LINGUAGGI FUNZIONALI

Church → λ calcolo $\simeq$ Turing

$E ::= x \mid EE \mid \lambda x. E$

TESI DI CHURCH
(mai dimostrata)

.... | 0 | 1 | 0 | 1 | 0 | 0 | 1 | ....

$\boxed{f, g}$ → lettore

McCarthy → LISP (= List Processing)

Robin Milner → ML ← Luca Cardelli, primo compilatore ML

↓

π-calcolo

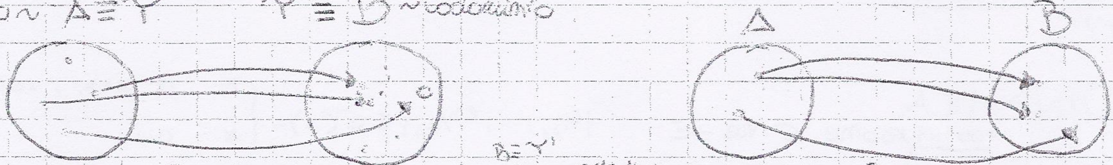
moduli

oggetti (IMP)

POLIMORFISMO DI TIPI
e TYPE INFERENCE

Differenza tra IMP e FUNZ: il secondo non ha bisogno di memoria
e' diverso quindi anche il passo di calcolo
IMP usa PC, legge e scrive memoria
FUNZ ha bisogno solo dell'ambiente, solo espressioni costanti

Dominio ~ $A \equiv \Upsilon$ $\quad \Upsilon' \equiv B$ ~ codominio



funzione $\qquad$ relazione

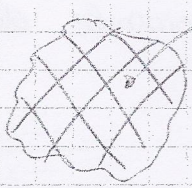
$B \equiv \Upsilon'$ $\quad$ costruttore di tipi

$f: A \to B \equiv \lambda x{:}\Upsilon. E : \Upsilon \to \Upsilon'$

$f: x \mapsto y$

$f(x) = y$

$\Lambda_{/=_\beta}$



$\lambda x. E$
$\downarrow \alpha$
$\lambda y. E\{y/x\}$

} stessa classe di equivalenza

$< \Lambda_{/=_\beta} , \to >$

λ-CALCOLO

SINTASSI

- $E ::= \underbrace{x}_{\text{variabili}} \mid \underbrace{(\lambda x.\ E)}_{\text{astrazione}} \mid \underbrace{(E_0 E_1)}_{\text{applicazione}}$ λ-espressioni

occorrenza di legame

applicazione → associa a sinistra
$E_0 E_1 E_2 \equiv (E_0 E_1) E_2$
applicazione ha precedenza su astrazione

astrazione ↓
campo di azione di $\lambda x.$ si estende a destra il più possibile
$\lambda x.\ \underrightarrow{E_0 E_1 E_2}$

- $FI(x) = \{x\}$
- $FI(\lambda x. E) = FI(E) \setminus \{x\}$
- $FI(E_0 E_1) = FI(E_0) \cup FI(E_1)$

SEMANTICA STATICA

// $E ::= x \mid (\lambda x : \tau.\ E) \mid (E_0 E_1)$

- $\Delta \vdash x : \Delta(x)$
- $\dfrac{\Delta[x:\tau] \vdash E : \tau'}{\Delta \vdash \lambda x : \tau.\ E : \tau \to \tau'}$
- $\dfrac{\Delta \vdash E_0 : \tau \to \tau' \qquad \Delta \vdash E_1 : \tau}{\Delta \vdash E_0 E_1 : \tau'}$

SEMANTICA DINAMICA

=sostituzione $\sigma : Var \to E$ $\{E_1/x_1, \dots, E_n/x_n\} \equiv \sigma(x) = \begin{cases} E_i & \text{if } x = x_i,\ i \in [1..n] \\ x & \text{o.w.} \end{cases}$

CONVERSIONI

- $\lambda x. E \xrightarrow{\alpha} \lambda y. E\boxed{\{y/x\}},\ y \notin FI(E)$ // cambiamento del nome non altera funz. stessa classe di equivalenza
  solo su $FI(E)$
  $E\{E'/x\}$ VALIDA SSE $BI(E) \cap FI(E) = \emptyset$ ⇒ altrimenti ho cattura di variabili
- $(\lambda x. E) E' \xrightarrow{\beta} E\{E'/x\}$ // posso cambiare classe di equivalenza
  passaggio di parametri per nome (CALL BY NAME)
- $(\lambda x. E x) \xrightarrow{\eta} E,\ x \notin FI(E)$ // equivalenza estensionale di funzioni

