アンダースタンディング・コンピュテーション　第 6 章
ラムダ計算

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2019 年 10 月 11 日

最終更新：2019 年 10 月 11 日　11:39

スケジュール

2 プログラム意味論 (5月)
3 有限オートマトン (6月)
4 プッシュダウン・オートマトン (7月)
5 チューリングマシン (7月)
6 ラムダ計算 (10月)
7 万能性
8 決定可能性
9 抽象解釈／静的意味論

今からやること

「関数だけ」を用いて，計算を行う

- $\rightsquigarrow$ ラムダ計算

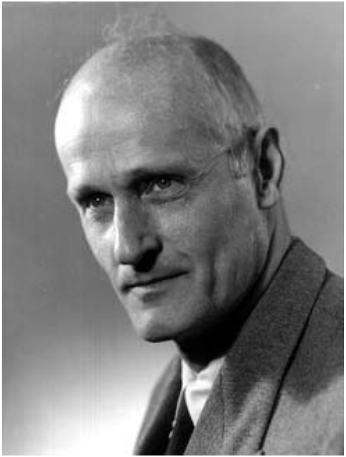
復習

前回まで :「機械」によって，いろいろな計算を行う

- 有限オートマトン
- プッシュダウン・オートマトン
- チューリングマシン

チャーチとクリーネ

Alonzo Church (1903–1995)



Stephen Kleene (1909–1994)

https://www-history.mcs.st-andrews.ac.uk/Biographies/Church.html
http://www-history.mcs.st-and.ac.uk/Biographies/Kleene.html

目次

1. ラムダ式
2. ラムダ式による計算
   - データの表現
   - 条件分岐，述語
   - 数値演算
   - 再帰
3. 個人プロジェクト案の例

ラムダ式：構文

ラムダ式とは？

次のように再帰的に定義される

- 記号 $(x, y, z, \ldots)$ はラムダ式である (変数)
- e がラムダ式であるとき，$\lambda x.e$ もラムダ式である (ラムダ抽象)
- e_1, e_2 がラムダ式であるとき，$e_1\ e_2$ もラムダ式である (関数適用)
- これで定義されるものだけがラムダ式である

例：$((\lambda x.((x\ x)x))\ (\lambda y.y))$

- Ruby の Proc を使って書くと，`-> x { x[x][x] }[-> y { y }]`

注：「$e_1\ e_2\ e_3$」は「$(e_1\ e_2)\ e_3$」を意味する

ラムダ式：スモールステップ意味論

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \to (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \to (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\to ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\to (\lambda y.y)\ (\lambda y.y)$
  $\to (\lambda y.y)$

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \rightarrow (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \rightarrow (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\rightarrow ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\rightarrow (\lambda y.y)\ (\lambda y.y)$
  $\rightarrow (\lambda y.y)$

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \to (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \to (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\to ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\to (\lambda y.y)\ (\lambda y.y)$
  $\to (\lambda y.y)$

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \to (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \to (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\to ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\to (\lambda y.y)\ (\lambda y.y)$
  $\to (\lambda y.y)$

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \to (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \to (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\to ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\to (\lambda y.y)\ (\lambda y.y)$
  $\to (\lambda y.y)$

$\boxed{\text{自由変数の置き換え}}$ (α 変換)

- $(x\ y)\ (\lambda y.(y\ x)) \rightarrow (z\ y)\ (\lambda y.(y\ z))$
- $(x\ y)\ (\lambda y.(y\ x)) \rightarrow (x\ z)\ (\lambda y.(y\ x))$

$\boxed{\text{式の簡約}}$ (β 簡約)

- $(\lambda x.((x\ x)\ x))\ (\lambda y.y)$
  $\rightarrow ((\lambda y.y)\ (\lambda y.y))\ (\lambda y.y)$
  $\rightarrow (\lambda y.y)\ (\lambda y.y)$
  $\rightarrow (\lambda y.y)$

目次

1. ラムダ式
2. ラムダ式による計算
   - データの表現
   - 条件分岐，述語
   - 数値演算
   - 再帰
3. 個人プロジェクト案の例

ブール値

Church boolean と呼ばれる書き方

- $\texttt{TRUE} \;= \lambda x.(\lambda y.x)$
- $\texttt{FALSE} = \lambda x.(\lambda y.y)$

アイディア：「二択」ができればよい

Church numeral と呼ばれる書き方 (1 進法)

- $\mathtt{ZERO} = \lambda p.(\lambda x.x)$
- $\mathtt{ONE} = \lambda p.(\lambda x.(p\ x))$
- $\mathtt{TWO} = \lambda p.(\lambda x.(p\ (p\ x)))$
- $\mathtt{THREE} = \lambda p.(\lambda x.(p\ (p\ (p\ x))))$

