

Функциональное программирование

Лекция 3. Просто типизированное лямбда-исчисление

Денис Николаевич Москвин

ВШЭ СПб, СПбШФМиКН,
бакалавриат ПМии

18.09.2020

- 1 Понятие типа
- 2 Просто типизированное λ -исчисление
- 3 Формализм систем λ_{\rightarrow}
- 4 Свойства λ_{\rightarrow}

- 1 Понятие типа
- 2 Просто типизированное λ -исчисление
- 3 Формализм систем λ_{\rightarrow}
- 4 Свойства λ_{\rightarrow}

Что такое типы?

Система типов — это гибко управляемый синтаксический метод доказательства отсутствия в программе определенных видов поведения при помощи классификации выражений языка по разновидностям вычисляемых ими значений.

Бенджамин Пирс

- В λ -исчислении:
 - выражения — λ -термы;
 - вычисление — их редукция;
 - значения — $(\text{WN})\text{NF}$.
- Типы — *синтаксические* конструкции, приписываемые термам по определённым правилам:

$M:\sigma$

Для чего нужны типы?

- Типы дают частичную спецификацию

$$f:\mathbb{N} \rightarrow \mathbb{N} \quad g:(\forall n:\mathbb{N}. \exists m:\mathbb{N}. m \leq n)$$

- Правильно типизированные программы не могут «сломаться». Робин Милнер (1978)

$$M:\sigma \wedge M \rightarrow v \Rightarrow v:\sigma$$

- Типизированные программы всегда завершаются.
(это не всегда так :)
- Проверка типов отлавливает простые ошибки.

- В большинстве систем типизации тождественной функции $I \equiv \lambda x. x$ может быть приписан тип $\alpha \rightarrow \alpha$

$$I : \alpha \rightarrow \alpha$$

- В общем случае $\alpha \rightarrow \beta$ является типом функции из α в β .
- Если имеется y типа α , являющийся аргументом функции I , то выражение $I y$ тоже имеет тип α .
- Гипотезы о типе переменных записывают в контексте

$$y : \alpha \vdash (I y) : \alpha$$

Примеры (на некотором условном языке)

`sin : Double → Double`

`length : Array → Int`

В λ -исчислении с типами выделяют два семейства систем типов.

Системы в стиле Карри

Термы те же, что и в бестиповой теории. Каждый терм обладает множеством различных типов (пустое, одно- или многоэлементное, бесконечное).

Системы в стиле Чёрча

Термы — аннотированные версии бестиповых термов. Каждый терм имеет тип (обычно уникальный), выводимый из способа, которым терм аннотирован.

Подход программиста

Термы интерпретируются как программы, а типы — как их частичные спецификации.

- Системы в стиле Карри: неявная типизация (например, Haskell, Ocaml).
- Системы в стиле Чёрча: явная типизация (большинство типизированных языков).

Логический подход

Типы интерпретируются как высказывания, а термы — как их доказательства.

Связь между «вычислительными» и логическими системами называют *соответствием Карри-Говарда*.

- 1 Понятие типа
- 2 Просто типизированное λ -исчисление
- 3 Формализм систем λ_{\rightarrow}
- 4 Свойства λ_{\rightarrow}

Просто типизированное λ -исчисление

Самая простая система — это *просто типизированное λ -исчисление* (λ_{\rightarrow} или Simple Type Theory (STT)).

Определение

Множество типов \mathbb{T} системы λ_{\rightarrow} определяется индуктивно:

$\alpha, \beta, \dots \in \mathbb{T}$ (переменные типа)

$\sigma, \tau \in \mathbb{T} \Rightarrow (\sigma \rightarrow \tau) \in \mathbb{T}$ (типы пространства функций)

- В абстрактном синтаксисе:

$$\mathbb{T} ::= \mathbb{V} \mid (\mathbb{T} \rightarrow \mathbb{T})$$

Здесь $\mathbb{V} = \{\alpha, \beta, \dots\}$ — множество типовых переменных.

- Соглашение: α, β, γ используем для типовых переменных, а σ, τ, ρ — для произвольных типов.