RIDUZIONI

- $\dfrac{E \xrightarrow{\alpha} E'}{E \to E'}$ - $\dfrac{E \xrightarrow{\beta} E'}{E \to E'}$ - $\dfrac{E \xrightarrow{\eta} E'}{E \to E'}$
- $\dfrac{E \to E'}{\lambda x. E \to \lambda x. E'}$ - $\dfrac{E \to E'}{E E_0 \to E' E_0}$ - $\dfrac{E \to E'}{E_0 E \to E_0 E'}$

FORMA NORMALE di A = non posso più applicare riduzioni
Non è detto che esista una forma normale
⇓
Se esiste la trovo con LEFT MOST-OUTER MOST (lo) ↗ è deterministico

$lo(x) = \emptyset$ $lo(\lambda x. E) = \begin{cases} lo(E) & \text{"no } \eta\text{"} \\ \lambda x. E & \text{"sì } \eta\text{"} \end{cases}$ $lo(E_0 E_1) = \begin{cases} E_0 E_1 & E_0 = \lambda x. E \\ lo(E_0) & E_0 \neq \lambda x. E \\ lo(E_1) & lo(E_0) = \emptyset \end{cases}$

$$\dfrac{\dfrac{\dfrac{\emptyset[x:\tau_0][y:\tau_1][z:\tau_2] \vdash y:\tau_1}{\emptyset[x:\tau_0][y:\tau_1] \vdash \lambda z:\tau_2 . y : \tau_2 \to \tau_1}}{\emptyset[x:\tau_0] \vdash \lambda y:\tau_1 . (\lambda z:\tau_2 . y) : \tau_1 \to (\tau_2 \to \tau_1)}}{\emptyset \vdash \lambda x:\tau_0 . (\lambda y . \tau_1 . (\lambda z:\tau_2 . y)) : \tau_0 \to (\tau_1 \to (\tau_2 \to \tau_1))}$$

Esempio semantica dinamica - α conversione

$$(\lambda x . x(\lambda x . x)\overset{\text{libera}}{y})\{y/x\}$$

$$(\underbrace{\lambda y . y(\lambda y . y)y}_{\text{legata}}$$ ⇒ CATTURA DI VARIABILI → DA EVITARE $(y \notin FI(E))$

Esempio semantica dinamica - η conversione

$f : A \to B$
$g : A \to B$

$f = g \qquad \forall x \in A . f(x) = g(x) \Rightarrow f = g$ ⇒ EQUIVALENZA ESTENSIONALE DI FU

$$(\lambda x . E x) E_0 \to E x\{E_0/x\} \equiv E E_0$$

Esempio forme normali:

$= (\lambda x . xx)(\lambda x . xx) \xrightarrow{\beta} xx\{\lambda x . xx/x\} = (\lambda x . xx)(\lambda x . xx)$ ⇒ infinite β-riduzioni

$$\begin{array}{l} (\lambda x . y)\Omega \to (\lambda x . y)\Omega \to \ldots \to \\ \downarrow \\ y\{\Omega/x\} \equiv y \end{array}$$ → forma normale

Esempio lo:

$$\begin{tikzcd} & E \arrow[dl, "*"'] \arrow[dr, "*"] & \\ E' \arrow[dr, "*"'] & & E'' \arrow[dl, "*"] \\ & E''' & \end{tikzcd}$$

⇒ confluenza ⇒ TEOREMA

ESEMPIO ... SEMANTICA STATICA

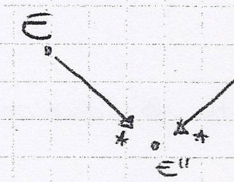
$E =_\beta E$ $\qquad \dfrac{E_0 =_\beta E_1}{E_1 =_\beta E_0}$ $\qquad \dfrac{E_0 =_\beta E_1 \quad E_1 =_\beta E_2}{E_0 =_\beta E_2}$ $\qquad \dfrac{E_0 \to E_1}{E_0 =_\beta E_1}$

