LOGICA PER LA PROGRAMMAZIONE - a.a. 2018-2019
Seconda Prova Intermedia – 20/12/2018 — Soluzioni Proposte

Attenzione: Le soluzioni che seguono sono considerate corrette dai docenti. Per ogni esercizio possono esistere altre soluzioni corrette, anche molto diverse da quelle proposte.

ESERCIZIO 1
Assumendo che P, Q, R e S contengano la variabile libera x, si provi che la seguente formula è valida:

$$(\forall x\,.\,\neg P \vee (P \wedge \neg R)) \;\wedge\; (\exists x\,.\,\neg(S \wedge P) \Rightarrow \neg Q) \;\Rightarrow\; \neg(\forall x\,.\,Q \wedge R)$$

SOLUZIONE ESERCIZIO 1

Semplifichiamo la conseguenza:

$\neg(\forall x\,.\,\underline{Q \wedge R})$
$\equiv$ {$(\neg \Rightarrow)$}
$\underline{\neg(\forall x\,.\,\neg(Q \Rightarrow \neg R)}$
$\equiv$ {(De Morgan), (doppia negazione)}
$(\exists x\,.\,Q \Rightarrow \neg R)$

A questo punto utilizzando la regola della **Skolemizzazione** è sufficiente dimostrare che:

$$(\forall x\,.\,\neg P \vee (P \wedge \neg R)) \;\wedge\; (\exists x\,.\,\neg(S \wedge P) \Rightarrow \neg Q) \;\wedge\; (\neg(S(a) \wedge P(a)) \Rightarrow \neg Q(a)) \;\Rightarrow\; (\exists x\,.\,Q \Rightarrow \neg R)$$

con a costante nuova e indicando, per una generica formula X, con $X(a)$ la formula ottenuta dalla sostituzione $X[^a/_x]$.
Per dimostrare la formula partiamo dalla premessa:

$\underline{(\forall x\,.\,\neg P \vee (P \wedge \neg R)) \;\wedge\; (\exists x\,.\,\neg(S \wedge P) \Rightarrow \neg Q) \;\wedge\; (\neg(S(a) \wedge P(a)) \Rightarrow \neg Q(a))}$
$\Rightarrow$ {(sempl-$\wedge$), occor. pos.}
$(\forall x\,.\,\neg P \vee (P \wedge \neg R)) \;\wedge\; \underline{(\neg(S(a) \wedge P(a)) \Rightarrow \neg Q(a))}$
$\equiv$ {(contronominale)}
$\underline{(\forall x\,.\,\neg P \vee (P \wedge \neg R))} \;\wedge\; (Q(a) \Rightarrow S(a) \wedge P(a))$
$\Rightarrow$ {(elim-$\forall$), occor. pos., a termine chiuso}
$\underline{(\neg P(a) \vee (P(a) \wedge \neg R(a)))} \;\wedge\; (Q(a) \Rightarrow S(a) \wedge P(a))$
$\equiv$ {(complemento)}
$\underline{(\neg P(a) \vee \neg R(a))} \;\wedge\; (Q(a) \Rightarrow (S(a) \wedge P(a)))$
$\equiv$ {(elim-$\Rightarrow$), al contrario}
$(P(a) \Rightarrow \neg R(a)) \;\wedge\; (Q(a) \Rightarrow \underline{(S(a) \wedge P(a))})$
$\Rightarrow$ {(sempl-$\wedge$), occor. pos.}
$\underline{(P(a) \Rightarrow \neg R(a)) \;\wedge\; (Q(a) \Rightarrow P(a))}$
$\Rightarrow$ {(trans.-$\Rightarrow$), occor. pos.}
$\underline{Q(a) \Rightarrow \neg R(a)}$
$\Rightarrow$ {(intro-$\exists$), occor. pos. }
$(\exists x\,.\,Q \Rightarrow \neg R)$

ESERCIZIO 2
Assumendo **a: array [0, n) of int**, con $n > 0$, si formalizzi il seguente enunciato con la logica del primo ordine:

"Se il primo elemento dell'array **a** è uguale ad 1, allora tutti gli altri sono uguali alla somma dei valori precedenti. Altrimenti, se il primo elemento è diverso da 1, tutti gli altri sono uguali a 0.

