LOGICA PER LA PROGRAMMAZIONE - a.a. 2015-2016
Primo Appello - 19/01/2016 — 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

Si dica se le seguenti proposizioni sono tautologie oppure no. Se una proposizione è una tautologia, lo si deve dimostrare senza usare le tabelle di verità; altrimenti va prodotto un controesempio mostrando esplicitamente che rende la formula falsa.

1. $((\neg Q \vee \neg P \Rightarrow R) \Rightarrow \neg S) \;\Rightarrow\; (P \Rightarrow \neg S)$
2. $(P \Rightarrow Q \wedge \neg R) \wedge \neg(P \Rightarrow S) \;\Rightarrow\; \neg(Q \Rightarrow S \vee R)$

SOLUZIONE ESERCIZIO 1

1. La formula non è una tautologia, in quanto l'interpretazione: $\{Q \mapsto \mathbf{F}, P \mapsto \mathbf{T}, R \mapsto \mathbf{F}, S \mapsto \mathbf{T}\}$ la rende falsa. La seguente valutazione (riga della tabella di verità corrispondente all'interpretazione) giustifica la risposta (i numeri indicano l'ordine in cui sono state valutate le sottoformule).:

Q	P	R	S	$((\neg$	Q	$\vee$	$\neg$	P	$\Rightarrow$	$R)$	$\Rightarrow$	$\neg$	$S)$	$\Rightarrow$	$(P$	$\Rightarrow$	$\neg$	$S)$
F	T	F	T	T	F	T	F	T	F	F	T	F	T	F	T	F	F	T
				2	1	3	2	1	4	1	5	2	1	6	1	3	2	1

2. La formula è una tautologia come mostrato di seguito. Consideriamo la conseguenza e semplifichiamola:

$$\begin{array}{ll} & \neg(Q \Rightarrow S \vee R) \\ \equiv & \quad \{(\neg\text{-}\Rightarrow)\} \\ & Q \wedge \neg(S \vee R) \\ \equiv & \quad \{(\text{De Morgan})\} \\ & Q \wedge \neg S \wedge \neg R \qquad (\dagger) \end{array}$$

Mostriamo ora che la premessa implica la formula (†):

$$\begin{array}{ll} & (P \Rightarrow Q \wedge \neg R) \;\wedge\; \neg(P \Rightarrow S) \\ \equiv & \quad \{(\neg\text{-}\Rightarrow)\} \\ & (P \Rightarrow Q \wedge \neg R) \;\wedge\; P \wedge \neg S \\ \Rightarrow & \quad \{(\text{Modus Ponens})\} \\ & Q \wedge \neg R \wedge \neg S \end{array}$$

ESERCIZIO 2

Si formalizzi il seguente enunciato usando l'alfabeto con simboli di costante $\mathcal{C} = \{C\}$ e simboli di predicato $\mathcal{P} = \{amico(\_,\_), abitano(\_,\_)\}$ rispetto all'interpretazione fissata $(\mathcal{D}, \alpha)$, dove $\mathcal{D}$ è l'insieme di tutti gli esseri viventi, e

- $\alpha(C)$ è la persona Carlo,
- $\alpha(amico)(p, q)$ è vera se e solo se p e q sono amici,
- $\alpha(abitano)(p, q)$ è vera se e solo se p e q abitano nella stessa città.

"Tutte le persone che abitano nella stessa città di Carlo hanno almeno un amico in comune."

SOLUZIONE ESERCIZIO 2

$$(\exists x \,.\, (\forall y \,.\, (abitano(y, C) \Rightarrow amico(x, y))))$$

oppure

$$(\forall x \,.\, (\forall y \,.\, abitano(x, C) \wedge abitano(y, C) \Rightarrow (\exists z \,.\, amico(x, z) \wedge amico(y, z))))$$

ESERCIZIO 3
Si provi che la seguente formula è valida (P, Q, R e S contengono la variabile libera x):

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

SOLUZIONE ESERCIZIO 3
Cominciamo col semplificare la conseguenza:

$$\begin{array}{ll}
 & \neg(\forall x \,.\, S \vee \neg R) \\
\equiv & \{(\text{De Morgan})\} \\
 & (\exists x \,.\, \neg(S \vee \neg R)) \\
\equiv & \{(\text{De Morgan}), (\text{Doppia negazione})\} \\
 & (\exists x \,.\, \neg S \wedge R) \qquad (\dagger)
\end{array}$$

Per il Principio di Skolemizzazione, per mostrare che la premessa implica la formula (†) è sufficiente dimostrare la seguente implicazione, dove **d** è una nuova costante:

$$(\forall x \,.\, P \wedge \neg Q) \;\wedge\; (\exists x \,.\, P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)) \;\wedge\; \big(P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)\big)[^{\mathbf{d}}/_{x}] \;\Rightarrow\; (\exists x \,.\, \neg S \wedge R)$$

