Теоретические домашние задания

Теория типов, ИТМО, M3235-M3239, осень 2021 года

Домашнее задание №1: «вводная лекция»

1. Напомним правила расстановки скобок в лямбда-выражениях. Лямбда-абстракция ведёт себя жадно: включает всё, что может. Пример: $\lambda z.(\lambda x.a\ b\ c\ \lambda y.d\ e)\ f$ эквивалентно $\lambda z.((\lambda x.(a\ b\ c\ (\lambda y.(d\ e))))\ f)$. В аппликациях скобки расставляются слева направо: $\lambda z.(\lambda x.(a\ b\ c\ (\lambda y.(d\ e))))\ f$ можно преобразовать в $(\lambda z.((\lambda x.((((a\ b)\ c)\ (\lambda y.(d\ e)))))\ f))$.

   (a) Расставьте скобки в выражении: $\lambda z.\lambda x.a\ b\ c\ \lambda y.d\ e\ f$

   (b) Уберите все «лишние» скобки из выражения: $(\lambda f.((\lambda x.(f\ (f\ (x\ (\lambda z.(z\ x))))))\ z))$

   (c) Всегда ли лишние скобки можно убрать единственным образом (когда из результирующего выражения нельзя больше убрать ни одной пары скобок)? Докажите или опровергните.

2. Напомним определения с лекций:

Обозначение	лямбда-терм	название
T	$\lambda a.\lambda b.a$	истина
F	$\lambda a.\lambda b.b$	ложь
Not	$\lambda x.x\ F\ T$	отрицание
And	$\lambda x.\lambda y.x\ y\ F$	конъюнкция

   Проредуцируйте следующие выражения и найдите нормальную форму:

   (a) $T\ F$

   (b) $(T\ Not\ (\lambda t.t))\ F$

   (c) $And\ (And\ F\ F)\ T$

3. Постройте лямбда-выражения для следующих булевских выражений:

   (a) Дизъюнкция

   (b) Штрих Шеффера («и-не»)

   (c) Исключающее или

4. Напомним определения с лекций:

$$f^{(n)}\ X ::= \begin{cases} X, & n = 0 \\ f^{(n-1)}\ (f\ X), & n > 0 \end{cases}$$

Обозначение	лямбда-терм	название
$\overline{n}$	$\lambda f.\lambda x.f^{(n)}\ x$	чёрчевский нумерал
$(+1)$	$\lambda n.\lambda f.\lambda x.n\ f\ (f\ x)$	прибавление 1
$(+)$	$\lambda a.\lambda b.a\ (+1)\ b$	сложение
$(\cdot)$	$\lambda a.\lambda b.a\ ((+)\ b)\ \overline{0}$	умножение

   Используя данные определения, постройте выражения для следующих операций над числами:

   (a) Умножение на 2 ($Mul2$)

   (b) Возведение в степень

   (c) Проверка на чётность

   (d) IsZero: возвращает T, если аргумент равен нулю, иначе F

5. Напомним определения с лекций:

Обозначение	лямбда-терм	название
$MkPair$	$\lambda a.\lambda b.(\lambda x.x\ a\ b)$	создание пары
PrL	$\lambda p.p\ T$	левая проекция
PrR	$\lambda p.p\ F$	правая проекция

   (a) Убедитесь, что $PrL\ (MkPair\ a\ b) \twoheadrightarrow_\beta a$.

(b) Постройте операцию вычитания 1 из числа

(c) Постройте операцию вычитания чисел

(d) Постройте операцию деления на 3 (могут потребоваться пары и/или вычитания)

(e) Постройте операцию деления чисел

(f) Сравнение двух чисел ($IsLess$) — истина, если первый аргумент меньше второго.

6. Существует ли выражение A, что существуют такие выражения B и C, что $A \to_\beta B$ и $A \to_\beta C$, но B и C различны?