SOLUZIONE ESERCIZIO 2

Presentiamo due soluzioni alternative, evidenziando il diverso uso di connettivi logici:

$$\big(\mathbf{a}[0] = 1 \Rightarrow (\forall x.\ x \in [1,n) \Rightarrow \mathbf{a}[x] = (\Sigma y : y \in [0,x)\,.\,\mathbf{a}[y])\big) \ \wedge\ \big(\neg(\mathbf{a}[0] = 1) \Rightarrow (\forall x\,.\,x \in [1,n) \Rightarrow \mathbf{a}[x] = 0)\big)$$

$$\big(\mathbf{a}[0] = 1 \ \wedge\ (\forall x.\ x \in [1,n) \Rightarrow \mathbf{a}[x] = (\Sigma y : y \in [0,x)\,.\,\mathbf{a}[y])\big) \ \vee\ \big(\neg(\mathbf{a}[0] = 1) \ \wedge\ (\forall x\,.\,x \in [1,n) \Rightarrow \mathbf{a}[x] = 0)\big)$$

ESERCIZIO 3

Per ognuna delle seguenti triple, si dica se è soddisfatta. Se lo è, fornire una dimostrazione formale; se non lo è, fornire un controesempio.

1. $\{x < y \ \wedge\ x > 0 \ \wedge\ z \geq 0\}$ `z:= z + x * y;` `x, y := z mod y, z mod x` $\{x \geq 0\}$
2. $\{x < y \ \wedge\ x > 0 \ \wedge\ z \geq 0\}$ `z:= z + x * y;` `x:= z mod y;` `y := z mod x` $\{x \geq 0\}$

SOLUZIONE ESERCIZIO 3

1. La tripla è soddisfatta. Per verificarla, trattandosi di una sequenza di comandi applichiamo la regola (SEQ). Quindi dobbiamo trovare una asserzione R tale che le seguenti triple siano soddisfatte:

   (a) $\{x < y \ \wedge\ x > 0 \ \wedge\ z \geq 0\}$ `z:= z + x * y` $\{R\}$

   (b) $\{R\}$ `x, y := z mod y, z mod x` $\{x \geq 0\}$

   Partiamo da (b). Per l'assioma dell'**Assegnamento Multiplo** la tripla è verificata per

   $$R \equiv def(z\ mod\ y) \wedge def(z\ mod\ x) \wedge (x \geq 0)[^{z\ mod\ y, z\ mod\ x}/_{x,y}]$$

   Semplificando, otteniamo

   R

   $\equiv$ {Def. di def, Sostituzione}

   $y \neq 0 \ \wedge\ x \neq 0 \ \wedge\ z\ mod\ y \geq 0$

   Quindi ci rimane da dimostrare (a). Per la Regola (ASS), tale tripla è verificata se

   $$x < y \ \wedge\ x > 0 \ \wedge\ z \geq 0 \ \Rightarrow\ \big(def(z + x * y) \ \wedge\ (y \neq 0 \ \wedge\ x \neq 0 \ \wedge\ z\ mod\ y \geq 0)[^{z+x*y}/_{z}]\big)$$

   Partiamo dalla conseguenza.

   $def(z + x * y) \wedge (y \neq 0 \ \wedge\ x \neq 0 \ \wedge\ z\ mod\ y \geq 0)[^{z+x*y}/_{z}]$

   $\equiv$ {Def. di def, Sostituzione}

   $y \neq 0 \ \wedge\ x \neq 0 \ \wedge\ (z + x * y)\ mod\ y \geq 0$

   $\equiv$ {**Ip:** $x < y \ \wedge\ x > 0$}

   $(z + x * y)\ mod\ y \geq 0$

   $\equiv$ {**Ip**: $x < y \wedge x > 0 \wedge z \geq 0$, proprietà del modulo (non negativo se applicato a numeri non negativi)}

   T

2. La tripla non è soddisfatta. Infatti, esistono vari stati σ che soddisfano la precondizione $x < y \ \wedge\ x > 0 \ \wedge\ z \geq 0$, ma per i quali l'esecuzione del comando non porta ad un nuovo stato. Ad esempio prendendo $\sigma(z) = \sigma(k * y)$ per un qualsiasi intero k, si ha che lo stato σ' ottenuto dopo l'esecuzione di `z:= z + x * y` assegna a z l'intero $k * y + x * y$, cioè $(k + x) * y$. Si osservi adesso che $(k + x) * y\ mod\ y = 0$, pertanto, nello stato σ'' ottenuto dopo l'esecuzione del comando $x := z\ mod\ y$, ad x sarà assegnato il valore 0. A questo punto, la valutazione dell'espressione $z\ mod\ x$ non da nessun risultato e quindi l'esecuzione si blocca.