Соглашения и примеры

Стрелка *правоассоциативна*: если $\sigma_1, \dots, \sigma_n \in \mathbb{T}$, то

$$\sigma_1 \rightarrow \sigma_2 \rightarrow \dots \rightarrow \sigma_n \equiv (\sigma_1 \rightarrow (\sigma_2 \rightarrow \dots \rightarrow (\sigma_{n-1} \rightarrow \sigma_n) \dots))$$

$$(\alpha \rightarrow \beta) \equiv \alpha \rightarrow \beta$$

$$(\alpha \rightarrow (\beta \rightarrow \gamma)) \equiv \alpha \rightarrow \beta \rightarrow \gamma$$

$$((\alpha \rightarrow \beta) \rightarrow \gamma) \equiv (\alpha \rightarrow \beta) \rightarrow \gamma$$

$$\begin{aligned} ((\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \gamma))) &\equiv \\ (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma & \end{aligned}$$

$$\begin{aligned} ((\alpha \rightarrow \beta) \rightarrow (((\alpha \rightarrow \beta) \rightarrow \beta) \rightarrow \beta)) &\equiv \\ (\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow \beta) \rightarrow \beta) \rightarrow \beta & \end{aligned}$$

Всякий тип в λ_{\rightarrow} может быть записан в виде

$$\sigma_1 \rightarrow \sigma_2 \rightarrow \dots \rightarrow \sigma_n \rightarrow \alpha$$

Как приписать тип терму? (переменные и аппликация)

- Если терм *переменная* — как угодно:

$$\begin{aligned}x &: \alpha \\y &: \alpha \rightarrow \beta \\z &: (\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow \beta) \rightarrow \beta) \rightarrow \beta\end{aligned}$$

- Если терм *аппликация* $M N$, то
 - M должно быть функцией, то есть иметь стрелочный тип $M : \sigma \rightarrow \tau$;
 - N должно быть «подходящим» аргументом, то есть иметь тип $N : \sigma$;
 - вся аппликация при этом получит тип результата функции: $M N : \tau$.

$$\begin{array}{l}x : \alpha, y : \alpha \rightarrow \beta \quad \vdash \quad y x : \beta \\x : \alpha, y : \alpha \rightarrow \beta, z : \beta \rightarrow \gamma \quad \vdash \quad z (y x) : \gamma\end{array}$$

А какие должны иметь типы x и y , чтобы $x(y x) : \gamma$?

Как приписать тип терму? (абстракция)

- Если терм **абстракция** $\lambda x. M$, то
 - его тип должен быть стрелочным $\lambda x. M : \sigma \rightarrow \tau$;
 - тип аргумента x должен быть σ ;
 - тип тела абстракции M должен быть τ .
- Например, для $x : \alpha$ имеем $\lambda x. x : \alpha \rightarrow \alpha$
- Но писать $x : \alpha \vdash \lambda x. x : \alpha \rightarrow \alpha$ — плохая идея! Контекст глобален, а переменная x локальна, и ее имя может использоваться многократно в разных областях видимости.
- Можно ли как-то указать, что переменная x имеет тип α ?
 - Если не указать, то допустимо и $\lambda x. x : \beta \rightarrow \beta$ и даже $\lambda x. x : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$ — **стиль Карри**.
 - Если указать $\lambda x^\alpha. x : \alpha \rightarrow \alpha$, то тип терма определяется однозначно — **стиль Чёрча**.
- **Типизируйте по Чёрчу: $\lambda x^?. \lambda y^?. x (y x) : ?$**

Как приписать тип терму? (ассоциативность)

Правила ассоциативности для типов (вправо), аппликации (влево) и абстракции (вправо) хорошо согласованы друг с другом.

В предположении $M:\alpha, N:\beta, P:\gamma, Q:\gamma$

$$F : \alpha \rightarrow (\beta \rightarrow (\gamma \rightarrow \delta))$$

$$FM : \beta \rightarrow (\gamma \rightarrow \delta)$$

$$(FM)N : \gamma \rightarrow \delta$$

$$((FM)N)P : \delta$$

$$\lambda y^\beta. Q : \beta \rightarrow \gamma$$

$$\lambda x^\alpha. (\lambda y^\beta. Q) : \alpha \rightarrow (\beta \rightarrow \gamma)$$

Зелёные скобки необязательны и обычно опускаются.