$\dfrac{E_0 =_\beta E_1}{\lambda x.E_0 =_\beta \lambda x.E_1}$ $\qquad \dfrac{E_0 =_\beta E_1}{E_0 E =_\beta E_1 E}$ $\qquad \dfrac{E_0 =_\beta E_1}{E E_0 =_\beta E E_1}$

$E =_\beta E'$ sse $\exists E_0, E_1, \dots, E_n$ t.c. $E_0 \equiv E$, $E_n \equiv E'$
$\forall i \in [1..n-1]. (E_i \to E_{i+1} \vee E_{i+1} \to E_i)$

$E = E$ $\qquad \dfrac{E = E'}{E' = E}$ $\qquad \dfrac{E = E' \quad E' = E''}{E = E''}$

$E = E'$ sse $\exists E_0, \dots, E_n$
$E_0 = E$, $E_n = E'$ $\qquad \forall i < n. (E_i \to E_{i+1} \vee E_{i+1} \to E_i)$



$$\dfrac{E = E'' \quad , \quad \dfrac{E' = E''}{E'' = E'}}{E = E'}$$

$$E ::= x \mid E E \mid \lambda x. E$$

$$\alpha ::= \lambda x. E \xrightarrow{\alpha} \lambda y. E\{y/x\} \longrightarrow$$ sostituendo y alle occorrenze libere di E

$$\beta ::= (\lambda x. E)E' \xrightarrow{\beta} E\{E'/x\} \longrightarrow$$ è valida sss i nomi legati di E $\cap$ nomi liberi di $E' = \emptyset$

$$B(E) \cap FV(E') = \emptyset$$

[eta]

$$\eta ::= \lambda x.(Ex) \xrightarrow{\eta} E, \; x \notin FV(E)$$

$$\frac{E \xrightarrow{\alpha} E'}{E \longrightarrow E'} \qquad \frac{E \xrightarrow{\beta} E'}{E \longrightarrow E'} \qquad \frac{E \xrightarrow{\eta} E'}{E \longrightarrow E'}$$

$$\frac{E \longrightarrow E'}{\lambda x. E \longrightarrow \lambda x. E'} \qquad \frac{E \longrightarrow E'}{E E_0 \longrightarrow E' E_0} \qquad \frac{E \longrightarrow E'}{E_0 E \longrightarrow E_0 E'}$$

$$E = E \qquad \frac{E = E'}{E' = E} \qquad \frac{E = E', \; E' = E''}{E = E''} \qquad \frac{E \longrightarrow E'}{E = E'}$$

Conversioni conservano la congruenza

$$\frac{E = E'}{E E_0 = E' E_0} \qquad \frac{E = E'}{E_0 E = E_0 E'} \qquad \frac{E = E'}{\lambda x. E = \lambda x E'}$$

$E = E'$ sss $\exists E_0, \dots, E_n$ tc

$$E_0 = E, \; E_n = E', \; \forall i < n. \left(E_i \to E_{i+1} \overset{\text{or}}{\vee} E_{i+1} \to E_i\right)$$

$$\begin{tikzcd} E \arrow[dr] & & E' \arrow[dl] \\ & E'' & \end{tikzcd} \qquad \frac{E = E'' \quad E' = E''}{E = E'}$$

Assumiamo di rappresentare ~~lo zero~~ gli interi mediante il lambda term.

$0 \triangleq \lambda f x . x$

$1 \triangleq \lambda f x . f x$

$2 \triangleq \lambda f x . f(f x)$

$3 \triangleq \lambda f x . f(f(f x))$

$\vdots$

$n \triangleq \lambda f x . f^n x$

funzione successore:

$succ \triangleq \lambda m f x . f(m f x)$

es:

$succ\ 0 = 1$ devo dimostrarlo:

$$\underbrace{[\lambda m f x . f(m f x)]}_{\lambda x . E}\underbrace{(\lambda f x . x)}_{E'} \xrightarrow{\beta} \underbrace{[\lambda f x . f(m f x)]\left\{\frac{(\lambda f x . x)}{m}\right\}}_{\text{posso eseguirlo perché in ① non ci sono variabili libere}}$$

x corrisponde a m

$$\equiv \underbrace{\lambda f x . f((\lambda f x . x) f x)}_{\text{non essendoci parentesi in } (\lambda f x . x) f x \text{ applico f al termine a sx}}$$

$$① \ \underbrace{(\lambda f x . x)}_{\lambda x . E}\underbrace{f}_{E} \xrightarrow{\beta} (\lambda x . x)\left\{\frac{f}{f}\right\} \equiv \lambda x . x$$

x coincide con f

①

$(\lambda f x . x)$

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

$\boxed{\lambda f x . f x \equiv 1}$

$(\lambda x . x) x \xrightarrow{\beta} x\left\{\frac{x}{x}\right\} \equiv x$

$(\lambda x . x) \longrightarrow x$

$f((\lambda x . x) x) \longrightarrow f x$

$\lambda f x . f((\lambda x . x) x) \longrightarrow \lambda f x . f x$

• PLUS $\triangleq \lambda mnfx.\, fx.\, \cancel{m}nfx(mfx)$ λ

Dimostro PLUS 1 2 $\equiv$ 3

$\equiv$

$([\lambda mnfx.\, nfx(mfx)]\,1)\,2 \to ([\lambda nfx.\, nf(1fx)]\,2) \to$

$\lambda fx.\, 2f(1fx) \equiv \lambda fx.\, 2f((\lambda fx.\, fx) fx) \to$

$\lambda fx.\, 2f((\lambda x.\, fx)\, x) \to \lambda fx.\, 2f(fx) \equiv \lambda fx.\, (\lambda fx.\, f(fx)) f (fx) \to$

$\lambda fx.\, (\lambda x.\, f(fx))(fx) \to \boxed{\lambda fx.\, f(f(fx)) \equiv 3}$

• MULT $\triangleq \lambda mnfx.\, m(nf)x \triangleq \lambda mn.\, m(\text{PLUS}\, n)\, 0$

$\boxed{\text{NB: } xfn = (\lambda fx.\, f^n x) fx \to (\lambda x.\, f^n x) x \to f^n x}$

MULT 2 3

$[\lambda mnfx.\, m(nf)x]\,2\,3 \to [\lambda nfx.\, 2(nf)x]\,3 \to \lambda fx.\, 2(3f)x \equiv$

$\lambda fx.\, (\lambda fx.\, f(fx))(3f)x \to \lambda fx.\, [\lambda x.\, (3f)((3f)x)]\, x \to$

$\lambda fx.\, \underbrace{3}_{n}f\underbrace{((3f)x)}_{x} \xrightarrow{*} \lambda fx.\, f^3(3fx) \xrightarrow{*} \lambda fx.\, f^3(f^3 x)$

nfx

$\downarrow$

$\boxed{\lambda fx.\, f^6 x \equiv 6}$

$$\text{PRED} \triangleq \lambda n f x . n(\lambda g h . h(g f))(\lambda u . x)(\lambda u . u)$$

$$\text{SUB} \triangleq \lambda m n . n \, \text{PRED} \, m$$

$$\text{SUB } 2 \ 1 \equiv [\lambda m n . n \, \text{PRED} \, m] \, 2 \ 1 \xrightarrow{2}$$

$$[1 \ \text{PRED}] 2 \equiv [(\lambda f x . f x) \, \text{PRED}] \, 2 \longrightarrow (\lambda x . \text{PRED} \, x) 2 \longrightarrow$$