ESERCIZIO 4
Assumendo **a: array [0, n) of int**, si verifichi la seguente tripla:

$$\{x \in [1,n) \ \wedge \ a[0] = 1 \wedge \big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ (a[i] > 0) \wedge (a[i] \geq a[i-1])\big)\}$$
$$a[x] := a[x-1] * 2$$
$$\{\big(\forall i \,.\, i \in [1,x] \ \Rightarrow \ a[i] \geq a[i-1]\big)\}$$

SOLUZIONE ESERCIZIO 4
Applicando la regola dell'**Aggiornamento Selettivo** (o la combinazione assioma (AGG-SEL) + regola PRE) dobbiamo verificare che:

$$P \ \Rightarrow \ def(x) \ \wedge \ def(a[x-1]*2) \ \wedge \ x \in dom(a) \ \wedge \ R[^{b}/_{a}]$$

dove $P \equiv x \in [1,n) \ \wedge \ a[0] = 1 \wedge \big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ (a[i] > 0) \wedge (a[i] \geq a[i-1])\big)$,
$R \equiv \big(\forall i \,.\, i \in [1,x] \ \Rightarrow \ a[i] \geq a[i-1]\big)$, and $b = a[^{a[x-1]*2}/_{x}]$.

Partiamo dalla conseguenza:

$\underline{def(x) \ \wedge \ def(a[x-1]*2)} \ \wedge \ x \in dom(a) \ \wedge \ R[^{b}/_{a}]$

$\equiv$ {definizione di def}

$\underline{x \in dom(a) \ \wedge \ x-1 \in dom(a)} \ \wedge \ R[^{b}/_{a}]$

$\equiv$ {$dom(a) = [0,n)$, Ip: $x \in [1,n)$, Unità, def di R, sostituzione}

$\big(\forall i \,.\, i \in [1,x] \ \Rightarrow \ b[i] \geq b[i-1]\big)$

$\equiv$ {(Intervallo-$\forall$), Ip: $x \in [1,n)$, quindi intervallo non vuoto}

$\big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ b[i] \geq b[i-1]\big) \ \wedge \ b[x] \geq b[x-1]$

$\equiv$ {Definizione di b, osservazione: per ogni $i \in [0,x)$ si ha $i \neq x$, quindi $a[i] = b[i]$}

$\big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ a[i] \geq a[i-1]\big) \ \wedge \ a[x-1]*2 \geq a[x-1]$

$\equiv$ {Ip: $\big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ (a[i] > 0) \wedge (a[i] \geq a[i-1])\big)$, quindi con (Sempl-$\wedge$), $\big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ a[i] \geq a[i-1]\big)$, Unità}

$a[x-1]*2 \geq a[x-1]$

$\equiv$ {Calcolo}

$a[x-1] \geq 0$

Si conclude per casi: se $x = 1$, abbiamo $P \Rightarrow a[0] = 1$, quindi $P \Rightarrow a[x-1] \geq 0$. Se $x > 1$, $P \ \Rightarrow \ \big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ (a[i] > 0) \wedge (a[i] \geq a[i-1])\big) \ \Rightarrow \ \big(\forall i \,.\, i \in [1,x) \ \Rightarrow \ (a[i] > 0)\big) \ \Rightarrow \ a[x-1] > 0$, perchè $x - 1 \in [1,x)$.

ESERCIZIO 5

Assumendo **a : array [0, n) of int**, si consideri il seguente frammento di programma annotato:

$\{\mathbf{T}\}$
```
c,x := 0,0;
```
{Inv: $x \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\}$}{t: $n - x$}
```
while (x < n) do
     if (a[x] mod 6 = 0)
          then c,x := c+1,x+1
          else x := x+1
     fi
endw
```
$\{c = \#\{y : y \in [0,n) \mid a[y] \ mod \ 6 = 0\}\ \}$

Si scrivano le ipotesi di progresso ed invarianza. Inoltre si dimostri l'ipotesi di invarianza.

