martedì 8 novembre 2016 09:1

$$Sem_d : Dec \times P \times M \longrightarrow P \times M$$

stringa generata a partire dalla categoria sintattica Dec

P è l'insieme di tutti gli ambienti

M è " " le memorie

martedì 8 novembre 2016 09:21

Dec → Type Ide = Exp; | Type Ide ;

$Sem_d(T\ x = e;, \rho, \mu) =$

~~dove $d = T\ x = e;$~~

sono nel caso in cui la dichiarazione è ottenuta con la prima produzione

martedì 8 novembre 2016 19:24

$$Sem_d(T\ x = e;, \rho, \mu) = \langle \rho[l/x]^{add}, \mu[v/l]^{add} \rangle$$

$T \in Type$
$x \in Ide$
$e \in Exp$
$\rho \in P$
$\mu \in M$

dove $v = Sem_e(e, \rho, \mu)$
$l = succloc(\mu)$

mi dà la semantica formale di una dichiarazione
$T\ x = e;$

succloc è una funzione con tipo
$succloc : M \rightarrow Loc$
Loc è l'insieme degli indirizzi (locazioni) di memoria.

succloc($\mu$) mi dà la prima locazione LIBERA nella memoria $\mu$

$$Sem_d(T\ x;, \rho, \mu) = \langle \rho[\ell/x]^{add}, \mu[?/\ell]^{add} \rangle$$

dove

$$\ell = succloc(\mu)$$

---

Semantica dei comandi

$$Com \rightarrow Ide = Exp; \mid if\ (Exp)\ Com\ else\ Com \mid$$
$$while\ (Exp)\ Com \mid \{\ Dec_list\ \ Com_list\ \}$$

$$Sem_c : Com \times P \times M \rightarrow M$$

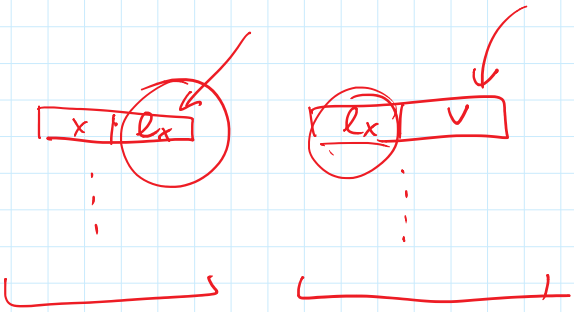
$$Sem_c(x = e;, \rho, \mu) = \mu\left[v/\rho(x)\right]^{mod}$$

dove

$$v = Sem_e(e, \rho, \mu)$$

Semantica formale di un assegnamento

Asm → Ide = Exp; |



$$Sem_c(\text{if}(e)\ c_1\ \text{else}\ c_2, \rho, \mu) = \begin{cases} Sem_c(c_1, \rho, \mu) & \text{se } Sem_e(e, \rho, \mu) = 1 \\ Sem_c(c_2, \rho, \mu) & \text{se } Sem_e(e, \rho, \mu) = \emptyset \end{cases}$$

$$Com \rightarrow \quad \text{if}\ (Exp)\ Com\ \text{else}\ Com \mid$$

$$Sem_c(\text{if } (e)\ c_1 \text{ else } c_2, \rho, \mu) = \mu'$$

dove

$$1 = Sem_e(e, \rho, \mu)$$

$$\mu' = Sem_c(c_1, \rho, \mu) \equiv Sem_c(c_1, \rho, \mu)$$

Definizione equivalente a quella della pagina precedente

$$Sem_c(\text{if } (e)\ c_1 \text{ else } c_2, \rho, \mu) = \mu'$$

dove

$$\emptyset = Sem_e(e, \rho, \mu)$$

$$\mu' = Sem_c(c_2, \rho, \mu)$$

$$\mathrm{Sem}_c(\mathrm{if}\ (e)\ c,\ \rho,\ \mu) = \begin{cases} \mathrm{Sem}_c(c, \rho, \mu) & \text{se } \mathrm{Sem}_e(e, \rho, \mu) = 1 \\ \mu & \text{se } \mathrm{Sem}_e(e, \rho, \mu) = \emptyset \end{cases}$$

$$\mathrm{Sem}_c(\text{while}(e)\,c, \rho, \mu) = \begin{cases} \mu & \text{se } \mathrm{Sem}_e(e, \rho, \mu) = \emptyset \\ \mathrm{Sem}_c(\text{while}(e)\,c, \rho, (\mathrm{Sem}_c(c, \rho, \mu))) & \text{se } \mathrm{Sem}_e(e, \rho, \mu) = 1 \end{cases}$$

