Practical type theory

lecture 1: introduction into $\lambda - calculus$

Воронов Михаил

ВМК МГУ, Fall 2022

План лекции

1. Устройство курса
2. Введение в λ-исчисление
3. Подстановка и преобразования

План лекции

1. Устройство курса
2. Введение в λ-исчисление
3. Подстановка и преобразования

План курса

1. Simple untyped λ-calculus
2. $\lambda \to$
3. $\lambda 2$
4. $\lambda \omega$
5. λP
6. λC
7. λD
8. Основы MLTT
9. Основы HOTT
10. Основы CubicTT
11. Введение в практическое использование Coq и Agda

Критерий оценивания

На данный момент планируется 6 домашних заданий, за каждую можно получить 40-50 баллов. Также за экзамен можно получить до 150 баллов, критерии оценивания:

1. [0 – 79] - 2
2. [80 – 149] - 3
3. [150 – 199] - 4
4. [200 – ..] - 5

Литература

1. Rob Nederpelt, Herman Geuvers "Type theory and formal proof"
2. Samuel Mimram "Program = proof"
3. Yves Bertot "Interactive program proving and program development"
4. Курс Москвитина Дениса Николаевича "Функциональное программирование" *youtube*

План лекции

1 Устройство курса

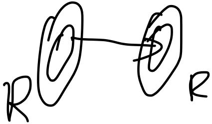
2 Введение в λ-исчисление

3 Подстановка и преобразования

λ-исчисление

- λ-исчисление - формальная система, разработанная Алонзо Чёрчем в 1930-х для формализации и анализа понятия вычислимости
- Имеет бестиповую (simple untyped) и множество типовых версий
- Позволяет описывать семантику вычислительных процессов
- Является теоретической основой для многих пруверов

Поведение функций



Рассмотрим функцию $f(x) : x^2 + 1$

- эта функция имеет один "вход"(другими словами, зависит от одной переменной) и один "выход": $f : \mathbb{R} \to \mathbb{R}$
- в некотором смысле, эту функцию можно рассматривать, как отображение $x \to x^2 + 1$
- чтобы подчеркнуть "абстрактную"роль x, используют специальный символ λ: $\lambda x.(x^2 + 1)$
- данная нотация выражает то, что x - это не конкретное число, а некоторая абстракция
- для конкретного значения можно "применить"данную функцию: $(\lambda x.x^2 + 1)(3)$

def foo (y)
return x^2+1

foo(3)

Введение в λ-исчисление

Из предыдущего слайда следует, что для работы с функциями достаточно двух способов построения выражений:

1. Абстракция: из выражения M и переменной x можно составить новое выражение $\lambda x.M$ (абстракция x по M)
2. Применение (application): из двух выражений M и N можно составить новое выражение MN

Введение в λ-исчисление: абстракция

1. Пусть $M = M[x]$ - выражение, возможно содержащее x
2. Тогда абстракция $\lambda x.M$ обозначает функцию $x \to M[x]$
3. Абстракция - способ задать неименованную функцию функцию
4. Если x в $M[x]$ отсутствует, то $\lambda x.M$ - константная функция со значением M.

Введение в λ-исчисление: применение

1. С точки зрения разработки ПО, применение F к X - это применение алгоритма (F) к данным X
2. Однако явного различия между алгоритмами и данными нет, в частности, возможно самоприменение: FF
3. В общем случае применение - это так называемая β-эквивалентность:

$$(\cancel{\lambda x}.M)N =_\beta M[x := N]$$

4. $M[x := N]$ - это M, в котором вместо N подставлено вместо x

Введение в λ-исчисление: β-редукция

1. $(\lambda x.x^2 + 1)(3) = (x^2 + 1)[x := 3] = 3^2 + 1$
2. $(\lambda y.5)(1) = 5[x := 1] = 5$
3. $(\lambda x.x)(\lambda y.y) = x[x := (\lambda y.y)] = (\lambda y.y)$
4. $\lambda z.((\lambda x.x)(\lambda y.y)) = \lambda z.(x[x := (\lambda y.y])) = \lambda z.(\lambda y.y)$

Термы λ-исчисления

$xy \quad (xy)$

Definition

Множество λ-термов Λ определяется индуктивно из переменных $V = x, y, z, \dots$:

- $x \in V \to x \in \Lambda$
- $\underline{M, N \in \Lambda \to (MN)} \in \Lambda$
- $M \in \Lambda, x \in V \to (\lambda x.M) \in \Lambda$

$M = (\lambda x.\ x y)$

$N = z$

$MN = ((\lambda x. xy) z) =$
$= (zy) = zy$

- В абстрактном синтаксисе:

$$\Lambda ::= V|(\Lambda\Lambda)|(\lambda V.\Lambda)$$

- Произвольные термы будем обозначать заглавными буквами, а переменные - строчными

Примеры термов

- x
- (xz)
- $(\lambda x.(xz))$
- $((\lambda x.(xz))y)$
- $(\lambda y.((\lambda x.(xz))y))$
- $((\lambda y.((\lambda x.(xz))y))w)$
- $(\lambda z.(\lambda w.((\lambda y.((\lambda x.(xz))y))w)))$