$$\boxed{\text{PRED } 2 \equiv 1}$$

/5/5/14/

PREDICATI LOGICI

TRUE $\triangleq \lambda xy.x$

FALSE $\triangleq \lambda xy.y$

AND $\triangleq \lambda pq.pqp$

OR $\triangleq \lambda pq.ppq$

NOT $\triangleq \lambda pab.pba$

IFTHENELSE $\triangleq \lambda p.p$

ISZERO $\triangleq \lambda n.n(\lambda x.\text{FALSE})\text{TRUE}$

$0 \triangleq \lambda fx.x$

$n \triangleq \lambda fx.f^n x$

PLUS $\triangleq \lambda mnfx.\ nf(mfx)$

AND TRUE FALSE = FALSE

$(\lambda pq.pqp)(\lambda xy.x)(\lambda xy.y) \to$

$(\lambda q.(\lambda xy.x)q(\lambda xy.x))(\lambda xy.y) \to$

$(\lambda xy.x)(\lambda xy.y)(\lambda xy.x) \to$

$(\lambda y.(\lambda xy.y))(\lambda xy.x) \to$

$\lambda xy.y = \text{FALSE}$

ISZERO 3 = FALSE

$(\lambda n.n(\lambda x.\text{FALSE})\text{TRUE})(3) \to$

$[(\lambda fx.f[f(fx)])(\lambda x.\text{FALSE})\text{TRUE}] \to$

$[(\lambda x.(\lambda x.\text{FALSE})^3 x)\text{TRUE}] \to$

$(\lambda x.\text{FALSE})^3\text{TRUE} \longrightarrow \text{FALSE}$

OR FALSE TRUE = TRUE

$(\lambda pq . ppq)(\lambda xy . y)(\lambda xy . x) \rightarrow$

$(\lambda q . (\lambda xy . y)(\lambda xy . y) q)(\lambda xy . x) \rightarrow$

$(\lambda xy . y)(\lambda xy . y)(\lambda xy . x) \xrightarrow{*} \lambda xy . x = \text{TRUE}$

NOT (ISZERO 3) = TRUE

$(\lambda pab . pba)\left[(\lambda n . n(\lambda x . \text{FALSE})\text{TRUE})(\lambda fx . f^3 x)\right]$

$(\lambda pab . pba)\left[(\lambda fx . f^3 x)(\lambda x . \text{FALSE})\text{TRUE}\right]$

$(\lambda pab . pba)\left[(\lambda x . (\lambda x . \text{FALSE})^3 x)\text{TRUE}\right]$

$(\lambda pab . pba)\left[(\lambda x . \text{FALSE})^3 \text{TRUE}\right]$

$(\lambda pab . pba)(\text{FALSE})$

$\lambda ab . (\lambda xy . y) ba \rightarrow$

$\lambda ab . a \equiv \text{TRUE}$

LISTE

$PAIR \triangleq \lambda xyf.fxy$ $NIL\ \lambda x.TRUE$

$FIRST \triangleq \lambda p.pTRUE$

$SECOND \triangleq \lambda p.pFALSE$

$PRED \triangleq \lambda n.FIRST(n\,\phi(PAIR\ 0\ 0))$

$\phi \triangleq \lambda x.PAIR(SECOND\,x)(SUCC(SECOND\,x))$

$SUCC \triangleq \lambda nfx.f(nfx)$

$FIRST(PAIR\ \varepsilon\ \varepsilon')$

$[\lambda p.pTRUE]((\lambda xyf.fxy)\varepsilon\ \varepsilon')$

$[(\lambda xyf.fxy)\varepsilon\ \varepsilon']TRUE$

$[(\lambda yf.f\varepsilon y)\varepsilon']TRUE$

$(\lambda f.f\varepsilon\varepsilon')TRUE$

$(\lambda xy.x)\varepsilon\varepsilon' \rightarrow \varepsilon$

SUB 5 6 = 0

$(\lambda mn.\ n\ PRED\ m)\ 5\ 6 \xrightarrow{2}$

$6\ PRED\ 5$

$(\lambda fx.\ f^6 x)\ PRED\ (\lambda fx.\ f^5 x) \rightarrow$

$(\lambda x.\ (PRED)^6 x)\ (\lambda fx.\ f^5 x) \rightarrow$

$(PRED)^6 (\lambda fx.\ f^5 x) \rightarrow$

$(PRED)^5 (PRED\ 5)$

$$\text{fact} = \lambda n.\ \text{ifthenelse}\ (\text{iszero}\ n)\ 1\ (\text{mult}\ n\ [\text{fact}\ (\text{pred}\ n)])$$

Non essendoci la ricorsione non posso riusare fact nel corpo del λ

$$\underbrace{\left[\lambda f n.\ \text{ifthenelse}\ (\text{iszero}\ n)\ 1\ (\text{mult}\ n\ [f\ (\text{pred}\ n)])\right]}_{F}\ \text{fact}$$