- 1 Понятие типа
- 2 Просто типизированное λ -исчисление
- 3 Формализм систем λ_{\rightarrow}**
- 4 Свойства λ_{\rightarrow}

Определение

Множество *предтермов* (или *псевдотермов*) Λ строится из переменных из $V = \{x, y, z, \dots\}$ с помощью аппликации и абстракции:

$$\begin{aligned}x \in V &\Rightarrow x \in \Lambda \\M, N \in \Lambda &\Rightarrow (MN) \in \Lambda \\M \in \Lambda, x \in V &\Rightarrow (\lambda x. M) \in \Lambda\end{aligned}$$

- В абстрактном синтаксисе

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

- Предтермы системы в стиле Карри — это в точности термы бестипового λ -исчисления.

Определение

Множество *предтермов* $\Lambda_{\mathbb{T}}$ строится из переменных из $V = \{x, y, z, \dots\}$ с помощью аппликации и **аннотированной типами** абстракции:

$$\begin{aligned}x \in V &\Rightarrow x \in \Lambda_{\mathbb{T}} \\M, N \in \Lambda_{\mathbb{T}} &\Rightarrow (MN) \in \Lambda_{\mathbb{T}} \\M \in \Lambda_{\mathbb{T}}, x \in V, \sigma \in \mathbb{T} &\Rightarrow (\lambda x^{\sigma}. M) \in \Lambda_{\mathbb{T}}\end{aligned}$$

- В абстрактном синтаксисе

$$\Lambda_{\mathbb{T}} ::= V \mid (\Lambda_{\mathbb{T}} \Lambda_{\mathbb{T}}) \mid (\lambda V^{\mathbb{T}}. \Lambda_{\mathbb{T}})$$

- Все соглашения о скобках и ассоциативности те же, что и в системе Λ .

Система λ_{\rightarrow} а ля Карри:

$$\lambda x y. x$$
$$\lambda f g x. f (g x)$$
$$\lambda x. x x$$

Система λ_{\rightarrow} а ля Чёрч:

$\lambda x^{\alpha} y^{\beta}. x$	\equiv	$\lambda x:\alpha. \lambda y:\beta. x$
$\lambda x^{\alpha} y^{\alpha}. x$	\equiv	$\lambda x:\alpha. \lambda y:\alpha. x$
$\lambda f^{\alpha} g^{\beta} x^{\gamma}. f (g x)$	\equiv	$\lambda f:\alpha. \lambda g:\beta. \lambda x:\gamma. f (g x)$
$\lambda f^{\beta \rightarrow \gamma} g^{\alpha \rightarrow \beta} x^{\alpha}. f (g x)$	\equiv	$\lambda f:(\beta \rightarrow \gamma). \lambda g:(\alpha \rightarrow \beta). \lambda x:\alpha. f (g x)$
$\lambda x^{\alpha}. x x$	\equiv	$\lambda x:\alpha. x x$

Определение

Утверждение типизации в λ_{\rightarrow} «а ля Карри» имеет вид

$$M : \tau$$

где $M \in \Lambda$ и $\tau \in \mathbb{T}$. Тип τ иногда называют *предикатом*, а терм M — *субъектом* утверждения.

Для λ_{\rightarrow} «а ля Чёрч» надо лишь заменить Λ на $\Lambda_{\mathbb{T}}$.

Примеры утверждений типизации

Система в стиле Карри

$$\lambda x. x : \alpha \rightarrow \alpha$$

$$\lambda x. x : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$$

$$\lambda x y. x : \alpha \rightarrow \beta \rightarrow \alpha$$

Система в стиле Чёрча

$$\lambda x^{\alpha}. x : \alpha \rightarrow \alpha$$

$$\lambda x^{\alpha \rightarrow \beta}. x : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$$

$$\lambda x^{\alpha} y^{\beta}. x : \alpha \rightarrow \beta \rightarrow \alpha$$

Определение

Объявление — это утверждение типизации с термовой переменной в качестве субъекта.