Partiamo dalla premessa:

$$\begin{array}{ll}
 & (\forall x \,.\, P \wedge \neg Q) \;\wedge\; (\exists x \,.\, P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)) \;\wedge\; \big(P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)\big)[^{\mathbf{d}}/_{x}] \\
\Rightarrow & \{(\text{semp-}\wedge)\} \\
 & (\forall x \,.\, P \wedge \neg Q) \;\wedge\; \big(P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)\big)[^{\mathbf{d}}/_{x}] \\
\Rightarrow & \{(\text{elim-}\forall)\} \\
 & (P \wedge \neg Q)[^{\mathbf{d}}/_{x}] \wedge \big(P \Rightarrow (\neg Q \Rightarrow \neg S \wedge R)\big)[^{\mathbf{d}}/_{x}] \\
\equiv & \{\text{Sostituzione}\} \\
 & P[^{\mathbf{d}}/_{x}] \wedge \neg Q[^{\mathbf{d}}/_{x}] \wedge \big(P[^{\mathbf{d}}/_{x}] \Rightarrow (\neg Q[^{\mathbf{d}}/_{x}] \Rightarrow \neg S[^{\mathbf{d}}/_{x}] \wedge R[^{\mathbf{d}}/_{x}])\big) \\
\Rightarrow & \{(\text{Modus Ponens})\} \\
 & \neg Q[^{\mathbf{d}}/_{x}] \wedge (\neg Q[^{\mathbf{d}}/_{x}] \Rightarrow \neg S[^{\mathbf{d}}/_{x}] \wedge R[^{\mathbf{d}}/_{x}]) \\
\Rightarrow & \{(\text{Modus Ponens})\} \\
 & \neg S[^{\mathbf{d}}/_{x}] \wedge R[^{\mathbf{d}}/_{x}] \\
\Rightarrow & \{(\text{intro-}\Rightarrow)\} \\
 & (\exists x \,.\, \neg S \wedge R)
\end{array}$$

ESERCIZIO 4
Si formalizzi il seguente enunciato (assumendo **a, b: array [0, n) of int**):

"Ogni elemento dell'array **b** è maggiore
della somma dei quadrati degli elementi di **a** con indice minore o uguale al suo."

SOLUZIONE ESERCIZIO 4

$$\big(\forall x \,.\, x \in [0, n) \Rightarrow b[x] > (\Sigma y \,.\, y \in [0, x] \,.\, a[y]^2)\big)$$

ESERCIZIO 5
Si verifichi la seguente tripla di Hoare (assumendo **b: array [0, n) of int**):

$\{x \in (0,n) \ \wedge \ \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big)\}$

```
if (b[x] > b[x-1] )
   then skip
   else b[x] := b[x-1]+1
fi
```

$\{\big(\forall i \,.\, i \in (0,x] \Rightarrow b[i] > b[i-1]\big)\}$

SOLUZIONE ESERCIZIO 5
Chiamiamo Q la postcondizione: $Q = \big(\forall i \,.\, i \in (0,x] \Rightarrow b[i] > b[i-1]\big)$

Per dimostrare la tripla data applichiamo la regola del Condizionale riducendoci a dimostrare che

(5.1) $x \in (0,n) \ \wedge \ \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big) \Rightarrow def(b[x] > b[x-1])$

(5.2) $\{x \in (0,n) \ \wedge \ \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big) \wedge b[x] > b[x-1] \ \}$ `skip` $\{Q\}$

(5.3) $\{x \in (0,n) \ \wedge \ \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big) \wedge \neg(b[x] > b[x-1]) \ \}$ `b[x]:=b[x-1]+1` $\{Q\}$

Per dimostrare (5.1) partiamo dalla conseguenza

$def(b[x] > b[x-1])$
$\equiv$ {definizione di def}
$x \in dom(b) \wedge x-1 \in dom(b)$
$\equiv$ {**Ip**: $x \in (0,n) \ \wedge \ dom(b) = [0,n)$}
T

Per dimostrare (5.2), per la regola (SKIP) è sufficiente dimostrare:

$$x \in (0,n) \wedge \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big) \wedge b[x] > b[x-1] \ \Rightarrow \ Q$$

Poiché $Q = \big(\forall i \,.\, i \in (0,x] \Rightarrow b[i] > b[i-1]\big)$, l'implicazione è vera per la legge (Intervallo-$\forall$) e $b[x] > b[x-1]$.

Per la (5.3) applichiamo l'Assioma dell'Aggiornamento Selettivo, e la regola PRE. Quindi ci riduciamo a verificare che

$$\big(x \in (0,n) \ \wedge \ \big(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]\big) \wedge \neg(b[x] > b[x-1])\big) \ \Rightarrow \ def(x) \wedge def(b[x-1]+1) \wedge x \in dom(b) \wedge Q[^{b'}/_{b}]$$

dove $b' = b\,[^{b[x-1]+1}/_{x}]$.