$=$

$$Sem_c(\text{while}(e)\,c, \rho, \mu) = \mu$$

dove

$$\emptyset = Sem_e(e, \rho, \mu)$$

---

$$Sem_c(\text{while}(e)\,c, \rho, \mu) = \mu''$$

dove

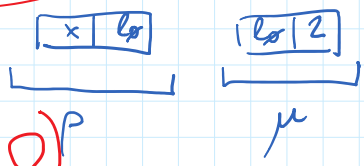
$$1 = Sem_e(e, \rho, \mu)$$

$$\mu' = Sem_c(c, \rho, \mu)$$

$$\mu'' = \underline{Sem_c(\text{while}(e)\,c, \rho, \mu')}$$

martedì 8 novembre 2016 09:58

$\rho = \{\langle x, l_\emptyset\rangle\}.\Omega \qquad \mu = \{\langle l_\emptyset, 2\rangle\}.\Omega$




while $(x>0)$ $x=x-1;$

$Sem_c(\text{while}(x>0)\ x=x-1;, \{\langle x, l_\emptyset\rangle\}.\Omega, \{\langle l_\emptyset, 2\rangle\}.\Omega)$

$= \{\text{def } Sem_c,\ e = x>0,\ c = x=x+1,\ Sem_e(x>0, \rho, \mu) = 1\}$

$Sem_c(\text{while}(x>0)\ x=x-1;, \rho, \underbrace{Sem_c(x=x-1, \rho, \mu)})$

$= \{Sem_c(x=x-1, \rho, \mu) = \mu[\cancel{\rho(x)}\ l_\emptyset]\}$ mod

$Sem_c(\text{while}(x>0)\ x=x-1, \rho, \{\langle l_\emptyset, 1\rangle\}.\Omega)$ $\mu'$

$= \{Sem_e(x>0, \rho, \{\langle l_\emptyset, 1\rangle\}.\Omega) = 1\}$

$Sem_c(\text{while}(x>0), \rho, Sem_c(x=x-1; \rho, \{\langle l_\emptyset, 1\rangle\}.\Omega))$

$= \{Sem_c(x=x-1, \rho, \{\langle l_\emptyset, 1\rangle\}.\Omega) = \{\langle l_\emptyset, \emptyset\rangle\}.\Omega\}$

$Sem_c(\text{while}(x>0)\ x=x-1;\ \rho, \{\langle l_\emptyset, \emptyset\rangle.\Omega\})$

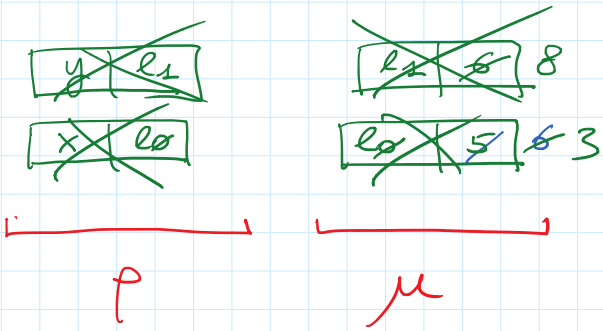
$= \{Sem_e(x>0, \rho, \{\langle l_\emptyset, \emptyset\rangle\}.\Omega) = \emptyset$

$\{\langle l_\emptyset, \emptyset\rangle\}.\Omega$

# Semantica di un BLOCCO

martedì 8 novembre 2016 10:25

```
{ int x = 5;
  x = x + 1;
  { int y = 6;
    y = x + 2;
  }
  x = 3;
}
```



