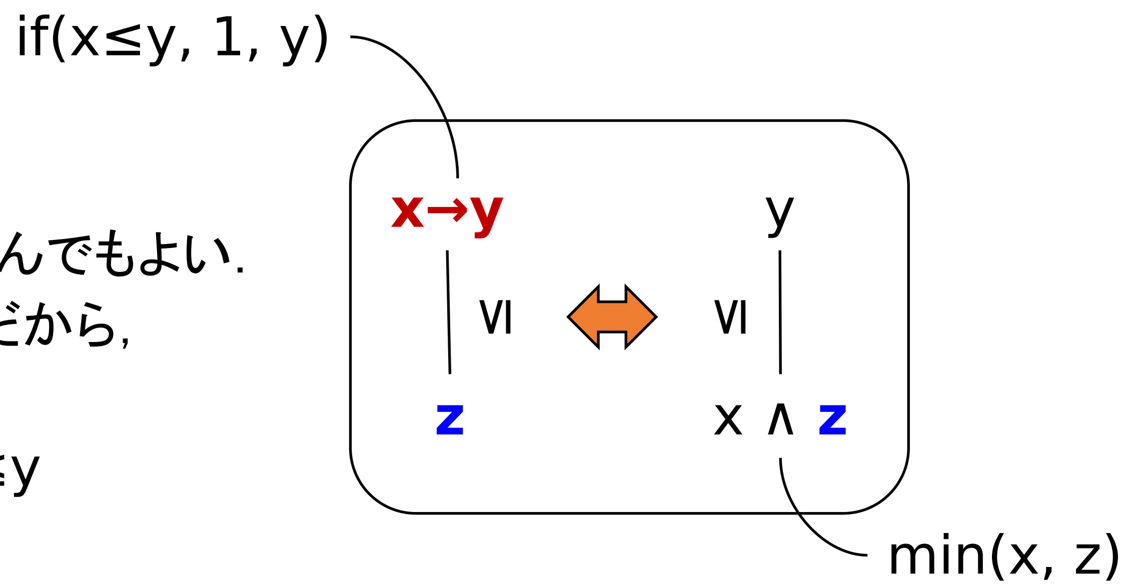
(H, ∨, ∧, 0, 1)



- ■ このような **c が存在すれば**それは a と b に対して **一意に決まる**ので，この **c** を **a→b** と表し，**a と b のimplication** と呼ぶ．（一意性：c と c' がこのような性質を持てば，**c ≤c' かつ c'≤c だから c=c'**）
- ■ Heyting 代数は→があることを強調して **(H, ∨, ∧, →, 0, 1)** と書くことにする
- ■ Heyting 代数に補元があるとは限らないが，補元相当の元として **¬X** := **X→0** で定義することがある
- ■ Bool 代数は**x→y := ¬x ∨ y** とすれば Heyting 代数になる．

Heyting 代数の例（1）

- 有界の全順序集合 (L, ≤, 0, 1) 次の定義で Heyting 代数になる
  - $x \vee y := \max(x, y)$
  - $x \wedge y := \min(x, y)$
  - $x \to y := \mathrm{if}(x \le y, 1, y)$



右図で

➢ **x≤y** なら $x \to y = 1$
　だから，そのときは z はなんでもよい．

➢ **y<x** のときは $x \to y = y$ だから，
　右図の条件は，

$$z \le y \quad \Leftrightarrow \quad \min(x, z) \le y$$

となる．

→) $z \le y$ ならば $\min(x, z) = z$ となり $\min(x, z) \le y$ である

←) $\min(x, z) \le y$ ならば，$x > y$ だから，$z \le y$ となる

Heyting 代数の例（2）

- H = **{0, 1/2, 1}** に，普通の**有理数の順序**を入れれば，これは**全順序集合**だから次の演算で **Heyting 代数**になる
  - X ∨ Y := max(X, Y)
  - X ∧ Y := min(X, Y)
  - X → Y := if(X ≤ Y, 1, Y)

  - ¬X := X→0 と定義すると，
    - **1/2 ∨ ¬1/2 = 1/2 ∨ 0 = 1/2**

    となり，**1 にならない**.