è un punto fisso del fattoriale

$$(\underline{Y}F)_3 = [F(\underline{Y}F)]_3 \equiv [(\lambda f n.\ \text{ite}\ (\text{isz}\ n)\ 1\ (\text{mul}\ n\ f\ (\text{pred}\ n)))(\underline{Y}F)]_3$$

$$\rightarrow [\lambda n.\ \text{ite}\ (\text{isz}\ n)\ 1\ (\text{mul}\ n\ [(\underline{Y}F)(\text{pred}\ n)])]_3 \rightarrow$$

$$\text{ite}\ (\text{isz}\ 3)\ 1\ (\text{mul}\ 3\ [(\underline{Y}F)(\text{pred}\ 3)]) \xrightarrow{*}$$

$$\text{mul}\ 3\ [(\underline{Y}F)\ 2] \equiv \text{mul}\ 3\ [(F(\underline{Y}F))\ 2] \equiv$$

$$\text{mul}\ 3\ [[\lambda f n.\ \text{ite}\ (\text{isz}\ n)\ 1\ (\text{mul}\ n\ f(\text{pred}\ n))](\underline{Y}F)]\ 2 \rightarrow$$

$$\text{mul}\ 3\ [[\lambda n.\ \text{ite}\ (\text{isz}\ n)\ 1\ (\text{mul}\ n\ [(\underline{Y}F)(\text{pred}\ n)])]\ 2] \rightarrow$$

$$\text{mul}\ 3\ [\text{ite}\ (\text{isz}\ 2)\ 1\ (\text{mul}\ 2\ [(\underline{Y}F)(\text{pred}\ n)])] \xrightarrow{*}$$

$$\text{mul}\ 3\ (\text{mul}\ 2\ [(\underline{Y}F)\ 1]) = \text{mul}\ 3\ (\text{mul}\ 2\ [F(\underline{Y}F)\ 1]) =$$

$$\text{mul}\ 3\ (\text{mul}\ 2\ [\lambda f n.\ \text{ite}\ (\text{isz}\ n)\ 1\ (\text{mul}\ n\ f\ (\text{pred}\ n))(\underline{Y}F)\ 1]$$

|09/05/14|

RICORSIONE

Punto fisso: $f(x) = x$ $\quad x$: punto fisso di f.

Un COMBINATORE DI PUNTO FISSO $\underline{Y}$ è un λ-termine chiuso tale che $\forall$ λ-termine f vale $f(\underline{Y}f) = \underline{Y}f$

$\downarrow$

$FV(\underline{Y}) = \emptyset$

$$\underline{Y} \triangleq \lambda f.\,(\lambda x.\, f(xx))(\lambda x.\, f(xx))$$

Dimostro che è un combinatore di punto fisso.

$$\underline{Y}f = [\lambda f.\,(\lambda x. f(xx))(\lambda x. f(xx))]f \rightarrow$$

$$[(\lambda x. f(xx))(\lambda x. f(xx))] \rightarrow f[(\lambda x. f(xx))(\lambda x. f(xx))]$$

$$\equiv$$

$$f(\underline{Y}f)$$

Combinatore di Turing:

$$T \triangleq (\lambda xy.\, y(xxy))(\lambda xy.\, y(xxy)) \qquad Tf = f(Tf)$$

12/05/14

$\boxed{\text{FUN}}$

Sintassi:

$E ::= k \mid x \mid E_0 \text{ bop } E_1 \mid \text{uop } E_1 \mid \text{if } E_0 \text{ then } E_1 \text{ else } E_2 \mid \text{let } D \text{ into } E \mid f(ae)$

$D ::= nil \mid x : \tau = E \mid D_0 ; D_1 \mid D_0 \text{ in } D_1 \mid D_0 \text{ and } D_1 \mid \text{function } f(form) : \tau = E \mid$
$\quad \text{rec} D \mid form = ae \mid \rho$

$form ::= \circ \mid x : \tau, form \qquad ae ::= \circ \mid E, ae$

$abs ::= \lambda form . E : \tau \qquad Typ ::= int \mid bool \mid aet \to \tau$

$dv ::= K \mid abs$ (valori denotabili)

$dt ::= \tau \mid aet \to \tau$

$aet ::= \circ \mid dt, aet$

- $FV(k) = \emptyset$, $FV(x) = \{x\}$, $FV(E_0 \text{ bop } E_1) = FV(E_0) \cup FV(E_1)$, $FV(\text{uop } E) = FV(E)$
- $FV(\text{if } E_0 \text{ then } E_1 \text{ else } E_2) = FV(E_1) \cup FV(E_2) \cup FV(E_0)$
- $FV(\text{let } D \text{ into } E) = FV(D) \cup [FV(E) \setminus BV(D)]$
- $FV(f(ae)) = \{f\} \cup FV(ae)$