7. Будем говорить, что лямбда-выражение находится в нормальной форме, если в нём невозможно провести бета-редукции. Нормальной формой выражения называется результат (возможно, многократной) его бета-редукции, находящийся в нормальной форме. Проредуцируйте выражение и найдите его нормальную форму:

   (a) $\overline{2}\ \overline{2}$

   (b) $\overline{2}\ \overline{2}\ \overline{2}$

   (c) $\overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}$

8. Напомним определение Y-комбинатора: $\lambda f.(\lambda x.f\ (x\ x))\ (\lambda x.f\ (x\ x))$. Напомним, что отношение бета-эквивалентности ($=_\beta$) есть транзитивное, рефлексивное и симметричное замыкание отношения бета-редукции. Будем говорить, что выражение не имеет нормальной формы, если никакая конечная последовательность его бета-редукций не приводит к выражению в нормальной форме.

   (a) Покажите, что $Y\ f =_\beta f\ (Y\ f)$.

   (b) Покажите, что выражение $Y\ f$ не имеет нормальной формы;

   (c) Покажите, что выражение $Y\ (\lambda f.\overline{0})$ имеет нормальную форму.

   (d) Покажите, что выражение $Y\ (\lambda f.\lambda x.(IsZero\ x)\ \overline{0}\ (f\ Minus1\ x))\ 2$ имеет нормальную форму.

   (e) Какова нормальная форма выражения $Y\ (\lambda f.\lambda x.(IsZero\ x)\ \overline{0}\ ((+1)\ (f\ (Minus1\ x))))\ \overline{n}$?

   (f) Какова нормальная форма выражения $Y\ (\lambda f.\lambda x.(IsZero\ x)\ \overline{1}\ (Mul2\ (f\ (Minus1\ x))))\ \overline{n}$?

   (g) Определите с помощью Y-комбинатора функцию для вычисления n-го числа Фибоначчи.

9. Определим на языке Хаскель следующую функцию: `show_church n = show (n (+1) 0)` Убедитесь, что `show_church (\f -> \x -> f (f x))` вернёт 2. Пользуясь данным определением и его идеей, реализуйте следующие функции:

   (a) `int_to_church` — возвращает чёрчевский нумерал (т.е. функцию от двух аргументов) по целому числу. Каков точный тип результата этой функции?

   (b) сложение двух чёрчевских нумералов.

   (c) умножение двух чёрчевских нумералов.

   (d) можно ли определить функцию вычитания 1 и вычитания двух чисел? Что получается, а что — нет?

10. Бесконечное количество комбинаторов неподвижной точки. Дадим следующие определения

$$L := \lambda abcdefghijklmnopqstuvwxyzr.r(thisisafixedpointcombinator)$$
$$R := LLLLLLLLLLLLLLLLLLLLLLLLLL$$

   В данном определении терм R является комбинатором неподвижной точки: каков бы ни был терм F, выполнено $R\ F =_\beta F\ (R\ F)$.

   (a) Докажите, что данный комбинатор — действительно комбинатор неподвижной точки.

   (b) Пусть в качестве имён переменных разрешены русские буквы. Постройте аналогичное выражение по-русски: с 32 параметрами (без ё) и осмысленной русской фразой в терме L; покажите, что оно является комбинатором неподвижной точки.

11. Дадим определение: комбинатор — лямбда-выражение без свободных переменных.

    Также напомним определение:

    $$S := \lambda x.\lambda y.\lambda z.x\ z\ (y\ z)$$
    $$K := \lambda x.\lambda y.x$$
    $$I := \lambda x.x$$

    Известна теорема о том, что для любого комбинатора X можно найти выражение P (состоящее только из скобок, пробелов и комбинаторов S и K), что $X =_\beta P$. Будем говорить, что комбинатор P *выражает* комбинатор X в базисе SK.