X	Y	X ∨ Y	X ∧ Y	X → Y
0	0	0	0	1
0	1/2	1/2	0	1
0	1	1	0	1
1/2	0	1/2	0	0
1/2	1/2	1/2	1/2	1
1/2	1	1	1/2	1
1	0	1	0	0
1	1/2	1	1/2	1/2
1	1	1	1	1

従って，この Heyting 代数では

x ∨ ¬ x = 1 （排中律）

は必ずしもなりたたない

Heyting 代数の性質

- Heyting 代数 H には，a, b∈H に対して，演算 a→b が定義される

  Def1: **$x \leq a \to b$** ⇔ **$x \wedge a \leq b$**

- 論理演算子「→」の諸性質

■ **$b \leq a \to b$**

  - $b \wedge a \leq b$ であるから，Def1 により，$b \leq a \to b$

■ **$a \to (b \to c) \leq (a \to b) \to (a \to c)$**

  - $(a \to (b \to c)) \wedge (a \to b) \leq a \to c$ を言えば良い
    - 両辺に a を∧で作用させるた次の不等式が言えればよい
      - $(a \wedge (a \to (b \to c))) \wedge (a \wedge (a \to b)) \leq c$
    - 左辺は $(b \to c) \wedge b$ 以下であるから，この式は成立する

Heyting 代数の性質

- **$\neg a \leq a \to b$**
  - $\neg a := a \to 0$ だから，$a \to 0 \leq a \to b$ を言えばよい
    - $x \leq a \to 0 \Leftrightarrow a \wedge x \leq 0$.
    - したがって，$a \wedge (a \to 0) \leq 0 \leq b \Rightarrow a \to 0 \leq a \to b$

- **$a \leq \neg\neg a$**
  - $\neg a := a \to 0$ であるから，$\neg\neg a = (a \to 0) \to 0$
  - $x \leq \neg\neg a \Leftrightarrow x \wedge (a \to 0) \leq 0$
  - この右辺の式に $x=a$ を入れてみると，$a \wedge (a \to 0)$ となり，これは$a \wedge (a \to 0) \leq 0$. 従って，$a \leq \neg\neg a$.

- **$1 = a \to b \quad \Leftrightarrow \quad a \leq b$**

$\Rightarrow$

- $a \wedge (a \to b) \leq b$ だが，$a \to b = 1$ なので $a \leq b$

$\Leftarrow$

- $a \leq b$ なら，どんな x に対しても $x \wedge a \leq b$. 故に $a \to b = 1$

代数を用いた
命題論理の意味論

Bool 代数による
古典命題論理の意味論

- **古典命題論理**の形式的体系 CPL（再掲）
  - **公理**
    - A1. A→(B→A)
    - A2. (A→(B→C))→((A→B)→(A→C))
    - A3. (¬A→¬B)→(B→A)
  - **推論規則**
    - Modus Ponens: A→B と A から B を導く

- **Bool 代数 (B, 0, 1, ∨, ∧, ¬)** による **CPL の解釈**
  - **命題変数の集合**を **Vars**, **命題の集合**を **Props** とする
  - **関数 f : Vars → B** が与えられたとき, **μ_f : Props →B** を次のように定義する
    - **$\mu_f(v) := f(v)$　for $v \in$ Vars**
    - **$\mu_f(A \to B) := \neg\mu_f(A) \vee \mu_f(B)$**
  - **任意の関数 f : Vars → B** に対して **$\mu_f(P)=1$** のとき **B ⊨ P** と書くことにする
  - また, **任意の Bool 代数 B** について, **B ⊨ P** のとき, **⊨ P** と書くことにする

古典命題論理の解釈の性質

- CPC はブール代数に対して，健全性と完全性を持つ

- **健全性**
  - **⊢P ならば，任意の f : Vars→B に対して，$\mu_f(P)$ = 1**

  証明
  - ➢ すべての公理 P に対して$\mu_f(P)$ = 1であることは，Bool 代数の性質として証明した
  - ➢ ルール（MP）はμ_f の値が 1 であることを保存すること
    - ➢ $\mu_f(A)$=1 かつ$\mu_f(A→B)$=1 なら，$\mu_f(B)$ = 1 になる

