Informatica — 2014-09-10

Nota: Scrivete su **tutti** i fogli nome e matricola.

Esercizio 1. *Si illustrino le* *differenze* *tra gli algoritmi di* ricerca lineare *e* ricerca binaria *su un vettore. Anche senza presentarli sotto forma di programma in pseudo-codice, si discuta in modo succinto de: 1) il loro scopo, 2) le loro ipotesi di funzionamento, 3) il loro principio di funzionamento, 4) la loro complessità computazionale asintotica.*

Esercizio 2. *Si considerino i seguenti insiemi di regole di inferenza sui naturali:*

$$\mathcal{R} = \left\{ \frac{}{1} \right\} \cup \left\{ \frac{x}{4 \cdot x} \;\middle|\; x \in \mathbb{N} \right\} \qquad \mathcal{R}' = \left\{ \frac{}{1} \right\} \cup \left\{ \frac{x}{2 \cdot x} \;\middle|\; x \in \mathbb{N} \right\}$$

Si indichi con X il minimo punto fisso di $\hat{\mathcal{R}}$, e con X' quello di $\hat{\mathcal{R}}'$. Si dimostri che $X \subseteq X'$.

Soluzione (bozza). Per il principio d'induzione, per dimostrare $X \subseteq X'$ basta che $\hat{\mathcal{R}}(X') \subseteq X'$. Sia quindi $y \in \hat{\mathcal{R}}(X')$ e dimostriamo $y \in X'$. Visto che $y \in \hat{\mathcal{R}}(X') = \{1\} \cup \{4 \cdot x \mid x \in X'\}$, consideriamo i due casi:

- Caso $y = 1$. In questo caso, banalmente $y \in X'$ a causa della prima regola di $\mathcal{R}'$.
- Caso $y = 4 \cdot x$ per un qualche $x \in X'$. In questo caso, $2 \cdot x \in \hat{\mathcal{R}}'(X')$ per la seconda regola di $\hat{\mathcal{R}}'$. Siccome X' è punto fisso di $\hat{\mathcal{R}}'$, si può semplificare ottenendo $2 \cdot x \in X'$. Ripetendo il procedimento, si ha $2 \cdot (2 \cdot x) \in \hat{\mathcal{R}}'(X') = X'$ per la seconda regola di $\hat{\mathcal{R}}'$. Quindi, $y = 4 \cdot x \in X'$.

$\square$

Esercizio 3. *Si consideri l'estensione di IMP ottenuta aggiungendo la seguente espressione intera* eq*:*

$$\mathsf{eq}(e_1, e_2)$$

dove e_1, e_2 sono espressioni intere. Il valore di tale espressione è 1 *se e_1 ed e_2 hanno lo stesso valore, e* 0 *altrimenti.*

1. *Si formalizzi la semantica di tale espressione usando regole di inferenza.*

Siano dati c un comando di IMP esteso, ed e un'espressione arbitraria di IMP esteso. Si considerino quindi i comandi di IMP esteso seguenti:

$$\begin{aligned} c_1 = \quad & x := 0;\ ((\mathsf{if}\ e \neq 0\ \mathsf{then}\ c\ \mathsf{else\ skip});\ x := 0) \\ c_2 = \quad & x := 0;\ ((\mathsf{while}\ e \cdot \mathsf{eq}(x, 0) \neq 0\ \mathsf{do}\ (c;\ x := 1));\ x := 0) \end{aligned}$$

2. *Si dimostri che c_1 e c_2 sono comandi equivalenti nel modo seguente:*

(a) *Assumendo* $\langle c_1, \sigma\rangle \to_b \sigma'$, *giustificare brevemente e* informalmente *perché vale anche* $\langle c_2, \sigma\rangle \to_b \sigma'$.