アイディア：「p」の数で自然数を表現

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$

- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$
- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda {\color{red}y}.(\lambda f.(f\ a\ y))\ {\color{blue}b}$
  $\to \lambda f.(f\ a\ b)$

- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\texttt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\texttt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\texttt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\texttt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$
- $\texttt{LEFT}\ (\texttt{PAIR}\ a\ b)$
  $\to \texttt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば，

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$

- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$

- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$
- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda y.a)\ b$
  $\to a$

2 個組 (ペア) の実現：Church pair と呼ばれる書き方

- $\mathtt{PAIR} = \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))$
- $\mathtt{LEFT} = \lambda p.(p\ (\lambda x.(\lambda y.x)))$
- $\mathtt{RIGHT} = \lambda p.(p\ (\lambda x.(\lambda y.y)))$

例えば,

- $\mathtt{PAIR}\ a\ b$
  $\to \lambda x.(\lambda y.(\lambda f.(f\ x\ y)))\ a\ b$
  $\to \lambda y.(\lambda f.(f\ a\ y))\ b$
  $\to \lambda f.(f\ a\ b)$
- $\mathtt{LEFT}\ (\mathtt{PAIR}\ a\ b)$
  $\to \mathtt{LEFT}\ (\lambda f.(f\ a\ b))$
  $\to \lambda p.(p\ (\lambda x.(\lambda y.x)))\ (\lambda f.(f\ a\ b))$
  $\to \lambda f.(f\ a\ b)\ \lambda x.(\lambda y.x)$
  $\to \lambda x.(\lambda y.x)\ a\ b$
  $\to (\lambda {\color{red}y}.a)\ {\color{blue}b}$
  $\to a$

If 文 (If-then-else 文)：「If b then x else y」風

- $\mathtt{IF} = \lambda b.(\lambda x.(\lambda y.(b\ x\ y)))$

例

- $\mathtt{IF\ TRUE\ A\ B}$
  $\to \lambda b.(\lambda x.(\lambda y.(b\ x\ y)))\ \mathtt{TRUE\ A\ B}$
  $\to \lambda x.(\lambda y.(\mathtt{TRUE}\ x\ y))\ \mathtt{A\ B}$
  $\to \lambda y.(\mathtt{TRUE\ A}\ y)\ \mathtt{B}$
  $\to \mathtt{TRUE\ A\ B}$
  $\to \lambda x.(\lambda y.x)\ \mathtt{A\ B}$
  $\to \lambda y.\mathtt{A\ B}$
  $\to \mathtt{A}$

- $\mathtt{IF\ FALSE\ A\ B}$
  $\to \mathtt{FALSE\ A\ B}$
  $\to \lambda x.(\lambda y.y)\ \mathtt{A\ B}$
  $\to \lambda y.y\ \mathtt{B}$
  $\to \mathtt{B}$

If 文 (If-then-else 文)：次のようにも定義できる

- $\mathtt{IF} = \lambda b.b$

例

- $\mathtt{IF\ TRUE\ A\ B}$
  $\to \lambda b.b\ \mathtt{TRUE\ A\ B}$
  $\to \mathtt{TRUE\ A\ B}$
  $\to \lambda x.(\lambda y.x)\ \mathtt{A\ B}$
  $\to \lambda y.\mathtt{A\ B}$
  $\to \mathtt{A}$
- $\mathtt{IF\ FALSE\ A\ B}$
  $\to \mathtt{FALSE\ A\ B}$
  $\to \lambda x.(\lambda y.y)\ \mathtt{A\ B}$
  $\to \lambda y.y\ \mathtt{B}$
  $\to \mathtt{B}$

ゼロの判定

- $\mathtt{IS\_ZERO} = \lambda n.(n\ (\lambda x.\mathtt{FALSE})\ \mathtt{TRUE})$

例

- $\mathtt{IS\_ZERO}\ \mathtt{ZERO}$
  $\to (\lambda n.(n\ (\lambda x.\mathtt{FALSE})\ \mathtt{TRUE}))\ (\lambda p.(\lambda y.y))$
  $\to \lambda p.(\lambda y.y)\ (\lambda x.\mathtt{FALSE})\ \mathtt{TRUE}$
  $\to (\lambda y.y)\ \mathtt{TRUE}$
  $\to \mathtt{TRUE}$
- $\mathtt{IS\_ZERO}\ \mathtt{TWO}$
  $\to (\lambda n.(n\ (\lambda x.\mathtt{FALSE})\ \mathtt{TRUE}))\ (\lambda p.(\lambda y.(p\ (p\ y))))$
  $\to (\lambda p.(\lambda y.(p\ (p\ y)))\ (\lambda x.\mathtt{FALSE}))\ \mathtt{TRUE}$
  $\to (\lambda y.((\lambda x.\mathtt{FALSE})\ ((\lambda x.\mathtt{FALSE})\ y)))\ \mathtt{TRUE}$
  $\to (\lambda x.\mathtt{FALSE})\ ((\lambda x.\mathtt{FALSE})\ \mathtt{TRUE})$
  $\to \mathtt{FALSE}$