- **完全性**
  - **⊨ P** (任意のブール代数 B の変数の に対して B⊨P) **なら ⊢P**

  証明
  - ➢ ⊨ P なら，2={0, 1} ⊨ P である.
  - ➢ これは，古典命題論理の完全性で証明したように ⊢ P を意味する

  補足
  - ➢ 健全性と合わせると，⊢ P ならば，⊨ P だから，2 ⊨ P なら ⊨ P になる

Heyting 代数による
直観主義命題論理の意味論

- 直観主義命題論理 IPC（再掲）　　　ただし，**¬A** := **A→⊥**
  - **公理**
    - A1. A→(B→A)
    - A2. (A→(B→C))→((A→B)→(A→C))
    - A3'. ¬A→(A→B)　　　[古典論理：(¬A→¬B)→(B→A)]
  - **推論規則**
    - Modus Ponens: A→B と A から B を導く

- Heyting 代数 (H, ∨, ∧, →, 0, 1) による IPC の**解釈**
  - **f : Vars→H** が与えられたとき，命題 P に対する$\mu_f(P)$ を次のように定義する
    - **$\mu_f(v) := f(v)$　　　for $v \in \mathrm{Vars}$**
    - **$\mu_f(\bot) := 0$**
    - **$\mu(P \to Q) := \mu_f(P) \to \mu_f(Q)$**
  - 任意の f : Vars→H に対して，$\mu_f(P) = 1$ のとき，**H ⊨ P** と書くことにする
  - 任意の Heyting 代数 H に対して H ⊨ P のとき，**⊨ P** と書くことにする

直観主義命題論理の解釈の性質

- 直観主義命題論理 IPC はHeyting代数に対して, **健全性**と**完全性**を持つ

- **健全性**
  - $\vdash P$ ならば, $\models P$

- **完全性**
  - $\models P$ ならば $\vdash P$

- **注意**
  - ここでの $\models P$ は, **すべての Heyting 代数 H に対して, H $\models$ P** が要求されることに注意せよ. **Bool 代数だけとか, H = {0, 1/2, 1} だけ**に対してだけではない
  - **完全性の証明はかなり長く**なり, ここでの範囲を超えているので, **ここでは健全性だけ**証明する

直観主義命題論理 IPC の Heyting 代数に対する健全性の証明

- 健全性：IPC で証明される命題 P は ⊨ P となる

公理について

- 任意の Heyting 代数 H = (H, ∨, ∧, →, 0, 1) に対して
  - A1. H ⊨ A→(B→A)
  - A2. H ⊨ (A→(B→C))→((A→B)→(A→C))
  - A3. H ⊨ ¬A→(A→B)

を言う必要があるが，これらは Heyting 代数の諸性質で述べたものそのものである（例えば，Heyting 代数 H では x→(y→x) = 1 for ∀x, y∈H が常になりたつ）.

推論ルールについて

- $\mu_f(A \to B) = 1$ かつ$\mu_f(A) = 1$ なら$\mu_f(B) = 1$となることを言う
  - $\mu_f(A \to B) = 1$ であるから，$\mu_f(A) \to \mu_f(B) = 1$
  - $x \le \mu_f(A) \to \mu_f(B) \Leftrightarrow x \wedge \mu_f(A) \le \mu_f(B) \Leftrightarrow \mu_f(A) \le \mu_f(B)$
  - $\mu_f(A) = 1$ だから $\mu_f(B) = 1$

まとめ

- これから説明する圏論の**デカルト閉圏**の**応用対象**として**命題論理**とその解釈のための **Bool 代数**, **Heyting 代数**を説明した

- **命題論理**
  - **古典命題論理**
    - 公理と推論規則
    - 二値の Bool 代数 **2**={0, 1}
    - 古典命題論理は, **2**={0, 1} に対して健全で完全なこと
  - **直観主義命題論理**

- **Bool 代数, Heyting 代数**
  - **順序集合(Poset)**から**半束**, **束**, **分配束**と拡張していく代数の系列
  - **Bool 代数** ... Bool 代数による解釈, および, 健全性と完全性
  - **Heyting 代数 ...** Heyting 代数による解釈, および, 健全性と完全性