Примеры объявлений

$$x : \alpha$$
$$y : \beta$$
$$f : \alpha \rightarrow \beta$$
$$g : (\alpha \rightarrow \beta) \rightarrow \gamma$$

Определение

Контекст — это множество объявлений с *различными* переменными в качестве субъекта:

$$\Gamma = \{x_1:\sigma_1, x_2:\sigma_2, \dots, x_n:\sigma_n\}$$

Контекст иногда называют *базисом* или *окружением*.

- Фигурные скобки множества иногда опускают:

$$\Gamma = x:\alpha, f:\alpha \rightarrow \beta, g:(\alpha \rightarrow \beta) \rightarrow \gamma \equiv x^\alpha, f^{\alpha \rightarrow \beta}, g^{(\alpha \rightarrow \beta) \rightarrow \gamma}$$

- Контексты можно *расширять*, добавляя объявление *новой* переменной:

$$\Delta = \Gamma, y^\beta = x^\alpha, f^{\alpha \rightarrow \beta}, g^{(\alpha \rightarrow \beta) \rightarrow \gamma}, y^\beta$$

- Контекст можно рассматривать как (частичную) функцию из множества переменных V в множество типов \mathbb{T} .

Утверждение $M : \tau$ называется **выводимым** в контексте Γ , обозначение

$$\Gamma \vdash M : \tau$$

если его вывод может быть произведен по правилам:

$$\begin{aligned} x^\sigma \in \Gamma &\Rightarrow \Gamma \vdash x : \sigma \\ \Gamma \vdash M : \sigma \rightarrow \tau, \quad \Gamma \vdash N : \sigma &\Rightarrow \Gamma \vdash MN : \tau \\ \Gamma, x^\sigma \vdash M : \tau &\Rightarrow \Gamma \vdash \lambda x. M : \sigma \rightarrow \tau \end{aligned}$$

Если существуют Γ и τ , такие что $\Gamma \vdash M : \tau$, то предтерм M называют **(допустимым) термом**.

(аксиома) $\Gamma \vdash x : \sigma$, если $x^\sigma \in \Gamma$

$(\rightarrow E)$
$$\frac{\Gamma \vdash M : \sigma \rightarrow \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

$(\rightarrow I)$
$$\frac{\Gamma, x^\sigma \vdash M : \tau}{\Gamma \vdash \lambda x. M : \sigma \rightarrow \tau}$$

$$\frac{x^\alpha, y^\beta \vdash x : \alpha}{x^\alpha \vdash \lambda y. x : \beta \rightarrow \alpha} (\rightarrow I)$$

$$\frac{x^\alpha \vdash \lambda y. x : \beta \rightarrow \alpha}{\vdash \lambda x y. x : \alpha \rightarrow \beta \rightarrow \alpha} (\rightarrow I)$$

Для любых $\sigma, \tau \in \mathbb{T}$ верно $\vdash \lambda x y. x : \sigma \rightarrow \tau \rightarrow \sigma$.

Пример дерева вывода для \mathbf{V} в λ_{\rightarrow} «а ля Карри»

Введем сокращение $\Gamma \equiv f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta}, x^{\alpha}$ для повторяющегося контекста.

$$\frac{\frac{\frac{\frac{\Gamma \vdash f : \beta \rightarrow \gamma}{f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta}, x^{\alpha} \vdash f(gx) : \gamma} (\rightarrow I)}{f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta} \vdash \lambda x. f(gx) : \alpha \rightarrow \gamma} (\rightarrow I)}{f^{\beta \rightarrow \gamma} \vdash \lambda g x. f(gx) : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma} (\rightarrow I)}{\vdash \lambda f g x. f(gx) : (\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma} (\rightarrow I)$$

$\Gamma \vdash g : \alpha \rightarrow \beta$ $\Gamma \vdash x : \alpha$ $(\rightarrow E)$

$\Gamma \vdash gx : \beta$ $(\rightarrow E)$

(аксиома) $\Gamma \vdash x : \sigma$, если $x^\sigma \in \Gamma$

$(\rightarrow E)$
$$\frac{\Gamma \vdash M : \sigma \rightarrow \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

$(\rightarrow I)$
$$\frac{\Gamma, x^\sigma \vdash M : \tau}{\Gamma \vdash \lambda x^\sigma. M : \sigma \rightarrow \tau}$$

$$\frac{x^\alpha, y^\beta \vdash x : \alpha}{x^\alpha \vdash \lambda y^\beta. x : \beta \rightarrow \alpha} (\rightarrow I)$$

$$\frac{x^\alpha \vdash \lambda y^\beta. x : \beta \rightarrow \alpha}{\vdash \lambda x^\alpha y^\beta. x : \alpha \rightarrow \beta \rightarrow \alpha} (\rightarrow I)$$

Для **каждой** пары $\sigma, \tau \in \mathbb{T}$ верно $\vdash \lambda x^\sigma y^\tau. x : \sigma \rightarrow \tau \rightarrow \sigma$.