- $FV(nil) = \emptyset$, $FV(x : \tau = E) = FV(E)$, $FV(D_0 ; D_1) = FV(D_0) \cup [FV(D_1) \setminus BV(D_0)]$
- $FV(D_0 \text{ and } D_1) = FV(D_0) \cup FV(D_1)$, $FV(\text{function } f(form) : \tau = E) = FV(E) \setminus form$
- $FV(form) = \emptyset$, $FV(\circ) = \emptyset$, $FV(E, ae) = FV(E) \cup FV(ae)$

ESEMPIO

$$E \left[\begin{array}{l} \text{let rec function } f(x:int):int = \\ \quad \underbrace{\text{if } x=0 \text{ then } 1 \text{ else } x*f(x-1)}_{E_0} \\ \text{into } f(3) \end{array} \right.$$

Calcoliamo le variabili libere:

$$FV(E) = FV(\text{rec } D) \cup [FV(f(3)) \setminus BV(\text{rec } D)] =$$

$$\cancel{= FV(\text{rec } D) \cup [FV(f(3)) \setminus BV(\text{rec } D)]} =$$

$$= FV(\text{rec } D) \cup [(\{f\} \cup FV(3)) \setminus BV(\text{rec } D)] =$$

$$= FV(\text{rec } D) \cup [\{f\} \setminus BV(\text{rec } D)] =$$

$$\boxed{\text{Assumiamo che } FV(\text{rec}D) = FV(D)}$$

$$= FV(D) \cup [\{f\} \setminus BV(\text{rec } D)] =$$

$$= FV(E_0) \setminus \{x\} \cup [\{f\} \setminus BV(\text{rec}D)] =$$

$$= \{x, f\} \setminus \{x\} \cup [\{f\} \setminus BV(\text{rec } D)] =$$

$$= \{f\} \cup [\{f\} \setminus BV(\text{rec} D)] =$$

$$= \{f\} \cup [\{f\} \setminus \{x\}] = \{f\}$$

Assumendo che $BV(\text{rec } D) = BV(D)$

$$BV(E) = BV(\text{rec } D) \cup \underbrace{BV(f(3))}_{\phi} = BV(D) = \{f\} \cup \text{form}$$

In $BV(E)$ e $FV(E)$ compare l'elemento $\{f\}$, essendo che sono partizioni disgiunte dell'insieme dei nomi non dovrebbero averlo in comune. Questo accade nel caso della ricorsione

$$\boxed{FV(E) \cap BV(E) = \text{variabili definite ricorsivamente}}$$

Questo è un problemma nella semantica statica perché ~~nella~~ la dichiarazione della f compare nelle variabili libere.

$$\Delta[\Delta_0] \vdash x : int \quad , \quad \Delta[\Delta_0]\, f : ???$$

$$\frac{\Delta[\Delta_0] \vdash x=0 : bool \quad , \quad \Delta[\Delta_0] \vdash 1 : int \quad , \quad \Delta[\Delta_0] \vdash x * f(x-1)}{}$$

$$form : \Delta_0 \qquad \frac{\Delta[\Delta_0] \vdash E_0 : int}{}$$

$$\frac{\Delta \quad \vdash D : \Delta'}{\Delta \vdash rec\, D : \Delta'}$$

L'ideale sarebbe costruire un'ambiente che conosce già il fattore di ricorsione (funzione):

$$\frac{\Delta[\Delta'_{|V_0}] \vdash D\, \Delta'}{\Delta \vdash rec\, D : \Delta'} \quad , \quad V_0 = FV(D) \cap BV(D)$$

Δ esteso con Δ' che è ristretto a V_0 → Non si può costruire un ambiente Δ che viene esteso con Δ' che è quello che devo trovare!

