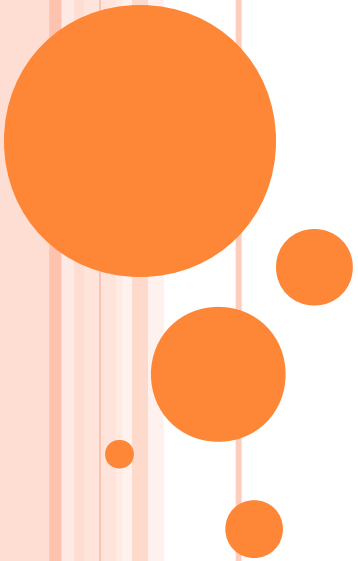

TRIPLE DI HOARE: SEQUENZE (ARRAY) E AGGIORNAMENTO SELETTIVO

Corso di Logica per la Programmazione

A.A. 2013/14

SEQUENZE: SINTASSI

- Estendiamo il linguaggio per usare *array* o *sequenze*
- Scriviamo **a : array [0,n) of T** per dire che **a** è una variabile di tipo:
  "sequenza di elementi di tipo **T** con dominio **[0,n)**",
  dove **T** può essere **int** o **bool**
- Il dominio di **a** viene indicato come *dom*(**a**)
- Scriviamo **a[E]** per denotare l'elemento di **a** di posizione **E**
  - Esempio: **a[0], a[4], a[2*x+1], a[a[0]+a[4]], ...**
- La **sintassi** delle espressioni diventa:

Exp ::= Const | Id | Ide[Exp] | (Exp) | Exp Op Exp | not Exp

SEQUENZE: SEMANTICA

- Ricordiamo che uno **stato** σ è una funzione σ : Ide → B ∪ Z
- Estendiamo il concetto di stato: se **a** è un array di tipo **T**,

  σ(**a**): *dom*(a) → **B** se **T** = **bool**

  σ(**a**): *dom*(a) → **Z** se **T** = **int**

- Esempio: **a: array [0,4) of int**

2	1	10	6
0	1	2	3

  - σ(**a**): **[0,4) → int** = {<0,2>, <1,1>, <2,10>, <3,6>}

- Estendiamo la funzione di **interpretazione semantica**:

  E(Ide[Exp],σ) = E(Ide,σ)(E(Exp,σ)) **se** E**(Exp,σ)** ∈ ***dom*(Ide)**

- Esempio: E(**a[0]**,σ) = E(**a**,σ) (E(**0**,σ)) = σ(**a**)(**0**) = **2**

SEQUENZE: SEMANTICA

- Esempio: valutazione di **a**[**a**[0]+**a**[1]] nello stato σ tale che
  - σ(**a**): **[0,4)** → **int** = {<0,2>, <1,1>, <2,10>, <3,6>}

2	1	10	6
0	1	2	3

E(**a**[**a**[0] + **a**[1]],σ) =

E(**a**,σ) (E(**a**[0],σ) + E(**a**[1],σ)) =

σ(**a**)(E(**a**,σ)(0) + E(**a**,σ)(1)) =

σ(**a**)(σ(**a**)(0) + σ(**a**)(1)) =

σ(**a**)(2+1) =

σ(**a**)(3) =

6

SEQUENZE: SEMANTICA

- **Attenzione:** Ide[Exp] non è sempre definito
  - Estendiamo la funzione *def* :

$$def(\text{Ide[Exp]}) = def(\text{Exp}) \wedge \text{Exp} \in dom(\text{Ide})$$

- Quindi abbiamo i seguenti casi in cui un'espressione potrebbe non essere definita:
  - $def(\text{E mod E}') = def(\text{E div E}') = def(\text{E}) \wedge def(\text{E}') \wedge \text{E}' \neq 0$
  - $def(\text{Ide[Exp]}) = def(\text{Exp}) \wedge \text{Exp} \in dom(\text{Ide})$

- Nota: le operazioni non possono essere applicate a sequenze, ma solo a singoli elementi di sequenze.
  Es: **a[2] < b[3]**, **a[2] * y + x** , ma non **a + b** !

AGGIORNAMENTO SELETTIVO

- Ogni elemento di una sequenza è una variabile, quindi può comparire a sinistra di un assegnamento. Es: **a[3] := 5**
- Estendiamo il comando di assegnamento: **Ide_List := Exp_List**
- Formalmente, cambia la definizione di **Ide_List** come segue:

  Ide_List ::= Ide | Ide, Ide_List | Ide[Exp] | Ide[Exp], Ide_List
- Un comando di assegnamento del tipo **v[E] := E′** è chiamato **aggiornamento selettivo**.
- L'effetto è di transire, a partire da uno stato σ, allo stato

  $$\sigma[w/v], \qquad \text{dove} \qquad w = v[E(E',\sigma)/E(E,\sigma)],$$

  a patto che le due espressioni E ed E' siano definite in σ, e che il valore di E in σ stia nel dominio di v.