Пример дерева вывода для \mathbf{B} в λ_{\rightarrow} «а ля Чёрч»

Введем сокращение $\Gamma \equiv f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta}, x^{\alpha}$ для повторяющегося контекста.

$$\frac{\frac{\frac{\Gamma \vdash f : \beta \rightarrow \gamma}{f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta}, x^{\alpha} \vdash f(gx) : \gamma} (\rightarrow I)}{f^{\beta \rightarrow \gamma}, g^{\alpha \rightarrow \beta} \vdash \lambda x^{\alpha}. f(gx) : \alpha \rightarrow \gamma} (\rightarrow I)}{f^{\beta \rightarrow \gamma} \vdash \lambda g^{\alpha \rightarrow \beta} x^{\alpha}. f(gx) : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma} (\rightarrow I)}{\vdash \lambda f^{\beta \rightarrow \gamma} g^{\alpha \rightarrow \beta} x^{\alpha}. f(gx) : (\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma} (\rightarrow I)$$

$\frac{\Gamma \vdash g : \alpha \rightarrow \beta \quad \Gamma \vdash x : \alpha}{\Gamma \vdash gx : \beta} (\rightarrow E)$

$\frac{\Gamma \vdash f : \beta \rightarrow \gamma \quad \Gamma \vdash gx : \beta}{\Gamma \vdash f(gx) : \gamma} (\rightarrow E)$

- 1 Понятие типа
- 2 Просто типизированное λ -исчисление
- 3 Формализм систем λ_{\rightarrow}
- 4 Свойства λ_{\rightarrow}

Лемма об инверсии (лемма генерации)

- $\Gamma \vdash x : \sigma \Rightarrow x^\sigma \in \Gamma$.
- $\Gamma \vdash MN : \tau \Rightarrow \exists \sigma [\Gamma \vdash M : \sigma \rightarrow \tau \wedge \Gamma \vdash N : \sigma]$.
- $\Gamma \vdash \lambda x. M : \rho \Rightarrow \exists \sigma, \tau [\Gamma, x^\sigma \vdash M : \tau \wedge \rho \equiv \sigma \rightarrow \tau]$.
($\lambda \rightarrow$ а ля Карри)
- $\Gamma \vdash \lambda x^\sigma. M : \rho \Rightarrow \exists \tau [\Gamma, x^\sigma \vdash M : \tau \wedge \rho \equiv \sigma \rightarrow \tau]$.
($\lambda \rightarrow$ а ля Чёрч)

Лемма о типизируемости подтерма

Пусть M' — подтерм M . Тогда $\Gamma \vdash M : \sigma \Rightarrow \Gamma' \vdash M' : \sigma'$ для некоторых Γ' и σ' .

То есть, если терм имеет тип, то каждый его подтерм тоже имеет тип.

Какой контекст требуется, чтобы произвести присваивание типов?

Лемма «разбавления» (Thinning)

Пусть Γ и Δ — контексты, причём $\Delta \supseteq \Gamma$. Тогда $\Gamma \vdash M : \sigma \Rightarrow \Delta \vdash M : \sigma$. *Расширение контекста не влияет на выводимость утверждения типизации.*

Лемма о свободных переменных

$\Gamma \vdash M : \sigma \Rightarrow FV(M) \subseteq \text{dom}(\Gamma)$. *Свободные переменные типизированного терма должны присутствовать в контексте.*

Лемма сужения

$\Gamma \vdash M : \sigma \Rightarrow \Gamma \upharpoonright FV(M) \vdash M : \sigma$. *Сужение контекста до множества свободных переменных терма не влияет на выводимость утверждения типизации.*