(b) *Assumendo* $\langle c_2, \sigma\rangle \to_b \sigma'$, *giustificare* formalmente *perché vale anche* $\langle c_1, \sigma\rangle \to_b \sigma'$, *facendo riferimento alle derivazioni coinvolte. (Suggerimento: questo punto può richiedere molto tempo, visto che è necessario analizzare un po' di casi. Pertanto, si consiglia di svolgere prima tutti gli altri esercizi.)*

Soluzione (bozza).
1) Basta aggiungere le regole

$$\frac{\langle e_1, \sigma\rangle \to_e z \qquad \langle e_2, \sigma\rangle \to_e z}{\langle \mathsf{eq}(e_1, e_2), \sigma\rangle \to_e 1} \qquad \frac{\langle e_1, \sigma\rangle \to_e z_1 \qquad \langle e_2, \sigma\rangle \to_e z_2 \neq z_2}{\langle \mathsf{eq}(e_1, e_2), \sigma\rangle \to_e 0}$$

2.a) Sia σ'' lo stato dopo che è stato eseguito $x := 0$ (cioè $\sigma[x \mapsto 0]$).

Se in σ'' si ha che e vale 0, l'IF di c_1 non fa nulla, si riesegue $x := 0$ (senza effetto), con stato finale $\sigma' = \sigma''$. In tal caso, il WHILE di c_2 non fa nessun ciclo visto che $e \cdot \mathsf{eq}(\ldots)$ viene valutato in σ'', quindi valuta a $0 \cdot z = 0$. Lo stato finale è quindi σ'', anche dopo il $x := 0$ finale.

Se in σ'' si ha che e vale $z \neq 0$, l'IF di c_1 esegue c (stato $= \sigma'''$) e poi si riesegue $x := 0$ con stato finale $\sigma' = \sigma'''[x \mapsto 0]$. In tal caso, in WHILE di c_2 non fa un primo ciclo ciclo visto che $e \cdot \mathsf{eq}(\ldots)$ viene valutato in σ'', quindi valuta a $1 \cdot z \neq 0$. Si esegue quindi c (stato $= \sigma'''$) e poi $x := 1$ (stato $= \sigma'''[x \mapsto 1]$). Infine si ripete il WHILE, ma ora $e \cdot \mathsf{eq}(\ldots)$ valuta a $0 \cdot z' = 0$, e quindi si ferma il ciclo. Dopo l'ultimo $x := 0$ si ha come stato finale $\sigma'''[x \mapsto 1][x \mapsto 0] = \sigma'''[x \mapsto 0] = \sigma'$.

2.b) Sotto, chiamiamo w il comando $\mathsf{while}\ e \cdot \mathsf{eq}(x, 0) \neq 0\ \mathsf{do}\ (c;\ x := 1)$.

La derivazione per $\langle c_2, \sigma\rangle \to_b \sigma'$ deve essere della forma

$$\dfrac{\dfrac{\dfrac{}{\langle 0, \sigma\rangle \to_e 0}}{\langle x := 0, \sigma\rangle \to_b \sigma_1} \qquad \dfrac{\dfrac{D}{\langle w, \sigma_1\rangle \to_b \sigma_2} \qquad \dfrac{\overline{\langle 0, \sigma_2\rangle \to_e 0}}{\langle x := 0, \sigma_2\rangle \to_b \sigma'}}{\langle w;\ x := 0, \sigma_1\rangle \to_b \sigma'}}{\langle c_2, \sigma\rangle \to_b \sigma'}$$

dove $\sigma_1 = \sigma[x \mapsto 0]$ e $\sigma' = \sigma_2[x \mapsto 0]$. Ora analizziamo la forma di D, esaminandone i vari casi.