SOLUZIONE ESERCIZIO 5

Invariante $Inv : x \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\}$
Funzione di terminazione $t : n - x$

1. **Ipotesi di Invarianza**:

$$\{x \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\} \ \wedge \ x < n\}$$

```
if (a[x] mod 6 = 0)  then c,x := c+1,x+1 else x:=x+1 fi
```

$$\{x \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\} \ \wedge \ def(x < n)\}$$

2. **Ipotesi di Progresso**:

$$\{x \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\} \ \wedge \ x < n \ \wedge \ n - x = V\}$$

```
if (a[x] mod 6 = 0)  then c,x := c+1,x+1 else x:=x+1 fi
```

$$\{n - x < V\}$$

Dimostriamo l'ipotesi di invarianza applicando la regola del **Condizionale**. Quindi dobbiamo verificare che

(5.1.1) $Inv \wedge x < n \ \Rightarrow \ def(a[x] \ mod \ 6 = 0)$

(5.1.2) $\{Inv \wedge x < n \wedge (a[x] \ mod \ 6 = 0)\}$ `c,x := c+1,x+1` $\{Inv \wedge def(x < n)\}$

(5.1.3) $\{Inv \wedge x < n \wedge \neg(a[x] \ mod \ 6 = 0)\}$ `x := x + 1` $\{Inv \wedge def(x < n)\}$

(5.1.1) Abbiamo che

$def(a[x] \ mod \ 6 = 0)$

$\equiv$ {definizione di *def*}

$x \in dom(a)$

$\equiv$ {**Ip**: $dom(a) = [0,n), x \in [0,n], x < n$}

T

(5.1.2) Per dimostrare la tripla applichiamo la regola dell' **Assegnamento Multiplo** e ci riduciamo a dimostrare

$$Inv \wedge x < n \wedge (a[x] \ mod \ 6 = 0) \ \Rightarrow \ def(c+1) \wedge def(x+1) \wedge \big(Inv \wedge def(x < n)\big)^{[c+1,x+1}/_{c,x]}$$

Partiamo dalla conseguenza

$def(c+1) \wedge def(x+1) \wedge \underline{\big(Inv \wedge def(x < n)\big)^{[c+1,x+1}/_{c,x]}}$

$\equiv$ {sostituzione}

$\underline{def(c+1) \wedge def(x+1)} \wedge Inv^{[c+1,x+1}/_{c,x]} \wedge \underline{def(x+1 < n)}$

$\equiv$ {definizione di *def*}

$Inv^{[c+1,x+1}/_{c,x]}$

$\equiv$ {sostituzione}

$x+1 \in [0,n] \ \wedge \ c+1 = \#\{y : y \in \underline{[0,x+1)} \mid a[y] \ mod \ 6 = 0\}$

$\equiv$ {calcolo}

$x+1 \in [0,n] \ \wedge \ c+1 = \underline{\#\{y : y \in [0,x] \mid a[y] \ mod \ 6 = 0\}}$

$\equiv$ {(interv-#), **Ip**: $a[x] \ mod \ 6 = 0$ }

$x+1 \in [0,n] \ \wedge \ c+1 = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\} + 1$

$\equiv$ {**Ip**: $c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\}$ }

$x+1 \in [0,n] \ \wedge \ c+1 = c+1$

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

T

(5.1.3) Applicando la regola dell' **Assegnamento** ci riduciamo a dimostrare che

$$Inv \wedge x < n \wedge \neg(a[x] \ mod \ 6 = 0) \ \Rightarrow \ def(x+1) \wedge \big(Inv \wedge def(x < n)\big)[^{x+1}/_{x}]$$

Partiamo dalla conseguenza, applicando la sostituzione

$\underline{def(x+1)} \ \wedge \ Inv[^{x+1}/_{x}] \ \wedge \ \underline{def(x+1<n)}$

$\equiv$ {definizione di def}

$Inv[^{x+1}/_{x}]$

$\equiv$ {sostituzione}

$x+1 \in [0,n] \ \wedge \ c = \#\{y : y \in \underline{[0,x+1)} \mid a[y] \ mod \ 6 = 0\}$

$\equiv$ {calcolo}

$x+1 \in [0,n] \ \wedge \ c = \underline{\#\{y : y \in [0,x] \mid a[y] \ mod \ 6 = 0\}}$

$\equiv$ {(interv-#), **Ip**: $\neg(a[x] \ mod \ 6 = 0)$ }

$x+1 \in [0,n] \ \wedge \ c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\}$

$\equiv$ {**Ip**: $c = \#\{y : y \in [0,x) \mid a[y] \ mod \ 6 = 0\}$ }

$x+1 \in [0,n] \ \wedge \ c = c$

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

T