Applied type theory

lecture 1: introduction into $\lambda - calculus$

Воронов Михаил

ВМК МГУ, Fall 2024

План лекции

План лекции

Теоретический план курса

1. Simple untyped λ-calculus
2. $\lambda \rightarrow$
3. Алгебра типов
4. Классические и неклассические логики
5. $\lambda 2$
6. $\lambda \omega$
7. λP
8. λC
9. Основы MLTT
10. Основы HOTT
11. Основы CubicTT

Практический план курса

1. Задачи на всевозможный λ-calculus
2. Coq
3. Немного Agda, TLA+, Ocaml

Критерий оценивания

На данный момент планируется 6 домашних заданий, за каждый можно получить 40-50 баллов. Также за экзамен можно получить до 150 баллов, критерии оценивания (могут немного поменяться к концу семестра) за баллы по обязательным задачам:

1. $< 55\%$ - 2
2. $55\% - 75\%$ - 3
3. $75\% - 95\%$ - 4
4. $\geq 95\%$ - 5

Литература

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

План лекции

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: $(\lambda x.x^2 + 1)(3)$

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
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:

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

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

Definition

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

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

- 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

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

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 свободную до этого 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:

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

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

$$FV(\lambda x.M) = FV(M) \setminus x;$$

Связанные переменные

Definition

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

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

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

$$BV(\lambda x.M) = BV(M) \cup x;$$

Комбинаторы

Definition

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

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

План лекции

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 терма

Definition

Обозначим через **M[x := N]** подстановку N вместо свободных вхождений x в M, которая подчиняется следующим правилам

1. $x[x := N] = N$
2. $y[x := N] = y$
3. $(PQ)[x := N] = (P[x := N])(Q[x := N])$
4. $(\lambda y.P)[x := N] = \lambda y.(P[x := N]), y \notin FV(N)$
5. $(\lambda x.P)[x := N] = (\lambda x.P)$

η-преобразование

Definition

Обозначим через η преобразование следующего вида:
$(\lambda x.M)x = M, x \notin FV(M)$