

Tüübisüsteemid

- Tüübid spetsifitseerivad programmide omadusi.
 - λ -arvutuses klassifitseerivad normaalkujusid.
 - Kui term on tüüpi `Nat`, ja ta on normaalkujul, siis see term esitab naturaalarvu.
- Tüübisüsteem koosneb kolmest komponendist:
 - tüüpide hulk;
 - programmide hulk;
 - tüüpimisreeglid.

Lihtsad tüübid

- Olgu antud loenduv hulk tüübimuutujaid (baastüübid).
- Lihtsad tüübid = ei ole polümorfseid.
- Lihtsate tüüpide süntaks:

$$\begin{array}{ll}
 \tau & ::= \alpha & \text{tüübimuutuja} \\
 & | \tau_1 \rightarrow \tau_2 & \text{funktsioonitüüp}
 \end{array}$$

- Tüübikonstruktor \rightarrow on paremassotsiatiivne.

$$\tau_1 \rightarrow \tau_2 \rightarrow \tau_3 \equiv \tau_1 \rightarrow (\tau_2 \rightarrow \tau_3)$$

Church vs. Curry

- Tüüpidega λ -avaldiste esitamiseks on kaks peamist stiili.
- **Church'i stiil:** funktsiooni parameetrite tüübid esitatakse ilmutatult λ -termis $\lambda x : \tau. e$.
 - Näiteks: $\lambda x : \text{Nat}. x + x : \text{Nat} \rightarrow \text{Nat}$.
 - Igal termil on unikaalne tüüp.
- **Curry stiil:** termid on tavalised (s.o. ilma tüüpideta) ja kasutatakse tüübikorrektuse predikaati (tüübituletus).
 - Näide: $\lambda x. x + x : \text{Nat} \rightarrow \text{Nat}$.
 - Termidel on palju tüüpe (näit. $\lambda x. x : \text{Nat} \rightarrow \text{Nat}$, kuid ka $\lambda x. x : \text{Bool} \rightarrow \text{Bool}$).
- Meie kasutame põhiliselt esimest stiili.

Termid ja redutseerimine

- Termide süntaks:

e	::=	x	muutuja
		$e_1 e_2$	aplikatsioon
		$\lambda x : \tau. e$	abstraktsioon

- Samad süntaktilised konventsioonid kui puhtas λ -arvutuses.
- Substitutsioon, reduktsioon jmt on defineeritud analoogselt puhta λ -avutusega ignoreerides tüübiannotatsioone.

Tüüpimisrelatsioon

- Et teha kindlaks mis termid on millist tüüpi, defineerime tüüpimisrelatsiooni.
- Baasrelatsioon on kujul $\Gamma \vdash e : \tau$
 - "term e on tüüpi τ kontekstis Γ ".
- Kontekst Γ on muutujast ja tüübist koosnevate paaride jada $\Gamma = \{x_1 : \tau_1, \dots, x_n : \tau_n\}$.
 - Konteksti Γ doomenit tähistame $\text{dom}(\Gamma) = \{x_1, \dots, x_n\}$.
 - Korrektses tüüpimisrelatsioonis peab olema $\text{FV}(e) \subseteq \text{dom}(\Gamma)$.
- $\Gamma \leq \Delta$ tähendab, et kui $x \in \text{dom}(\Gamma)$, siis $x \in \text{dom}(\Delta)$ ja $\Gamma(x) = \Delta(x)$.
- Programm (s.o. kinnine term) e on tüüpi τ , kui tal on see tüüp tühjas kontekstis $\vdash e : \tau$.

Tüüpimisrelatsioon

- Tüüpimisrelatsioon on defineeritud tüüpimisreeglite abil.
- Lihtsalt tüübitud λ -arvutuse tüüpimisreeglid:

$$\frac{}{\Gamma, x : \tau \vdash x : \tau} \textit{var} \qquad \frac{\Gamma, x : \sigma \vdash e : \tau}{\Gamma \vdash \lambda x : \sigma. e : \sigma \rightarrow \tau} \textit{abs}$$

$$\frac{\Gamma \vdash e_1 : \sigma \rightarrow \tau \quad \Gamma \vdash e_2 : \sigma}{\Gamma \vdash e_1 e_2 : \tau} \textit{app}$$

kus $x \notin \text{dom}(\Gamma)$.

- Edaspidi tähistame lihtsalt tüübitud λ -arvutust $\lambda \rightarrow$.