Caso While-False.

$$\dfrac{\dfrac{\dfrac{D'}{\langle e, \sigma_1\rangle \to_e z_1} \qquad \dfrac{D''}{\langle \mathsf{eq}(x, 0), \sigma_1\rangle \to_e z_2}}{\langle e \cdot \mathsf{eq}(x, 0), \sigma_1\rangle \to_e z_1 \cdot z_2 = 0}}{\langle w, \sigma_1\rangle \to_b \sigma_2 = \sigma_1}[While - False]$$

Qui si ha $z_2 \neq 0$, visto che $\sigma_1(x) = \sigma[x \mapsto 0](x) = 0$. Quindi, siccome $z_1 \cdot z_2 = 0$, abbiamo che $z_1 = 0$. Ricostruiamo la derivazione per c_1 come segue:

$$\dfrac{\dfrac{\dfrac{}{\langle 0,\sigma\rangle \to_e 0}}{\langle x := 0,\sigma\rangle \to_b \sigma_1} \qquad \dfrac{\dfrac{\dfrac{D'}{\langle e,\sigma_1\rangle \to_e z_1 = 0} \quad \dfrac{}{\langle \mathsf{skip},\sigma_1\rangle \to_b \sigma_1}}{\langle \mathsf{if}\ e \neq 0\ \mathsf{then}\ c\ \mathsf{else\ skip},\sigma_1\rangle \to_b \sigma_1 = \sigma_2} \quad \dfrac{\dfrac{}{\langle 0,\sigma_2\rangle \to_e 0}}{\langle x := 0,\sigma_2\rangle \to_b \sigma'}}{\langle (\mathsf{if}\ e \neq 0\ \mathsf{then}\ c\ \mathsf{else\ skip});\ x := 0,\sigma_1\rangle \to_b \sigma'}}{\langle c_1,\sigma\rangle \to_b \sigma'}$$

Caso While-True. La forma di D è la seguente:

$$\dfrac{\dfrac{\dfrac{D_1}{\langle e,\sigma_1\rangle \to_e z_1} \quad \dfrac{D_2}{\langle \mathsf{eq}(x,0),\sigma_1\rangle \to_e z_2}}{\langle e \cdot \mathsf{eq}(x,0),\sigma_1\rangle \to_e z_1 \cdot z_2 \neq 0} \qquad \dfrac{\dfrac{\dfrac{D_3}{\langle c,\sigma_1\rangle \to_b \sigma_4} \quad \dfrac{\dfrac{D_4}{\langle 1,\sigma_4\rangle \to_e 1}}{\langle x := 1,\sigma_4\rangle \to_b \sigma_3}}{\langle c; x := 1,\sigma_1\rangle \to_b \sigma_3} \quad \dfrac{D_5}{\langle w,\sigma_3\rangle \to_b \sigma_2}}{\langle c; x := 1; w,\sigma_1\rangle \to_b \sigma_2}}{\langle w,\sigma_1\rangle \to_b \sigma_2}\,[Wh$$

dove $\sigma_3 = \sigma_4[x \mapsto 1]$, e $z_1, z_2 \neq 0$. Notiamo che in D_5 dobbiamo avere per forza una While-False, visto che la guardia valuta a 0 in σ_3:

$$\dfrac{\dfrac{\vdots}{\langle e,\sigma_3\rangle \to_e z'} \qquad \dfrac{\dfrac{}{\langle x,\sigma_3 = \sigma_4[x \mapsto 1]\rangle \to_e 1} \quad \dfrac{}{\langle 0,\sigma_3\rangle \to_e 0 \neq 1}}{\langle \mathsf{eq}(x,0),\sigma_3\rangle \to_e 0}}{\langle e \cdot \mathsf{eq}(x,0),\sigma_3\rangle \to_e z' \cdot 0 = 0}$$