Термы (соглашения)

Общеприняты следующие соглашения:

- Внешние скобки опускаются
- Применение левоассоциативно:

$$FXYZ \text{ обозначает } (((FX)Y)Z)$$

- Абстракция правоассоциативна:

$$\lambda xyz.M \text{ обозначает } (\lambda x.(\lambda y.(\lambda z.(M))))$$

- Тело абстракции простирается вправо насколько это возможно

$$\lambda x.MNK \text{ обозначает } \lambda x.(MNK)$$

$\lambda x.\lambda y.\lambda z.M$

$\lambda xyz.M$

$\backslash x \backslash y \backslash z.M$

Примеры термов

- $x = x$
- $(xz) = xz$
- $(\lambda x.(xz)) = \lambda x.xz$
- $((\lambda x.(xz))y) = (\lambda x.xz)y$
- $(\lambda y.((\lambda x.(xz))y)) = \lambda y.(\lambda x.xz)y$
- $((\lambda y.((\lambda x.(xz))y))w) = (\lambda y.(\lambda x.xz)y)w$
- $(\lambda z.(\lambda w.((\lambda y.((\lambda x.(xz))y))w))) = \lambda zw.(\lambda y.(\lambda x.xz)y)w$

Свободные и связанные переменные

Абстракция $\lambda x.M[x]$ связывает дотоле свободную переменную x в терме M.

Example

$$(\lambda y.(\lambda x.xz)y)w$$

Переменные x и y - связанные, а z и w - свободные

Example

$$(\lambda x.(\lambda x.xz)x)x$$

?

Свободные переменные

Definition

Множество $FV(T)$ **свободных переменных** в терме T определяется рекурсивно:

$$FV(x) = \{x\}$$

$$FV(MN) = FV(M) \cup FV(N);$$

$$FV(\lambda x.N) = FV(\cancel{M}) \setminus x;$$

Связанные переменные

Definition

Множество $BV(T)$ **связанных переменных** в терме T определяется рекурсивно:

$$BV(x) = \{\varnothing\};$$

$$BV(MN) = BV(M) \cup BV(N);$$

$$BV(\lambda x.N) = BV(\overset{N}{\cancel{M}}) \cup x;$$

Комбинаторы

Definition

Комбинатор (замкнутый λ-терм) M - это такой λ-терм, что $FV(M) = \varnothing$. Множество всех замкнутых термов обозначается Λ^0.

- $I = \lambda x.x$
- $\omega = \lambda x.xx$
- $\Omega = \omega\ \omega = (\lambda x.xx)(\lambda x.xx)$
- $K = \lambda xy.x$
- $K_* = \lambda xy.y$
- $S = \lambda fgx.fx(gx)$
- $B = \lambda fgx.f(gx)$

$B\ F\ G\ 5 = F(G(5))$

План лекции

α-преобразование и α-эквивалентность

$\lambda x.x \quad \lambda y.y$

$\lambda x.xy \quad \lambda y.yx$

Definition

α-преобразование и α-эквивалентность $=_\alpha$ - это отношение, удовлетворящее следующим свойствам: $\lambda x.\, x\,(\lambda x.x)$

1. (Переименование) $\lambda x.M =_\alpha \lambda y.M^{\underline{x \to y}}$ free
2. (Сопоставимость) $\in V$
   $M =_\alpha N \to ML =_\alpha NL, LM =_\alpha LN, \forall x \lambda x.M =_\alpha = \lambda x.N$
   $x \notin FV(M)$
3. (Рефлексивность) $M =_\alpha M$ $\notin FV(N)$
4. (Симметричность) $M =_\alpha N \to N =_\alpha M$
5. (Транзитивность) $M =_\alpha N, N =_\alpha L \to M =_\alpha L$

$\lambda xy.xzy \qquad \lambda xy.xzy \qquad \lambda xy.(zzy)[z:=x]$

$\lambda vu.vzu \qquad \lambda zy.zzy =_\alpha \lambda xy.xxy$

α-преобразование и α-эквивалентность

Definition

α**-преобразование и** α**-эквивалентность** $=_\alpha$ - это отношение, удовлетворящее следующим свойствам:

1. (Переименование) $\lambda x.M =_\alpha \lambda y.M^{x \to y}$
2. (Сопоставимость) $M =_\alpha N \to ML =_\alpha NL, LM =_\alpha LN, \forall x \lambda x.M =_\alpha = \lambda x.N$
3. (Рефлексивность) $M =_\alpha M$
4. (Симметричность) $M =_\alpha N \to N =_\alpha M$
5. (Транзитивность) $M =_\alpha N, N =_\alpha L \to M =_\alpha L$