Свойства λ_{\rightarrow} : нетипизируемые предтермы

- Рассмотрим предтерм $x x$. Предположим, что это терм.
- Тогда имеются Γ и τ , такие что

$$\Gamma \vdash x x : \tau$$

- По лемме об инверсии существует такой σ , что правый подтерм $x : \sigma$, а левый подтерм (тоже x) имеет тип $\sigma \rightarrow \tau$.
- По лемме о контекстах $x \in \text{dom}(\Gamma)$ и должен иметь там *единственное* связывание по определению контекста. То есть $\sigma = \sigma \rightarrow \tau$ — тип является подвыражением себя, чего не может быть, поскольку (*и пока*) типы конечны.

Предтермы $\omega = \lambda x. x x$, $\Omega = \omega \omega$ и

$Y = \lambda f. (\lambda x. f(x x))(\lambda x. f(x x))$ не имеют типа по лемме о типизируемости подтерма.

$$x^\sigma \not\vdash x x : \tau, \quad \not\vdash \omega : \sigma, \quad \not\vdash \Omega : \sigma, \quad \not\vdash Y : \sigma.$$

Лемма подстановки типа для λ_{\rightarrow}

Определение

Для $\sigma, \tau \in \mathbb{T}$ *подстановку* τ вместо α в σ обозначим $[\alpha \mapsto \tau]\sigma$.

$$[\alpha \mapsto (\gamma \rightarrow \gamma)](\alpha \rightarrow \beta \rightarrow \alpha) = (\gamma \rightarrow \gamma) \rightarrow \beta \rightarrow \gamma \rightarrow \gamma$$

Лемма подстановки типа

$\Gamma \vdash M : \sigma \Rightarrow [\alpha \mapsto \tau]\Gamma \vdash M : [\alpha \mapsto \tau]\sigma$. (λ_{\rightarrow} Карри)

$\Gamma \vdash M : \sigma \Rightarrow [\alpha \mapsto \tau]\Gamma \vdash [\alpha \mapsto \tau]M : [\alpha \mapsto \tau]\sigma$. (λ_{\rightarrow} Чёрч)

Подстановка $[\alpha \mapsto (\gamma \rightarrow \gamma)]$:

$$\begin{array}{l} x^{\alpha} \vdash \lambda y^{\alpha} z^{\beta}. x : \alpha \rightarrow \beta \rightarrow \alpha \Rightarrow \\ x^{\gamma \rightarrow \gamma} \vdash (\lambda y^{\gamma \rightarrow \gamma} z^{\beta}. x) : (\gamma \rightarrow \gamma) \rightarrow \beta \rightarrow \gamma \rightarrow \gamma \end{array}$$

Лемма подстановки терма

Пусть $\Gamma, x^\sigma \vdash M : \tau$ и $\Gamma \vdash N : \sigma$, тогда $\Gamma \vdash [x \mapsto N]M : \tau$.

То есть, подходящая по типу *подстановка терма сохраняет тип*.

Пример

Берём утверждение о типизации

$$x^{\gamma \rightarrow \gamma} \vdash \lambda y^\beta. x : \beta \rightarrow \gamma \rightarrow \gamma$$

и подставляем в него вместо свободной переменной x типа $\gamma \rightarrow \gamma$ терм $\lambda z^\gamma. z$ подходящего типа $\gamma \rightarrow \gamma$. Получаем

$$\vdash \lambda y^\beta z^\gamma. z : \beta \rightarrow \gamma \rightarrow \gamma$$

Что произойдет с деревом вывода типа при такой подстановке?

Лемма подстановки терма позволяет доказать теорему о сохранении типа в процессе вычислений.

Теорема о редукции субъекта

Пусть $M \rightarrow_{\beta} N$. Тогда $\Gamma \vdash M : \sigma \Rightarrow \Gamma \vdash N : \sigma$.

- То есть *тип терма сохраняется при β -редукциях*.
- С вычислительной точки зрения это одно из ключевых свойств любой системы типов.

Следствие

Множество типизируемых в λ_{\rightarrow} термов замкнуто относительно редукции.

В обратную сторону эта теорема (и следствие из нее) не верны для λ_{\rightarrow} .