(Pignoleria: una derivazione per e esiste a causa della totalità). Siccome D_5 è While-False, ho che $\sigma_2 = \sigma_3$, e conseguentemente $\sigma' = \sigma_2[x \mapsto 0] = \sigma_4[\mapsto 0]$. Quindi, ricostruiamo la derivazione per c_1 come segue:

$$\dfrac{\dfrac{\dfrac{}{\langle 0,\sigma\rangle \to_e 0}}{\langle x := 0,\sigma\rangle \to_b \sigma_1} \qquad \dfrac{\dfrac{\dfrac{D_1}{\langle e,\sigma_1\rangle \to_e z_1 \neq 0} \quad \dfrac{D_3}{\langle c,\sigma_1\rangle \to_b \sigma_4}}{\langle \mathsf{if}\ e \neq 0\ \mathsf{then}\ c\ \mathsf{else\ skip},\sigma_1\rangle \to_b \sigma_4} \quad \dfrac{\dfrac{}{\langle 0,\sigma_4\rangle \to_e 0}}{\langle x := 0,\sigma_4\rangle \to_b \sigma_4[x \mapsto 0] = \sigma'}}{\langle (\mathsf{if}\ e \neq 0\ \mathsf{then}\ c\ \mathsf{else\ skip});\ x := 0,\sigma_1\rangle \to_b \sigma'}}{\langle c_1,\sigma\rangle \to_b \sigma'}$$

$\square$

Nome ______________________________ Matricola ___________

Esercizio 4. *Si dimostri formalmente la validità della tripla di Hoare seguente riempiendo le linee sottostanti con opportune asserzioni.*

$\{x = 0 \wedge y = 0 \wedge n = 2 \cdot N \geq 0\}$

while $n > 0$ do

if $x = 0$ then

$y := y + 1$

else

skip

$x := 1 - x$

$n := n - 1$

$\{y = N\}$

Giustificare qui sotto eventuali usi della regola *PrePost*.

Soluzione (bozza).

$\{x = 0 \land y = 0 \land n = 2 \cdot N \geq 0\}$
$\{INV : 2N + x = 2y + n \land 0 \leq x \leq 1 \land n \geq 0\}$ (1)
while $n > 0$ do
  $\{INV \land n > 0\}$
  if $x = 0$ then
    $\{INV \land n > 0 \land x = 0\}$
    $\{2N + 1 - x = 2(y + 1) + n - 1 \land 0 \leq 1 - x \leq 1 \land n - 1 \geq 0\}$ (2)
    $y := y + 1$
  else
    $\{INV \land n > 0 \land x \neq 0\}$
    $\{2N + 1 - x = 2y + n - 1 \land 0 \leq 1 - x \leq 1 \land n - 1 \geq 0\}$ (3)
    skip
  $\{2N + 1 - x = 2y + n - 1 \land 0 \leq 1 - x \leq 1 \land n - 1 \geq 0\}$
  $x := 1 - x$
  $\{2N + x = 2y + n - 1 \land 0 \leq x \leq 1 \land n - 1 \geq 0\}$
  $n := n - 1$
$\{INV \land \neg(n > 0)\}$
$\{y = N\}$(4)

Per le PrePost:

1) aritmetica.

2) Per ipotesi ho che $x = 0$, quindi la prima parte della tesi si semplifica a

$$2N = 2y + n$$

Siccome per ipotesi $2N + x = 2y + n$, usando di nuovo $x = 0$ si conclude.

La seconda parte della tesi $0 \leq 1 - x \leq 1$ segue dall'invariante $0 \leq x \leq 1$. La terza parte $n - 1 \geq 0$ segue da $n > 0$.

3) Per ipotesi ho che $x = 1$ (tra 0 e 1 e $\neq 0$), quindi la prima parte della tesi si semplifica a

$$2N = 2y + n - 1$$

Per ipotesi ho che $2N + x = 2y + n$, e con $x = 1$ si conclude.

Per la seconda e terza parte della tesi si ragiona come in 2).

4) Dall'invariante $n \geq 0$ e $n \leq 0$ ho che $n = 0$ quindi $2N + x = 2y$. Ne segue che x è pari, e quindi per INV è 0. Quindi $2N = 2y$ da cui la tesi.

□