# Blocco

Sintassi

Com → . . . . | { Dec_list Com_list }

Dec_list → Dec | Dec Dec_list

Com_list → Com | Com Com_list

Dec_list → ε | Dec Dec_list

CORRETTO

{ }

non posso avere sequenze vuote di dichiarazioni

{ x = x + 1 ; y = y + 1; } ← CORRETTO

$$Sem_{dl} : \underline{Dec_list \times P \times M} \longrightarrow P \times M$$

$$Sem_{cl} : Com_list \times P \times M \longrightarrow M$$

$$Sem_{dl}(d, \rho, \mu) = Sem_d(d, \rho, \mu)$$

$$Sem_{dl}(d\ dl, \rho, \mu) = Sem_{dl}(\underline{dl, \rho', \mu'})$$

dove

$$\langle \rho', \mu' \rangle = Sem_d(d, \rho, \mu)$$

$$\cancel{= Sem_{dl}(dl, Sem_d(d, \rho, \mu))}$$

$$\cancel{= Sem_{dl}(dl, \langle \rho', \mu' \rangle)}$$

martedì 8 novembre 2016 10:41

$$Sem_{cl} : Com_list \times P \times M \longrightarrow M$$

$$Sem_{cl}(c, \rho, \mu) = Sem_c(c, \rho, \mu)$$

$$Sem_{cl}(c\ cl, \rho, \mu) = Sem_{cl}(cl, \rho, \mu')$$

dove

$$\mu' = Sem_c(c, \rho, \mu)$$

$$Sem_{cl}(c, \rho, \mu) = \mu'$$

dove

$$\mu' = Sem_c(c, \rho, \mu)$$

---

$$Sem_{cl}(c\ cl, \rho, \mu) = \mu''$$

dove

$$\mu' = Sem_c(c, \rho, \mu)$$

$$\mu'' = Sem_{cl}(cl, \rho, \mu')$$

martedì 8 novembre 2016 10:45

$Com \rightarrow \ldots \mid \{ Dec_list \quad Com_list \}$

<u>frame vuoto</u> $\omega$

frame vuoto di ambiente $\omega : Ide \rightarrow Loc_\perp$

$\omega(x) = \perp$ per ogni $x \in Ide$

frame vuoto di memoria $\omega : Loc \rightarrow Val_\perp$

$\omega(l) = \perp$ per ogni $l \in Loc$

martedì 8 novembre 2016 10:48

$$Sem_c(\{dl\ cl\}, \rho, \mu)$$

dove

$$\langle \varphi \cdot \rho, \nu \cdot \mu \rangle = Sem_{dl}(dl, \omega \cdot \rho, \omega \cdot \mu)$$

sono i frame vuoti accresciuti delle dichiarazioni dl

rimangono invariati. Le dichiarazioni dl modificano solamente il frame in testa alla pile

martedì 8 novembre 2016 10:53

$$Sem_c(\{dl\ cl\}, \rho, \mu) = \mu'$$

dove

$$\langle \varphi.\rho, \nu.\mu \rangle = Sem_{dl}(dl, \omega.\rho, \omega.\mu)$$

$$\nu'.\mu' = Sem_{cl}(cl, \varphi.\rho, \nu.\mu)$$

Exp → ..... | * Ide | & Ide

non il valore dell'identificatore
ma il valore puntato dal valore
dell'identificatore

non il valore dell'identificatore
ma il suo indirizzo (locazione)

Com → .... | * Ide = Exp; | ...

Dec → .... | Type * Ide = Exp;

martedì 8 novembre 2016 11:01

$$Sem_e(*x, \rho, \mu) = \mu(\mu(\rho(x)))$$

valore di x

loc di x

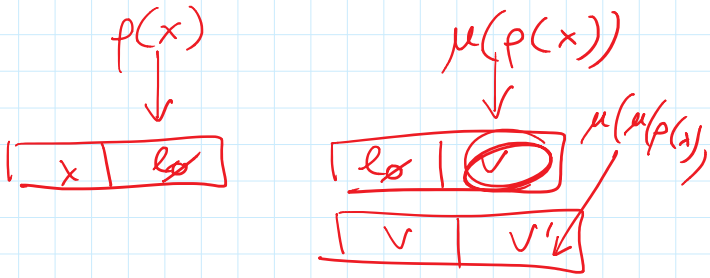
$$Sem_e(\&x, \rho, \mu) = \rho(x)$$

il valore di x $\mu(\rho(x))$ viene interpretato come la locazione di memoria da cui prendere il valore.

$$Sem_c(*x = e;, \rho, \mu) = \mu[v/\mu(\rho(x))]^{mod}$$

dove $v = Sem_e(e, \rho, \mu)$

$$Sem_d(T *x = e;, \rho, \mu)$$



martedì 8 novembre 2016 11:06

$$\text{Sem}(x = e;, \rho, \mu) = \mu\left[{}^{v}/_{p(x)}\right]^{mod}$$

dove

$$\text{Sem}(*x = e;, \rho, \mu) = \mu\left[{}^{v}/_{\mu(p(x))}\right]^{mod}$$

d