Booleans

t ::= true | false | if t_1 then t_2 else t_3
v ::= true | false

Berechnungs-Regeln:

$$\frac{}{\text{if true then } t_2 \text{ else } t_3 \rightarrow t_2}\text{E-IF True}$$
$$\frac{}{\text{if true then } t_2 \text{ else } t_3 \rightarrow t_3}\text{E-IF False}$$

Kongruenz-Regel:

$$\frac{t_1 \rightarrow t_1'}{\text{if true then } t_2 \text{ else } t_3 \rightarrow \text{ if } t_1' \text{ then } t_2 \text{ else } t_3}\text{E-If}$$

Typisierung:

$$\frac{}{\text{true:Bool}}\text{T-True} \qquad \frac{}{\text{false:Bool}}\text{T-False}$$
$$\frac{t_1\text{:Bool} \quad t_2{:}T \quad t_3{:}T}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3{:}T}\text{T-True}$$

Einfach-getypter λ-Kalkül

Beispiel. λxt : Fun
$(\lambda xx)true$: Bool
$(\lambda x\lambda yy)true$: Fun
(Hier sind (λxx) und $(\lambda x\lambda yy)$ vom Typ Fun)

$\lambda xx : Bool \rightarrow Bool$ $\qquad$ $\lambda xx : Int \rightarrow Int$
$\lambda x\lambda yy : Bool \rightarrow (Bool \rightarrow Bool)$ $\lambda x\lambda yy : Bool \rightarrow Int \rightarrow Int)$

$T_1 \rightarrow (T_2 \rightarrow T_3)$: Typ von Funktionen, die ein Argument vom Typ T_1 erwarten und eine Funktion vom Typ $T_2 \rightarrow T_3$ zurückliefern.
$(T_1 \rightarrow T_2) \rightarrow T_3$: Typ der Funktionen, die eine Funktion vom Typ $T_1 \rightarrow T_2$ als Argument erwarten und etwas vom Typ T_3 zurückliefern.
$\lambda xx : A \rightarrow A$ SML: $(fnx \Rightarrow x) :' a \rightarrow' a$
$\lambda x : Bool.x : Bool \rightarrow Bool$
$\underbrace{\lambda x : Int.x}_{t} : \underbrace{Int \rightarrow Int}_{T}$

$f : R \rightarrow R$
$x \mapsto sin(2x)$
$f = \lambda x.sin(2x) : R \rightarrow R$

0.1 Definition. Syntax des reinen einfach getypten Lambda-Kalküls ($\lambda^{rightarrow}$)

Terme: t :: = x
| λ x : T t
| t_1 t_2

Typen: T ::= T → T

Kontexte: T :: = .		leerer Kontext
	| Γ , x: T	erweiterter Kontext

$\lambda x : Bool$	$\lambda y : Int\ x :$	$Bool \rightarrow Int \rightarrow Bool$
	$\lambda y : Int\ x :$	$Int \rightarrow Bool$
	$x :$	$Bool$

$$\cfrac{\cfrac{\cfrac{}{x{:}Bool, y{:}Int \vdash x{:}Bool}T-VAR}{x{:}Bool \vdash \lambda y{:}Int\ x{:}Int \rightarrow Bool}T-LAM}{\vdash \lambda x : Bool\ \lambda y : Int\ x : Bool \rightarrow Int \rightarrow Bool}T-LAM$$

$$\frac{\Gamma \vdash t_1 : T' \rightarrow T \quad \Gamma \vdash t_2 : T'}{\Gamma \vdash t_1\ t_2 : T}$$

0.2 Definition. Typisierungs-Relation $\Gamma \vdash t : T$
"Im Kontext Γ hat t den Typ T".

$$\frac{x : T \in \Gamma}{\Gamma \vdash x : T} T - Var$$

$$\frac{\Gamma, x : S \vdash t : T}{\Gamma \vdash \lambda x : S.t : S \rightarrow T} T - LAM$$

$$\frac{\Gamma \vdash t : S \rightarrow T \quad \Gamma \vdash s : S}{\Gamma \vdash ts : T} T - APP$$

0.3 Definition. ($\lambda^{\rightarrow, Bool}$)
Der Kalkül $\lambda^{\rightarrow, Bool}$ entsteht durch Hinzunehmen von Booleans zu $\lambda^{\rightarrow}$

Beispiel.

In $\lambda^{\rightarrow, Bool}$ gibt es folgende Typen:	Rang	
Bool	0	
$Bool \rightarrow Bool, Bool \rightarrow Bool \rightarrow Bool$	1	"first-order"
$(Bool \rightarrow Bool) \rightarrow Bool, ...$	2	↓
$((Bool \rightarrow Bool) \rightarrow Bool) \rightarrow Bool$	3	"higher-order"

0.4 Definition. (Rang, Ordnung)
$rk(Bool) = 0$
$rk(T_1 \rightarrow T_2) = max(rk(T_1) + 1, rk(T_2))$

Nun interessiert uns auch wieder die Frage der Typsicherheit: Dafüer benötigen wir auch wieder:

0.5 Definition. Auswertung cbv für geschlossene Terme
<u>Werte</u> $v ::= \lambda x : S.t$

$$\frac{}{(\lambda x : S.t)v \rightarrow [v/x]t} E - BetaV$$

$$\frac{t_1 \rightarrow t_1'}{t_1 t_2 \rightarrow t_1' t_2} E - AppT$$

$$\frac{t \rightarrow t'}{vt \rightarrow vt'} E - AppV$$

0.6 Satz. Fortschritt
Wenn $\vdash t : T$, dann $t \rightarrow t'$ oder $t = v$.

0.7 Lemma. Inversion, Erzeugung, Generation

1. Wenn $\Gamma \vdash x : T$, dann $x : T \in T$
2. Wenn $\Gamma \vdash \lambda x : S.t : R$, dann $R = S \rightarrow T$ und $\Gamma, x : S \vdash t : T$
3. Wenn $\Gamma \vdash t_1 t_2 : T$, dann $\Gamma \vdash t_1 : S \rightarrow T$ für ein S und $T \vdash t_2 : S$
4. Wenn $\Gamma \vdash true : T$, dann $T = Bool$
5. Wenn $\Gamma \vdash false : T$, dann $T = Bool$
6. Wenn $\Gamma \vdash if\ t_1\ then\ t_2\ else\ t_3 : T$, dann $\Gamma \vdash t_1 : Bool$ und $\Gamma \vdash t_2 : T$, $\Gamma \vdash t_3 : T$

Beweis. Durch Fallunterscheidung über die Typherleitung. □