AGGIORNAMENTO SELETTIVO

- La semantica del comando di **aggiornamento selettivo** è data dal seguente assioma (AGG-SEL):

$$\{def(E) \wedge def(E') \wedge E \in dom(v) \wedge P[\mathbf{w}/v]\} \quad v[E] := E' \quad \{P\}$$

dove $\mathbf{w} = \mathbf{v[E'/E]}$

- Con **v[E'/E]** intendiamo l'array **v** modificato in modo tale che nella posizione **E** abbia il valore **E'**. Quindi se **w = v[E'/E]**,
  - $w(x) = E'$ se $x = E$,
  - $w(x) = v(x)$ altrimenti
- Si può usare anche la seguente regola derivata:

$$\frac{R \Rightarrow def(E) \wedge def(E') \wedge E \in dom(v) \wedge P[\mathbf{w}/v] \qquad \mathbf{w} = v[E'/E]}{\{R\}\ v[E] := E'\ \{P\}}$$

ESEMPIO DI AGGIORNAMENTO SELETTIVO

- Si verifichi la tripla:

  $\{k \in dom(v) \wedge (\forall\, i\,.\, i \in dom(v) \wedge i \neq k \Rightarrow v[i] > 0)\}$

  $v[k] := 3$

  $\{(\forall\, i\,.\, i \in \mathrm{dom}(v) \Rightarrow v[i] > 0)\}$

- Applicando la regola (AGG-SEL), è sufficiente dimostrare:

  $k \in dom(v) \wedge (\forall\, i\,.\, i \in dom(v) \wedge i \neq k \Rightarrow v[i] > 0) \Rightarrow$
  $def(k) \wedge def(3) \wedge k \in dom(v) \wedge (\forall\, i\,.\, i \in \mathrm{dom}(w) \Rightarrow w[i] > 0)$

  dove $w = v[3/k]$

- **Esercizio:** si completi la dimostrazione sfruttando la regola dell'intervallo per la quantificazione universale:

- $(\forall x.P \Rightarrow Q) \equiv \begin{cases} (\forall x.P \wedge x \neq k \Rightarrow Q) \wedge Q[k/x] & \text{se } P[k/x] \\ (\forall x.P \wedge x \neq k \Rightarrow Q) & \text{se } \sim P[k/x] \end{cases}$

ESERCIZIO: SCANSIONE DI SEQUENZA

- Verificare il seguente programma annotato che conta il numero di elementi maggiori di zero in un array **a: array [0, n) of int**.

{a : array [0, n) of int }

x, c := 0, 0;

{*Inv* : c = #{j : j ∈ [0, x) ∧ a[j] > 0 } ∧ x∈ [0, n] } {*t* : n - x}

while x < n **do**

    if (a[x] > 0) **then** c := c+1 **else skip** **fi** ;

    x := x + 1

endw

{*Inv* ∧ ~(x < n)}

{ c = #{j : j ∈ [0, n) ∧ a[j] > 0 } }

- Scrivere e dimostrare la Condizione di Invarianza
- Scrivere e dimostrare la Condizione di Terminazione
- Scrivere e dimostrare la Condizione di Progresso

ESERCIZIO: INCREMENTO DI SEQUENZA

- Si consideri il seguente programma annotato che incrementa tutti gli elementi di un array **a: array [0, n) of int**.

$\{n \geq 0 \ \wedge$ a : array [0, n) of int $\ \wedge (\forall k: k \in [0, n) \Rightarrow a[k] = V[k])\}$

x := 0;

$\{Inv : x \in [0, n] \wedge (\forall k: k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$

$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\}\ \{t : n - x\}$

while x < n **do**

a[x] := a[x] + 1; x := x + 1

endw

$\{Inv \ \wedge \sim(x < n)\}$

$\{\ (\forall k.\ k \in [0, n) \Rightarrow a[k] = V[k] + 1)\ \}$

- Scrivere e dimostrare la Condizione di Invarianza
- Scrivere e dimostrare la Condizione di Terminazione
- Scrivere e dimostrare la Condizione di Progresso

- SOLUZIONE: Condizione di invarianza

$\{x \in [0, n] \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k]) \wedge x < n\}$

a[x] := a[x] + 1; **x := x + 1**

$\{x \in [0, n] \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k]) \wedge \text{def}(x < n)\}$

- Per la regola della sequenza dobbiamo determinare **R** in modo che

$\{x \in [0, n) \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\}$

a[x] := a[x] + 1;

$\{\mathbf{R}\}$

x := x + 1

$\{x \in [0, n] \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\}$

- In questo caso **R** viene determinata dall'assioma per l'assegnamento

$\{x \in [0, n) \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\}$

INCREMENTO DI SEQUENZA

a[x] := a[x] + 1;

$\{\mathbf{R} \equiv (\ def(x+1) \wedge x+1 \in [0, n] \wedge (\forall k.\ k \in [0, x+1) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x+1, n) \Rightarrow a[k] = V[k])\ )\}$

x := x + 1

$\{x \in [0, n] \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\ \}$

- Ci resta da dimostrare la tripla

$\{x \in [0, n) \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])\}$

a[x] := a[x] + 1;

