

LOGICA PER LA PROGRAMMAZIONE (A–B) - a.a. 2014-2015

SECONDO COMPITINO - 18/12/2014

Attenzione: Scrivere **nome, cognome, matricola** e **corso** in alto a destra su ogni foglio che si consegna.

ESERCIZIO 1

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

$$(\forall x. P \Rightarrow Q) \wedge \neg(\exists x. \neg(R \Rightarrow Q)) \wedge (\exists x. P) \Rightarrow (\exists x. \neg(Q \Rightarrow \neg P))$$

ESERCIZIO 2

Assumendo \mathbf{a} : **array** $[0, n)$ **of** **nat** con $n > 0$, si formalizzi il seguente enunciato:

“Nell’array \mathbf{a} c’è una posizione tale che la somma di tutti gli elementi dell’array fino a quella posizione compresa è uguale alla somma degli elementi che la seguono.”

Per esempio, dei seguenti array il primo soddisfa la proprietà, il secondo no:

4	0	1	3
---	---	---	---

1	8	4	3	5	2
---	---	---	---	---	---

ESERCIZIO 3

Si dica se la seguente tripla è verificata, motivando formalmente la risposta:

$$\{x = A \wedge y = B\} x := x + y; y := x - y; x := x - y \{x = B \wedge y = A\}$$

ESERCIZIO 4

Assumendo \mathbf{a} , \mathbf{b} : **array** $[0, n)$ **of** **int**, si consideri il seguente frammento di programma annotato:

```
{cond = true ∧ z = 0}
{Inv: z ∈ [0, n] ∧ (cond ≡ (∀x. x ∈ [0, z] ⇒ a[x] = b[x]))}{t: n - z}
while (z < n) do
  if not(a[z] = b[z])
    then cond := false
    else skip
  fi;
  z := z + 1
endw
{cond ≡ (∀x. x ∈ [0, n] ⇒ a[x] = b[x])}
```

Si scriva e si dimostri l’ipotesi di invarianza.

ESERCIZIO 5

Assumendo \mathbf{b} : **array** $[0, n)$ **of** **int**, si verifichi la seguente tripla:

$$\{x \in [1, n) \wedge (\forall i. i \in [0, x] \Rightarrow b[i] = i^2)\}$$
$$\mathbf{b}[x] := \mathbf{b}[x - 1] + 2 * x - 1$$
$$\{(\forall i. i \in [0, x] \Rightarrow b[i] = i^2)\}$$