    Выразите в базисе SK:

    (a) $F = \lambda x.\lambda y.y$
    (b) $\overline{1}$
    (c) Not
    (d) Xor
    (e) InL
    (f) $\overline{n}$

Домашнее задание №2: «формализация лямбда-исчисления»

1. Придумайте грамматику для лямбда-выражений, однозначно разбирающую любое выражение (в частности, учитывающую все сокращения скобок в записи).

2. Приведите пример лямбда-выражения, корректная бета-редукция которого невозможна без переименования связанных переменных. Возможно ли, чтобы в этом выражении все переменные в лямбда-абстракциях были различными?

3. Два выражения A и B назовём родственными, если существует C, что $A \twoheadrightarrow_\beta C$ и $B \twoheadrightarrow_\beta C$. Как соотносится родственность и бета-эквивалентность?

4. Рассмотрим представление лямбда-выражений де Брауна (de Bruijn): вместо имени связанной переменной будем указывать число промежуточных лямбда-абстракций между связывающей абстрацией и переменной. Например, $\lambda x.\lambda y.y\ x$ превратится в $\lambda.\lambda.0\ 1$.

    Докажите, что $A =_\alpha B$ тогда и только тогда, когда представления де Брауна для A и B совпадают. Сформулируйте правила (алгоритмы) для подстановки термов и бета-редукции для этого представления.

5. Как мы знаем, $\Omega \to_\beta \Omega$. А существуют ли такие лямбда-выражения A и B ($A \neq_\alpha B$), что $A \to_\beta B$ и $B \to_\beta A$?

6. Рассмотрим следующие лямбда-выражения для задания алгебраических типов:

Обозначение	лямбда-терм	название
$Case$	$\lambda l.\lambda r.\lambda c.c\ l\ r$	сопоставление с образцом
InL	$\lambda l.(\lambda x.\lambda y.x\ l)$	левая инъекция
InR	$\lambda r.(\lambda x.\lambda y.y\ r)$	правая инъекция

    Сопоставление с образцом — это функция от значения алгебраического типа и двух действий l и r, которая выполняет действие l, если значение создано «левым» конструктором, и r в случае «правого» конструктора. Иными словами, $Case\ l\ r\ c$ — это аналог `case c { InL x -> l x; InR x -> r x }`.

    Заметим, что список (например, целых чисел) — это алгебраический тип:

    `List = Nil | Cons Integer List`.

    Можно сконструировать значение данного типа: `Cons 3 (Cons 5 Nil)`. Можно, например, вычислить его длину:

    ```
    length Nil = 0
    length (Cons _ tail) = length tail + 1
    ```

    Определим $Nil = InL\ 0$, а $Cons\ a\ b = InR\ (MkPair\ a\ b)$. Заметим, что теперь списки могут быть впрямую перенесены в лямбда выражения.

    Определите следующие функции в лямбда-исчислении для списков:

(a) вычисление длины списка;

(b) построение списка длины n из элементов $0, 1, 2, \ldots, n-1$;

(c) разворот списка: из списка $a_1, a_2, \ldots, a_n$ сделать список $a_n, a_{n-1}, \ldots, a_1$;

(d) функцию высшего порядка *map*, которая по функции f и списку $a_1, a_2, \ldots, a_n$ строит список $f(a_1), f(a_2), \ldots, f(a_n)$.

Решением задачи является полный текст соответствующего лямбда-выражения с объяснениями механизма его работы. Используйте интерпретатор лямбда-выражений *lci* или аналогичный для демонстрации результата.

7. Чёрчевские нумералы соответствуют натуральным числам в аксиоматике Пеано.

(a) Предложите «двоичные нумералы» — способ кодирования чисел, аналогичный двоичной системе (такой, при котором длина записи числа соответствует логарифму числового значения).

(b) Предложите реализацию функции $(+1)$ в данном представлении.

(c) Предложите реализацию лямбда-выражения преобразования числа из двоичного нумерала в чёрчевский.

Аналогично прошлому заданию, решение должно содержать полный код лямбда-выражения вместе с объяснением механизма его работы.