$\{x+1 \in [0, n] \wedge (\forall k.\ k \in [0, x] \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in (x, n) \Rightarrow a[k] = V[k])\ \}$

- Avendo a che fare con un aggiornamento selettivo, dobbiamo mostrare

$x \in [0, n) \wedge (\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])$

INCREMENTO DI SEQUENZA

$\Rightarrow$

$x+1 \in [0, n] \wedge (\forall k.\ k \in [0, x] \Rightarrow b[k] = V[k] + 1) \wedge$
$(\forall k.\ k \in (x, n) \Rightarrow b[k] = V[k]) \wedge x \in [0, n) \wedge def(a[x]+1)$

con $b = a[a[x]+1 \,/\, x]$

Alcune osservazioni:

(1) $def(a[x]+1) \equiv x \in [0, n)$ che quindi possiamo omettere

(2) $(\forall k.\ k \in [0, x) \vee k \in (x, n) \Rightarrow b[k] = a[k])$

(3) $b[x] = a[x] + 1$

$x+1 \in [0, n] \wedge (\forall k.\ k \in [0, x] \Rightarrow b[k] = V[k] + 1) \wedge$

$(\forall k.\ k \in (x, n) \Rightarrow b[k] = V[k]) \wedge x \in [0, n)$

$\equiv$ { **Ip:** $x \in [0, n)$ }

$(\forall k.\ k \in [0, x] \Rightarrow b[k] = V[k] + 1) \wedge$

$(\forall k.\ k \in (x, n) \Rightarrow b[k] = V[k])$

$\equiv$ { Intervallo, $x \in [0, x]$ }

$(\forall k.\ k \in [0, x) \Rightarrow b[k] = V[k] + 1) \wedge\ b[x] = V[x] + 1 \wedge$

$(\forall k.\ k \in (x, n) \Rightarrow b[k] = V[k])$

$\equiv$ { Osservazioni (2) e (3) precedenti }

$(\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1) \wedge\ a[x] + 1 = V[x] + 1 \wedge$

$(\forall k.\ k \in (x, n) \Rightarrow a[k] = V[k])$

$\equiv$ { **Ip:** $(\forall k.\ k \in [0, x) \Rightarrow a[k] = V[k] + 1)$ }

$a[x] + 1 = V[x] + 1 \wedge (\forall k.\ k \in (x, n) \Rightarrow a[k] = V[k])$

$\equiv$ { calcolo, Intervallo }

$(\forall k.\ k \in [x, n) \Rightarrow a[k] = V[k])$, vero per ipotesi.

ESERCIZIO: Calcolo del fattoriale

- Si consideri la seguente specifica

  {n>0} C {f = n!}

```
{n>0}
   f , x:= 1, 1;
{Inv : ?} {t : ?}
while x <= n do
    f, x :=  f * x,  x + 1;
endw
{Inv  ∧ ~(x <= n)}
{f = n!}
```

- Come determinare l'invariante e la funzione di terminazione?

- Eseguiamo manualmente il programma (es: per n = 5)**:**

{n>0}	x	f
f , x:= 1, 1;	1	1
{*Inv* : ?} {*t* : ?}	**2**	**1**
while x <= n **do**	3	2
f, x := f * x, x + 1;	4	6
endw	5	24
{*Inv* ∧ ~(x <= n)}	**6**	**120**
{f = n!}		

- Osserviamo che, ad ogni iterazione, i valori di x e f sono legati dalla seguente relazione

  f = (x - 1)!

- Scegliendo questa formula come invariante non riusciamo a dimostrare

  f = (x - 1)! ∧ ~(x <= n) ⇒ f = n!

```
{n>0}
   f , x:= 1, 1;
{Inv : f = (x - 1)! ∧ x∈[0, n+1]} {t : ?}
while x <= n do
    f, x :=  f * x, x + 1;
endw
{Inv  ∧ ~(x <= n)}
{f = n!}
```

- Aggiungendo x∈[0, n+1] in *Inv* abbiamo:

$$Inv \wedge \sim(x <= n) \Rightarrow f = n!$$

- Dimostrazione per esercizio

- Ipotesi di invarianza

$\{f = (x-1)! \wedge x \in [0, n+1)\}$ f, x := f * x, x + 1 $\{f = (x-1)! \wedge x \in [0, n+1]\}$

- Per la regola dell'assegnamento multiplo, basta dimostrare:

$f = (x-1)! \wedge x \in [0, n+1) \Rightarrow def(f*x) \wedge def(x+1) \wedge f*x = x! \wedge x+1 \in [0, n+1]$

- Partiamo dalla conseguenza (eliminando i def(...) che sono **T**):

$f * x = x! \quad \wedge \quad x+1 \in [0, n+1]$

$\equiv$ { **Ip:** $f = (x-1)!$ }

$(x-1)! * x = x! \quad \wedge \quad x+1 \in [0, n+1]$

$\equiv$ { def. fattoriale}

$x+1 \in [0, n+1]$

$\equiv$ { **Ip:** $x \in [0, n+1)$, calcolo }

T

- La funzione di terminazione (che è molto semplice) è lasciata per esercizio.