インクリメント：1 だけ足す

- $\texttt{INCREMENT} = \lambda n.(\lambda p.(\lambda x(p\ (n\ p\ x))))$

例

- $\texttt{INCREMENT TWO}$
  $\to \lambda n.(\lambda p.(\lambda x.(p\ (n\ p\ x))))\ \texttt{TWO}$
  $\to \lambda p.(\lambda x.(p\ (\texttt{TWO}\ p\ x)))$
  $\to \lambda p.(\lambda x.(p\ ((\lambda q.(\lambda y.(q\ (q\ y))\ p\ x)))$
  $\to \lambda p.(\lambda x.(p\ (p\ (p\ x))))$
  $\to \texttt{THREE}$

加算，乗算，指数演算

- $\texttt{ADD} = \lambda m.(\lambda n.(n\ \texttt{INCREMENT}\ m))$
- $\texttt{MULTIPLY} = \lambda m.(\lambda n.(n\ (\texttt{ADD}\ m)\ \texttt{ZERO}))$
- $\texttt{POWER} = \lambda m.(\lambda n.(n\ (\texttt{MULTIPLY}\ m)\ \texttt{ONE}))$

減算はちょっと難しい (いまのところ，負の数を扱えないため)

- $\texttt{SLIDE} = \lambda p.(\texttt{PAIR}\ (\texttt{RIGHT}\ p)\ (\texttt{INCREMENT}\ (\texttt{RIGHT}\ p)))$
- $\texttt{DECREMENT} = \lambda n.(\texttt{LEFT}\ (n\ \texttt{SLIDE}\ (\texttt{PAIR}\ \texttt{ZERO}\ \texttt{ZERO})))$
- $\texttt{SUBTRACT} = \lambda m.(\lambda n.(n\ \texttt{DECREMENT}\ m))$

大小比較

- $\texttt{IS\_LESS\_OR\_EQUAL} = \lambda m.(\lambda n.(\texttt{IS\_ZERO}\ \texttt{SUBTRACT}\ m\ n))$

ラムダ計算で再帰を扱うことができる

ここでは，Z コンビネータ を使ってみる

- $\mathtt{Z} = \lambda f.(\lambda x.(f\ (\lambda y.(x\ x\ y)))\ (\lambda x.(f\ (\lambda y.(x\ x\ y))))$

ラムダ式 MOD を再帰的に定義してみる

- $\mathtt{MOD} = \mathtt{Z}\ \lambda f.(\lambda m.(\lambda n.($
  $\quad \mathtt{IF}\ (\mathtt{IS\_LESS\_OR\_EQUAL}\ n\ m)\ (\lambda x.(f\ (\mathtt{SUBTRACT}\ m\ n)\ n\ x))\ m)))$

これは次の再帰式に基づく

$$m \bmod n = \begin{cases} (m-n) \bmod n & (n \leq m \text{ のとき}) \\ m & (\text{その他のとき}) \end{cases}$$

同様に，再帰を用いて (自然数の) 除算を書くと次のようになる

$$m/n = \begin{cases} 1 + (m-n)/n & (n \leq m \text{ のとき}) \\ 0 & (\text{その他のとき}) \end{cases}$$

Z コンビネータを使って，除算をラムダ式として書く

- $\mathtt{DIV} = \mathtt{Z}\ \lambda f.(\lambda m.(\lambda n.($
  $\mathtt{IF}$
  $(\mathtt{IS\_LESS\_OR\_EQUAL}\ n\ m)$
  $(\lambda x.(\mathtt{INCREMENT}\ (f\ (\mathtt{SUBTRACT}\ m\ n))\ n\ x))$
  $\mathtt{ZERO}$
  $)))$

テキストでは，次もラムダ式として実装している

- リスト，および，リストに対する操作
- 文字列
- (無限) ストリーム

目次

個人プロジェクト案の例

- ラムダ計算を用いて，基本的なプログラムを書いてみる
  例えば，ソーティングのプログラムを書いてみる
- 負の数も扱えるようにする
  例えば，PAIR を使って負の数を表現する
  負の数も考慮して，四則演算ができるようにする
- 自然数を 2 進 (binary) で表現し，四則演算もできるようにする
  Church numeral では「p の数」で自然数を表現したが，
  例えば，2 進表現で「10110」である自然数を
  $\lambda p.(\lambda q.(\lambda x(p\ q\ p\ p\ q\ x)))$ のような形式で表現できるか？