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (1):

$$\vdash \lambda x:\tau, y:\sigma. x :$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (1):

$$\frac{\frac{}{\{x:\tau\} \vdash \lambda y:\sigma. x : \tau}}{\vdash \lambda x:\tau, y:\sigma. x : \tau \rightarrow \tau} \text{ abs}}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (1):

$$\frac{\frac{\overline{\{x:\tau, y:\sigma\} \vdash x : \sigma}}{\{x:\tau\} \vdash \lambda y:\sigma. x : \sigma \rightarrow \sigma} \text{ abs}}{\vdash \lambda x:\tau, y:\sigma. x : \tau \rightarrow \sigma \rightarrow \sigma} \text{ abs}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (1):

$$\frac{\frac{\frac{}{\{x:\tau, y:\sigma\} \vdash x:\tau} \text{var}}{\{x:\tau\} \vdash \lambda y:\sigma. x:\sigma \rightarrow \tau} \text{abs}}{\vdash \lambda x:\tau, y:\sigma. x:\tau \rightarrow \sigma \rightarrow \tau} \text{abs}}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (2):

$$\frac{\frac{\frac{\Gamma \vdash x : \sigma \rightarrow \tau \quad \frac{\Gamma \vdash y : \rho \rightarrow \sigma \quad \Gamma \vdash z : \rho}{\Gamma \vdash yz : \sigma}}{\Gamma \vdash x(yz) : \tau}}{\Gamma_2 \vdash \lambda z : \rho. x(yz) : \rho \rightarrow \tau}}{\Gamma_1 \vdash \lambda y : \rho \rightarrow \sigma, z : \rho. x(yz) : (\rho \rightarrow \sigma) \rightarrow \rho \rightarrow \tau}}{\vdash \lambda x : \sigma \rightarrow \tau, y : \rho \rightarrow \sigma, z : \rho. x(yz) : (\sigma \rightarrow \tau) \rightarrow (\rho \rightarrow \sigma) \rightarrow \rho \rightarrow \tau}$$

kus

$$\begin{aligned}
 \Gamma_1 &= \{x : \sigma \rightarrow \tau\} \\
 \Gamma_2 &= \{x : \sigma \rightarrow \tau, y : \rho \rightarrow \sigma\} \\
 \Gamma &= \{x : \sigma \rightarrow \tau, y : \rho \rightarrow \sigma, z : \rho\}
 \end{aligned}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (3):

$$\frac{}{\vdash \lambda x:\tau. x x :}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (3):

$$\frac{\{x:\tau\} \vdash x x : \tau}{\vdash \lambda x:\tau. x x : \tau \rightarrow \tau} \text{ abs}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (3):

$$\frac{\frac{\overline{\{x:\tau\} \vdash x : \sigma \rightarrow} \quad \overline{\{x:\tau\} \vdash x : \sigma}}{\overline{\{x:\tau\} \vdash x x :}} \quad app}{\overline{\vdash \lambda x:\tau. x x : \tau \rightarrow}} \quad abs$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (3):

$$\frac{\frac{\overline{\{x:\tau\} \vdash x : \tau \rightarrow} \quad \overline{\{x:\tau\} \vdash x : \tau}}{\{x:\tau\} \vdash x x :} \quad \begin{array}{l} \textit{var} \\ \textit{app} \end{array}}{\vdash \lambda x:\tau. x x : \tau \rightarrow} \quad \textit{abs}$$

Tüüpimispuu konstrueerimine

Tüüpimispuu konstrueerimise näide (3):

$$\frac{\frac{\frac{}{\{x:\tau\} \vdash x : \tau \rightarrow \rho} \text{var}}{\{x:\tau\} \vdash x : \tau} \text{app}}{\{x:\tau\} \vdash x x : \rho} \text{abs}}{\vdash \lambda x:\tau. x x : \tau \rightarrow \rho} \text{abs}$$

Tüüpimispuu konstrueerimine ebaõnnestub!

Tüübiohutus

- **Tüübiohutus:**
Korrektset tüübitud programmid "ei lähe valesti".
- Hind:
 - Mõned mõistlikud programmid hüljatakse.
- Koosneb kahest osast:
 - **Progress:** Korrektset tüübitud ei ole tupikus (ta on kas väärtus või saab teha ühe väärtustusreeglitele vastava sammu).
 - **Säilitamine:** Kui korrektset tüübitud term teeb ühe sammu, siis ka resultaat on korrektset tüübitud.

