Teorie λ

LKFP přednáška 1, jaro 2024/25

Michal Krupka

23. února 2025

1 Výrazy λ-kalkulu (λ-termy)

Abeceda.

- $x, y, z, \ldots$ — proměnné
- λ — abstraktor
- $(,)$ — závorky

Výrazy. Množina Λ všech výrazů obsahuje následující výrazy (upřesníme za chvíli)

1. proměnné,
2. $(\lambda x M)$, kde M je výraz (*abstrakce*),
3. MN, kde M a N jsou výrazy (*aplikace*).

Při zápisu výrazů se řídíme konvencemi (zde a dále znak $\equiv$ znamená *syntaktickou rovnost*): Vynechávání závorek:

- $MNL \equiv (MN)L$ (význam: parciální aplikace)
- $\lambda x.M \equiv (\lambda x M)$
- obecněji: $\lambda x_1 \ldots x_n.M \equiv \lambda \vec{x}.M \equiv (\lambda x_1(\ldots(\lambda x_n M)\ldots))$ (význam: currying)
- $\lambda x.M_1 \ldots M_n \equiv (\lambda x.M_1 \ldots M_n)$ (dokud to jde, vše je součástí abstrakce)

V dalším nerozlišujeme mezi proměnnou (např. x) a symbolem pro libovolnou proměnnou (např. x jako zástupce pro $x, y, z, \ldots$). Čtenář vždy pochopí, o co jde.

Podvýrazy. Množina $\mathrm{Sub}(M)$ všech podvýrazů výrazu M je definována takto:

- $\mathrm{Sub}(x) = \{x\}$
- $\mathrm{Sub}(\lambda x.M) = \{\lambda x.M\} \cup \mathrm{Sub}(M)$
- $\mathrm{Sub}(MN) = \{MN\} \cup \mathrm{Sub}(M) \cup \mathrm{Sub}(N)$

2 Volné a vázané proměnné

(Free, bound variables)

Pro výraz M definujeme množinu $\mathrm{FV}(M)$ jeho *volných proměnných* takto:

- $\mathrm{FV}(x) = \{x\}$
- $\mathrm{FV}(\lambda x.M) = \mathrm{FV}(M) \setminus \{x\}$
- $\mathrm{FV}(MN) = \mathrm{FV}(M) \cup \mathrm{FV}(N)$

Množina $\mathrm{BV}(M)$ *vázaných proměnných* výrazu M je definována takto:

- $\mathrm{BV}(x) = \emptyset$
- $\mathrm{BV}(\lambda x.M) = \mathrm{BV}(M) \cup \{x\}$
- $\mathrm{BV}(MN) = \mathrm{BV}(M) \cup \mathrm{BV}(N)$

Proměnná ve výrazu může být současně volná i vázaná. Např. pro

$$M \equiv x(\lambda xy.x)$$

Je $\mathrm{FV}(M) = \{x\}$ a $\mathrm{BV}(M) = \{x, y\}$ (cvičení).

V množině Λ ale uvažujeme pouze výrazy, pro které jsou $\mathrm{BV}(M)$ a $\mathrm{FV}(M)$ disjunktní. Přesně ji tedy definujeme jako nejmenší množinu Λ, pro kterou

1. všechny proměnné leží v Λ,
2. jestliže $M \in \Lambda$, pak $(\lambda xM) \in \Lambda$,
3. jestliže $M, N \in \Lambda$, $\mathrm{FV}(M) \cap \mathrm{BV}(M) = \emptyset$ a $\mathrm{BV}(M) \cap \mathrm{FV}(M) = \emptyset$, pak $MN \in \Lambda$.

Když $\mathrm{FV}(M) = \emptyset$, pak M je *uzavřený* (*kombinátor*). Množinu všech kombinátorů v Λ značíme Λ^0. Symbolem $\Lambda^0(\vec{x})$ značíme množinu všech M, jejichž volné proměnné jsou mezi proměnnými z $\{\vec{x}\}$ ($\mathrm{FV}(M) \subseteq \{\vec{x}\}$). Uzávěrem výrazu M rozumíme výraz $\lambda\vec{x}.M$, kde $\{\vec{x}\} = \mathrm{FV}(M)$. (Jestliže $\vec{x}$ je $x_1, \ldots, x_n$, pak $\{\vec{x}\}$ značí množinu $\{x_1, \ldots, x_n\}$.)

3 Teorie λ

Z λ termů vytvoříme formule a pro ně axiomy a odvozovací pravidla.

Formule jsou tzv. *rovnice*

$$M = N$$

pro $M, N \in \Lambda$.

Axiomy a odvozovací pravidla. Označme $M[x := N]$ výraz vzniklý z M tak, že se v M všechny *volné výskyty proměnné* x nahradí výrazem N (přesná definice: cvičení). Následující axiom je znám jako β-*konverze*.

$$(\lambda x.M)N = M[x := N] \tag{β}$$

Proměnná x nesmí být v M současně volná i vázaná, kdyby byla, byli bychom schopni v teorii λ „dokázat“ libovolnou formuli. Zkuste si jako cvičení. My jsme se problému vyhnuli dodatečným předpokladem.