Partiamo dalla conseguenza
$def(x) \wedge def(b[x-1]+1) \wedge x \in dom(b) \wedge Q[^{b'}/_{b}]$
$\equiv$ {definizione di def e di Q}
$x-1 \in dom(b) \wedge x \in dom(b) \wedge (\forall i \,.\, i \in (0,x] \Rightarrow b[i] > b[i-1])[b'/b]$
$\equiv$ {**Ip**: $x \in (0,n) \wedge dom(b) = [0,n)$, sostituzione}
$(\forall i \,.\, i \in (0,x] \Rightarrow b'[i] > b'[i-1])$
$\equiv$ {(Intervallo-$\forall$), $(0,x]$ non vuoto (infatti per ipotesi $x \in (0,n)$, quindi $x > 0$) }
$(\forall i \,.\, i \in (0,x) \Rightarrow b'[i] > b'[i-1]) \wedge b'[x] > b'[x-1]$
$\equiv$ {def di b', quindi $b'[k] = b[k]$ per ogni $k \in [0,x)$, e $b'[x] = b[x-1]+1$}
$(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1]) \wedge b[x-1]+1 > b[x-1]$
$\equiv$ {**Ip:** $(\forall i \,.\, i \in (0,x) \Rightarrow b[i] > b[i-1])$, calcolo}
T

ESERCIZIO 6

Si consideri il seguente programma annotato (assumendo **a: array [0, n) of int**):

$\{ n > 0 \}$

```
x := 0; c := 0
```

$\{\mathsf{Inv} : x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y])\}\{\text{t: } n - x\}$

```
while (x < n) do
    c := c - a[x] + x;
    x := x + 1
endw
```

$\{c = \big(n * (n-1)/2\big) - (\Sigma y : y \in [0, n) \,.\, a[y])\}$

Scrivere e dimostrare l'ipotesi di invarianza.

SOLUZIONE ESERCIZIO 6

Invariante $Inv : x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y])$
Funzione di terminazione $t : n - x$
Condizione $E : x < n$
Comando C : `c := c - a[x] + x; x:= x+1`

L' *Ipotesi di Invarianza* ($\{Inv \wedge E\}\ C\ \{Inv \wedge def(E)\}$) in questo caso è

$\{x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \wedge (x < n)\}$
`c := c - a[x] + x; x:= x+1`
$\{x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \ \wedge \ def(x < n)\}$

Applicando la Regola della Sequenza, dobbiamo trovare una asserzione R tale che le seguenti triple siano verificate:

(6.1) $\{x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \wedge (x < n)\}$ `c := c - a[x] + x` $\{R\}$

(6.2) $\{R\}$ `x := x + 1` $\{x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \ \wedge \ def(x < n)\}$

Per determinare R, usiamo l'Assioma dell'Assegnamento in (6.2) e troviamo

$$
\begin{array}{ll}
 & def(x+1) \wedge \big(x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \ \wedge \ def(x < n)\big)[^{x+1}/_{x}] \\
\equiv & \{\text{sostituzione, definizione di } def\text{, calcolo}\} \\
 & x + 1 \in [0, n] \ \wedge \ c = \big((x+1) * x/2\big) - (\Sigma y : y \in [0, x] \,.\, a[y]) \ \wedge \ def(x + 1 < n)
\end{array}
$$

Quindi resta da verificare la tripla (6.1) per il valore di R appena calcolato. Applicando la Regola dell'Assegnamento, dobbiamo verificare che

$$x \in [0, n] \ \wedge \ c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y]) \wedge (x < n) \ \Rightarrow \ def(c - a[x] + x) \wedge R[^{c-a[x]+x}/_{c}]$$

Partiamo dalla conseguenza, applicando la sostituzione

$$
\begin{array}{ll}
 & def(c - a[x] + x) \wedge R[^{c-a[x]+x}/_{c}] \\
\equiv & \{\text{definizione di } def\text{, definizione di } R\text{, sostituzione}\} \\
 & x \in dom(a) \wedge \big(x + 1 \in [0, n] \wedge c = \big((x+1) * x/2\big) - (\Sigma y : y \in [0, x] \,.\, a[y]) \wedge def(x + 1 < n)\big)[^{c-a[x]+x}/_{c}] \\
\equiv & \{\mathbf{Ip} : x \in [0, n], dom(a) = [0, n)\text{, sostituzione, def di } def\} \\
 & x + 1 \in [0, n] \wedge c - a[x] + x = \big((x+1) * x/2\big) - (\Sigma y : y \in [0, x] \,.\, a[y]) \\
\equiv & \{\mathbf{Ip} : x \in [0, n], x < n\text{, calcolo}\} \\
 & c - a[x] = (x^2 + x - 2x)/2 - (\Sigma y : y \in [0, x] \,.\, a[y]) \\
\equiv & \{(\text{Intervallo-}\Sigma)\text{, calcolo}\} \\
 & c - a[x] = x * (x-1)/2 - (\Sigma y : y \in [0, x) \,.\, a[y]) - a[x] \\
\equiv & \big\{\mathbf{Ip} : c = \big(x * (x-1)/2\big) - (\Sigma y : y \in [0, x) \,.\, a[y])\text{, calcolo}\big\} \\
 & \mathbf{T}
\end{array}
$$