⇓

Nuova tecnica di costruzione degli ambienti statici

1. Costruzione ← basato sulla sintassi: passo generare $\Delta'_{|V_0} = \{f : int \to int\}$

2. Validazione

① COSTRUZIONE

Davanti ad ogni regola c'è uno spazio vuoto che sta ad indicare che non parte da nessun ambiente (nemmeno voto)

$\vdash n : int$

$\vdash t : bool$

$\vdash nil : \emptyset$

$\vdash (x : \tau = e) : [x : \tau]$

$$\frac{\vdash D_0 : \Delta_0 \quad , \quad \vdash D_1 : \Delta_1}{\vdash D_0 ; D_1 : \Delta_0[\Delta_1]}$$

$$\frac{\vdash D_0 : \Delta_0 \quad , \quad \vdash D_1 : \Delta_1}{\vdash D_0 \text{ in } D_1 : \Delta_1}$$

$T(\circ) = \phi \quad , \quad T(x : \tau, form) = \tau, T(form)$

$$\frac{\vdash D_0 ; \Delta_0 \quad , \quad \vdash D_1 ; \Delta_1}{\vdash D_0 \text{ and } D_1 : \Delta_0, \Delta_1}$$

$\vdash \text{function} (form) : \tau = e :$

$[f : T(form) \rightarrow \tau]$

$$\frac{\vdash D : \Delta}{\vdash \text{rec } D ; \Delta}$$

② VALIDAZIONE | $\Delta \vdash D$: sto validando un'ambiente già costruito.

$$\Delta \vdash \text{nil}$$

$$\frac{\Delta \vdash D_0 ,\ \vdash D_0 : \Delta_0 ,\ \Delta[\Delta_0] \vdash D_1}{\Delta \vdash D_0 ; D_1}$$

$$\frac{\Delta \vdash E : \tau}{\Delta \vdash (x : \tau = E)}$$

$$\frac{\Delta \vdash D_0 ,\ \vdash D_0 : \Delta_0 ,\ \Delta[\Delta_0] \vdash D_1}{\Delta \vdash D_0 \text{ in } D_1}$$

$$\frac{\Delta \vdash D_0 \qquad \Delta \vdash D_1}{\Delta \vdash D_0 \text{ and } D_1} ,\ BV(D_0) \cap BV(D_1) = \emptyset$$

$$\frac{\text{form} : \Delta_0 ,\ \Delta[\Delta_0] \vdash E : \tau}{\Delta \vdash \text{function } f(\text{form}) : \tau = E}$$

$$\bullet : \emptyset$$

$$\frac{\text{form} : \Delta_0}{x : \tau , \text{form} : \Delta_0[x : \tau]}$$

$$\frac{\vdash D : \Delta' ,\ \Delta[\Delta'|_{V_0}] \vdash D}{\Delta \vdash \text{rec } D} ,\ V_0 = FV(D) \cap BV(D)$$

SEMANTICA STATICA ESPRESSIONI : chiamata di funzione.

$$\frac{\Delta \vdash ae : aet}{\Delta \vdash f(ae) : \tau} , \quad \Delta(f) = aet \to \tau$$

$$\Delta \vdash \bullet : \bullet$$

$$\frac{\Delta \vdash e : \tau \quad , \quad \Delta \vdash ae : aet}{\Delta \vdash e, ae : \tau, aet}$$

$$\circledast \quad \dfrac{\Delta \vdash 3 : int}{\Delta \vdash f(3) : int} \qquad \Delta(f) = int \to int$$

$$\dfrac{\dfrac{\dfrac{\Delta}{\vdash D : [f : int \to int]}}{\vdash rec\ D : \Delta} \qquad \dfrac{\vdash D : \Delta \;,\; \Delta \vdash D}{\emptyset \vdash rec\ D} \qquad \dfrac{\dfrac{\bullet : \emptyset}{x : int, \bullet : [x : int]} \;,\; \dfrac{\dfrac{\Delta_1 \vdash x : int \;,\; \Delta_1 \vdash 0 : int}{\Delta_1 \vdash x = 0 : bool} \quad \dfrac{1 : int}{\Delta_1 \vdash 1 : int} \quad \dfrac{\Delta_1 \vdash x : int \;,\; \dfrac{\Delta_1 \vdash x : int \;,\; \Delta_1 \vdash 1 : int}{\Delta_1 \vdash f(x-1) : int}}{\Delta_1 \vdash x * f(x-1) : int}}{\Delta[x : int] : E_0 : int}}{} \quad \circledast}{\emptyset \vdash E : int}$$