Единственность типа в λ_{\rightarrow}

Теорема о единственности типа для λ_{\rightarrow} а ля Чёрч

Пусть $\Gamma \vdash M : \sigma$ и $\Gamma \vdash M : \tau$. Тогда $\sigma \equiv \tau$.

Терм в λ_{\rightarrow} а ля Чёрч имеет единственный тип.

Следствие

Пусть $\Gamma \vdash M : \sigma$, $\Gamma \vdash N : \tau$ и $M =_{\beta} N$. Тогда $\sigma \equiv \tau$.

Типизируемые β -конвертируемые термы имеют одинаковый тип в λ_{\rightarrow} а ля Чёрч.

Контрпример для системы а ля Карри

Оба типа подходят для $K = \lambda x y. x$ в λ_{\rightarrow} а ля Карри:

$$\vdash \lambda x y. x : \alpha \rightarrow (\delta \rightarrow \gamma \rightarrow \delta) \rightarrow \alpha$$

$$\vdash \lambda x y. x : (\gamma \rightarrow \gamma) \rightarrow \beta \rightarrow \gamma \rightarrow \gamma$$

Для систем в стиле Карри единственности типа нет.

Связь между системами Карри и Чёрча

- Можно задать стирающее отображение $|\cdot| : \Lambda_{\mathbb{T}} \rightarrow \Lambda$:

$$\begin{aligned} |x| &\equiv x \\ |MN| &\equiv |M| |N| \\ |\lambda x^{\sigma}. M| &\equiv \lambda x. |M| \end{aligned}$$

- Все атрибутированные типами термы из версии Чёрча $\lambda \rightarrow$ «проектируются» в термы в версии Карри:

$$M \in \Lambda_{\mathbb{T}} \wedge \Gamma \vdash_{\mathbb{C}} M : \sigma \Rightarrow \Gamma \vdash_{\mathbb{K}} |M| : \sigma$$

- Термы из версии Карри $\lambda \rightarrow$ могут быть «подняты» в термы из версии Чёрча:

$$M \in \Lambda \wedge \Gamma \vdash_{\mathbb{K}} M : \sigma \Rightarrow \exists N \in \Lambda_{\mathbb{T}} [\Gamma \vdash_{\mathbb{C}} N : \sigma \wedge |N| \equiv M]$$

- Для произвольного типа $\sigma \in \mathbb{T}$ выполняется

$$\sigma \text{ обитаем в } \lambda \rightarrow \text{-Карри} \Leftrightarrow \sigma \text{ обитаем в } \lambda \rightarrow \text{-Чёрч}$$

- Есть ли алгоритм, который позволяют решить задачу?

$\vdash M : \sigma?$	Задача проверки типа Type Checking Problem	ЗПТ TCP
$\vdash M : ?$	Задача синтеза типа Type Synthesis (or Assgnment) Problem	ЗСТ TSP, TAP
$\vdash ? : \sigma$	Задача обитаемости типа Type Inhabitation Problem	ЗОТ TIP
- Для λ_{\rightarrow} (и в стиле Чёрча, и в стиле Карри) все эти задачи разрешимы.
- ЗПТ выглядит проще ЗСТ, но в системах Карри они эквивалентны: проверка $M N : \sigma?$ требует синтеза $N : ?$.

Определение

Терм называют *слабо (weak) нормализуемым (WN)*, если **существует** последовательность редукций, приводящих его к нормальной форме.

Определение

Терм называют *сильно (strong) нормализуемым (SN)*, если **любая** последовательность редукций, приводит его к нормальной форме.

Примеры

Терм KIK — сильно нормализуем,
терм $KI\Omega$ — слабо нормализуем,
терм Ω — не нормализуем.

Теорема о нормализации λ_{\rightarrow}

Определение

Систему типов называют *слабо нормализуемой* если все её допустимые термы слабо нормализуемы.

Определение

Систему типов называют *сильно нормализуемой* если все её допустимые термы сильно нормализуемы.

Теорема о нормализации λ_{\rightarrow}

Обе системы λ_{\rightarrow} (и Карри, и Чёрча) *сильно нормализуемы*.

То есть любой допустимый терм в λ_{\rightarrow} всегда редуцируется к нормальной форме.