Подстановка $M[x := N]$

$M \in \Lambda,\ N \in \Lambda \Rightarrow (MN) \in \Lambda$

1а $x[x := N] \equiv N$

1б $y[x := N] \equiv y \quad x \not\equiv y$

2. $(PQ)[x := N] \equiv (P[x := N])(Q[x := N])$

3. $(\lambda y . P)[x := N] \equiv \lambda z . (P^{y \to z}[x := N])$

если $\lambda z . P^{y \to z} =_\alpha \lambda y . P \quad z \notin FV(N)$

4. $(\lambda x . P)[x := N) = (\lambda x . P)$

$\lambda y . y (x y) [x := y] = \lambda y . y (y y)$

$x N$

$(\lambda x . x) N = N$

$y \notin FV(N)$

$FV(N)$

$y \in FV(N)$

$\lambda xy.(xyx)[x = \lambda y.y] = \lambda xy.(xyx)$

$x \in V \quad P \in \Lambda =$

$(\lambda x.P) \in \Lambda$

$(\lambda y.yx)[x := xy] =_\alpha (\lambda z.zx)[x := xy] =$

$= (\lambda z.(zx)[x := xy])$

$= (\lambda z.(z[x := xy])(x[x := xy])$

$= (\lambda z.z(xy))$

$= \lambda z.z(xy)$

$zxy = ((zx)y) \neq z(xy)$

$(\lambda x.x^2 + 1)1$

$\lambda xyz.xyxz = \underbrace{((\lambda x.(\lambda y.(\lambda z.xyxz)))t)}_{\text{redex}} =$

$(\lambda y.(\lambda z.tytz))$

$\underbrace{(\lambda x. P)\, t}_{\text{redex}} =_\beta P[x := t]$

$=_\alpha$ $\quad (\ =_\beta$

$\underline{(\lambda x. P)}\, t \neq \lambda x. Pt$

$(\lambda x. (\lambda y. (\lambda z. x y x z)))\, t$

$(\lambda x. (\lambda y. P)) = \lambda x y. P$ $\quad \lambda x \in \Lambda$

$x y$

x

$x y x \cancel{z}$

$(\lambda x. P)\, t = P[x := t]$

$P = y z \Rightarrow y z \neq y z t$

$P = x y z \Rightarrow t y z \neq x y z t$

$[x\, y\, (\lambda zy.\, zx\, (wx)\, y)][x := w\, (\lambda x.\, wx)] =$

$= (w(\lambda x.\, wx))\, y\, (\lambda zy.\, z\, (w\, (\lambda x.\, wx))$

$(w\, (w\, (\lambda x.\, wx)))\, y)$

1a $x[x := N] = N$

1б $y[x := N] = y$

2 $(PQ)[x := N]$

$P = [x\, y\, (\lambda zy.\, zx\, (wx)\, y)][\cancel{z} := w\, (\lambda x.\, wx)]$

N

$P[\cancel{z} := N] = P$

$x\, y\, (\lambda zy.\, zx\, (wx)\, y)\quad [x := w(\lambda x.\, yx)]$

$z = 10$

```
def foo (x, y, z):
    return x·y + z
```

$\Rightarrow \lambda xy.\, (x \cdot y + \cancel{z})$

Кодировка Чёрча

$true = \lambda t f.\ t = K$

$false = \lambda t f.\ f = K_*$

$if = \lambda b x y.\ b x y$

if (boolean)
x
else
y

$if\ true = (\lambda b x y.\ b x y)(\lambda t f.\ t) = \lambda x y.\ (\lambda t f.\ t)\ x y = \lambda x y.\ x =$

$if\ false = (\lambda b x y.\ b x y)(\lambda t.f.\ f) = \lambda x y.\ (\lambda t f.\ f)\ x y = \lambda x y.\ y =$

$\lambda x y.\ x$ $\lambda x y.\ y$

$if\ true\ A B = A$

$if\ false\ A B = B$

$\lambda b x y.\ b x y = \underline{(\lambda b(\lambda x(\lambda y.\ b x y)))}$

$(\lambda x.\ P)\ t = P[x := t]$

$(\lambda b\ \underbrace{(\lambda x\ (\lambda y.\ b x y))}_{P})\ t = (\lambda b.\ P)\ t =$

$= P[b := t]$

$\text{and} = \lambda xy.\ x\,y\ \text{false}$

$\text{and true true} = ((\lambda xy.\ x\,y\ \text{false})\ \text{true})\ \text{true} = \text{true true false} =$

$= (\lambda tf.\ t)(\lambda tf.\ t)(\lambda tf.\ f) = (\lambda f.\ (\lambda tf.\ t))(\lambda tf.\ f) =$

$= \lambda tf.\ t = \text{true}$

$\text{or} \equiv ?$

$\lambda xyz.\ P = (\lambda x.\ (\lambda y.\ (\lambda z.\ P)))$

$x\,y\,z = ((x\,y)\,z)$

право асс-ть

лево асс-ть

application

применение

$\text{or} = \lambda xy.\ x\ \text{true}\ y$

$\text{or false false} = \text{false true false}$

$(x^2+1)(3)$

$(\lambda xy.P)\,Q =$

$(\lambda x.\ P)\ Q$

$[(\lambda x.\ P)\ x]\ y =$

$[(\lambda x.\ P)\ (x\ y)]$