Tüüpide unikaalsus, säilimine ja progress

- **Tüüpide unikaalsus:**
Kui $\Gamma \vdash e : \tau_1$ ja $\Gamma \vdash e : \tau_2$, siis $\tau_1 = \tau_2$.
- NB! Tüüpide unikaalsus ei kehti mitmete rikkamate keelte korral.
- **Tüüpide säilimine (subjekt-reduktsioon):**
Kui $\Gamma \vdash e : \tau$ ja $e \rightarrow_{\beta} e'$, siis $\Gamma \vdash e' : \tau$.
- **Progress:**
Kui $\Gamma \vdash e : \sigma$, siis $\exists e' : \sigma. e \rightarrow_{\beta} e'$ või $e \in \text{Val}$, kus

$$v \in \text{Val} ::= x v \dots v \mid \lambda x : \tau. v$$

Tugev normaliseerimine

- **Tugev normaliseerimine:**

Lihtsalt tüübitud λ -arvutuses $\lambda \rightarrow$ on iga β -reduktsiooni jada lõplik.

- Mõned lihtsad järeldused:

- Termide võrdsuse küsimus on lahenduv.
- Püsipunktikombinaatorid ei ole defineeritavad.

Laiendus: tõeväärtused

- Tüübid: $\tau ::= \dots \mid \text{Bool}$
- Termid: $e ::= \dots \mid \text{true} \mid \text{false} \mid \text{if } e_0 \text{ then } e_1 \text{ else } e_2$
- Väärtused: $v ::= \dots \mid \text{true} \mid \text{false}$
- Tüüpimisreeglid:

$$\frac{}{\Gamma \vdash \text{true} : \text{Bool}} \quad \frac{}{\Gamma \vdash \text{false} : \text{Bool}}$$

$$\frac{\Gamma \vdash e_0 : \text{Bool} \quad \Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash \text{if } e_0 \text{ then } e_1 \text{ else } e_2 : \tau}$$

- Väärtustamisreeglid:

$$\begin{aligned} \text{if true then } e_1 \text{ else } e_2 &\rightarrow e_1 \\ \text{if false then } e_1 \text{ else } e_2 &\rightarrow e_2 \end{aligned}$$

Laiendus: ühiktüüp

- Tüübid: $\tau ::= \dots \mid \text{Unit}$
- Termid: $e ::= \dots \mid ()$
- Väärtused: $v ::= \dots \mid ()$
- Tüüpimisreeglid:

$$\frac{}{\Gamma \vdash () : \text{Unit}}$$

- Väärtustamisreegleid ei ole!

Laiendus: paarid

- Tüübid: $\tau ::= \dots \mid \tau_1 \times \tau_2$
- Termid: $e ::= \dots \mid (e_1, e_2) \mid \text{fst} \mid \text{snd}$
- Väärtused: $v ::= \dots \mid (v_1, v_2)$
- Tüüpimisreeglid:

$$\frac{\Gamma \vdash e_1 : \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash (e_1, e_2) : \tau_1 \times \tau_2}$$

$$\frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \text{fst } e : \tau_1} \quad \frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \text{snd } e : \tau_2}$$

- Väärtustamisreeglid:

$$\begin{aligned} \text{fst}(e_1, e_2) &\rightarrow e_1 \\ \text{snd}(e_1, e_2) &\rightarrow e_2 \end{aligned}$$

Laiendus: summatüüp

- Tüübid: $\tau ::= \dots \mid \tau_1 + \tau_2$
- Termid: $e ::= \dots \mid \text{inl} \mid \text{inr} \mid \text{case}(e_0; x_1.e_1; x_2.e_2)$
- Väärtused: $v ::= \dots \mid \text{inl } v_1 \mid \text{inr } v_2$
- Tüüpimisreeglid:

$$\frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \text{inl } e : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \text{inr } e : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash e_0 : \tau_1 + \tau_2 \quad \Gamma, x_1 : \tau_1 \vdash e_1 : \sigma \quad \Gamma, x_2 : \tau_2 \vdash e_2 : \sigma}{\Gamma \vdash \text{case}(e_0; x_1.e_1; x_2.e_2) : \sigma}$$

- Evaluation rules:

$$\text{case}(\text{inl } e_0; x_1.e_1; x_2.e_2) \rightarrow e_1[x_1 \mapsto e_0]$$

$$\text{case}(\text{inr } e_0; x_1.e_1; x_2.e_2) \rightarrow e_2[x_2 \mapsto e_0]$$