Church zaváděl α-*konverzi*:

$$\lambda x.M = \lambda y.M[x := y] \tag{α}$$

V současnosti se α-konverze nepoužívá, místo toho se uvedené výrazy považují za syntakticky totožné ($\lambda x.M \equiv \lambda y.M[x := y]$).

Ještě jeden axion:

$$M = M$$

a odvozovací pravidla:

$$\frac{M = N}{N = M}, \qquad \frac{M = N,\ N = L}{M = L},$$

$$\frac{M = N}{MZ = NZ}, \qquad \frac{M = N}{ZM = ZN},$$

$$\frac{M = N}{\lambda x.M = \lambda x.N} \qquad (\xi)$$

Pravidlu ξ se říká *slabá extenzionalita*.

Množinám formulí (rovnic) říkáme *teorie*. *Teorie* λ je nejmenší množina formulí, která obsahuje všechny formule dokazatelné z uvedených axiomů pomocí uvedených odvozovacích pravidel. Pokud řekneme něco jako „platí $M = N$ (v teorii λ)“, pak tím myslíme, že rovnice $M = N$ je prvkem teorie λ neboli *teorém*.

Příklad 3.1 (věta o pevném bodě)**.** Pro každé F existuje X tak, že $FX = X$: stačí položit $W \equiv \lambda x.F(xx)$ a $X \equiv WW$. Uvedenou rovnici pak lze snadno dokázat.

4 Kontexty a referenční transparentnost

Kontexty jsou pomocné výrazy, „λ-termy s dírami“, do kterých jde dosazovat jiné výrazy. Od λ-termů se liší tím, že na místě libovolného podvýrazu mohou mít „díru“, která je značena $[\,]$.

Množina $\mathcal{C}[\,]$ *λ-kontextů* je nejmenší množina taková že

1. $x \in \mathcal{C}[\,]$ pro každou proměnnou x,
2. $[\,] \in \mathcal{C}[\,]$,
3. jestliže $C_1[\,], C_2[\,] \in \mathcal{C}[\,]$, pak $(\lambda x.C_1[\,]) \in \mathcal{C}[\,]$ a $(C_1[\,]C_2[\,]) \in \mathcal{C}[\,]$.

Kontexty značíme vždy ukončené hranatými závorkami jako např. $C_1[\,], C_2[\,]$ výše. „Díra“ v kontextu je reprezentována „$[\,]$“. Značí místo, kam lze do kontextu dosadit výraz. U kontextů vynecháváme závorky podobně jako u λ-termů.

Například pro výraz M je

$$C[\,] \equiv (\lambda x.[\,]x)M$$

kontext.

Výraz vzniklý *dosazením* výrazu do kontextu lze snadno definovat (cvičení), pro kontext $C[\,]$ a výraz N se značí $C[N]$.

Například pro výše uvedený kontext $C[\,]$ je

$$C[\lambda y.y] \equiv (\lambda x.(\lambda y.y)x)M.$$

Pro kontexty *neplatí* lexikální totožnost analogická α-konverzi (neuvažujeme je „modulo α-kongruence“) (proč?).

Referenční transparentnost. *Jestliže* $C[\,]$ *je kontext a* $M = N$*, pak* $C[M] = C[N]$*.*

Důkaz. Indukcí přes strukturu kontextu (cvičení).

Referenční transparentnost je základní vlastnost funkcionálních programovacích jazyků. (Je-li N chápáno jako hodnota výrazu M, pak náhradou N za M v libovolném kontextu dostaneme výraz rovný původnímu výrazu.)

5 Extenzionalita

Rovnost ve formulích lze chápat jako intenzionální rovnost: dva výrazy se rovnají, pokud se jejich rovnost dá odvodit z jejich tvaru.

Extenzionální rovnost je něco zcela jiného. Např. dva algoritmy jsou extenzionálně rovny, pokud pro daný vstup mají stejný výstup (např. dva různé řadicí algoritmy).

Extenzionální rovnost by vystihovalo následující pravidlo:

$$\frac{Mx = Nx}{M = N} \text{ pro } x \notin \mathrm{FV}(MN). \tag{ext}$$

Jeho přidáním k pravidlům teorie λ dostáváme *teorii* $\lambda + ext$.

Druhou možností je přidat axiom (Church)

$$\lambda x.Mx = M \text{ pro } x \notin \mathrm{FV}(M). \tag{η}$$

Dostaneme *teorii* $\lambda\eta$.

Teorie $\lambda + ext$ *a* $\lambda\eta$ *jsou totožné.*

Důkaz. Nejprve ukážeme, že $\lambda x.Mx = M$ (pro $x \notin \mathrm{FV}(M)$) je teorém teorie $\lambda + \text{ext}$. Podle (β) máme

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

a z (ext) dostáváme $\lambda x.Mx = M$.

Naopak, předpokládejme, že pro $x \notin \mathrm{FV}(MN)$ je

$$Mx = Nx$$

teorém. Pak podle slabé extenzionality (ξ) je teorém i

$$\lambda x.Mx = \lambda x.Nx$$

a použitím (η) vlevo i vpravo (obnáší to použít víc pravidel) dostáváme teorém $M = N$.

Extenzionalita se v praxi příliš nepoužívá, my se k ní ještě vrátíme.