

Функциональное программирование. Семинар 1.

Подстановка и бета-преобразование.

Выполните подстановки:

```
(z y (\x -> x y) (\y -> y z) x)[y := v u]
(z y (\x -> x y) (\y -> y z) x)[z := y]
(z y (\x -> x y) (\y -> y z) x)[x := \v -> w (v t)]
(z y (\x -> x y) (\y -> y z) x)[z := \v -> w (v t)]
```

Можно ли после этого сделать в терминах бета-преобразование? Если да, выполните его:

Покажите, что:

```
I = S K K
K* = K I
```

Пары

Попробуем построить простенький язык программирования с помощью лямбда-исчисления. Для этого нам понадобится научиться кодировать различные штуки -- кортежи, натуральные числа, булевы значения, списки и так далее. Кроме этого, мы научимся определять функции над сущностями, которые мы определили ранее.

Начнем с кортежей. Пару можно закодировать следующим образом:

```
pair = \x y f -> f x y
```

Для пары нам понадобятся проекции. То есть функции, возвращающие первый и второй элемент пары. Реализуйте их:

```
p1 = \p -> p K
p2 = \p -> p K*
```

Булевы значения

Булевы значения можно закодировать следующим образом:

```
tru = \t f -> t
fls = \t f -> f
```

Определим, например, отрицание:

```
not = \b -> b fls tru
```

Несложно убедиться, что оно действительно работает. Определим конструкцию if:

```
if = \c t e -> c t e
```

Определите прочие логические связки:

```
or = \b1 b2 -> b1 tru b2 and = \b1 b2 -> b1 b2 fls
```

Убедитесь, что они действительно работают

Числа Чёрча

Нумералы Чёрча -- это способ закодировать натуральные числа в нашем исчислении.

```
0 = \s z -> z
1 = \s z -> s z
2 = \s z -> s (s z)
...
n = \s z -> s (s (s ... z))
```

Идея в том, что число кодируется количеством раз, которым мы применили некоторую функцию `s` к аргументу `z`. Ноль -- ноль раз и так далее. Воспользовавшись этой мыслью, можно на самом деле сделать многое. Для начала нам понадобится умение прибавлять единицу.

```
succ = \n s z -> s (n s z)
```

Научившись прибавлять единицу, мы можем складывать два числа Чёрча.

```
plus = \m n -> m succ n
```

Научившись складывать, напишите терм, перемножающий два числа Чёрча

```
mult = \m n -> m (plus n) 0
```

Заметьте, что я ничего не сказал про вычитание и деление. Их так же можно реализовать, но для этого понадобятся некоторые дополнительные знания. Мы увидим это на следующем занятии.

Домашнее задание

1. Выделите свободные и связанные переменные в термах и осуществите подстановки:

$x (\lambda x y \rightarrow y (x w) u) y [x := \lambda z \rightarrow z]$
 $(\lambda x \rightarrow x (\lambda y \rightarrow y x) w) (\lambda x \rightarrow v) [w := y (\lambda v \rightarrow v x)]$

2. Уберите лишние скобки и осуществите бета-преобразование(если это возможно):

$((\lambda x \rightarrow (\lambda y \rightarrow ((x y) z))) (a (b c)))$
 $((m n) m) (\lambda x \rightarrow ((x (u v)) y))$
 $((\lambda x \rightarrow (\lambda y \rightarrow ((y x) x))) (x (x y)) y)$

3. Покажите что

$B = S (K S) K$

4. Напишите терм, ведущий себя как логическая связка `xor`

$\text{xor} = \lambda b1 b2 \rightarrow ?$

5. Напишите терм, возводящий число Чёрча в степень:

$\text{pow} = \lambda b e \rightarrow ?$

6. (Опционально). Напишите терм, вычитающий единицу из числа Чёрча:

$\text{pred} = \lambda n \rightarrow ?$