

LOGICA PER LA PROGRAMMAZIONE (A,B) - a.a. 2012-2013

4 Febbraio 2013–SECONDO APPELLO

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

ESERCIZIO 1

Si provi che la seguente proposizione è una tautologia:

$$\neg(P \vee S) \wedge (R \Rightarrow \neg(P \vee S \Rightarrow Q)) \Rightarrow \neg R$$

ESERCIZIO 2

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

$$\neg(\exists x. P \wedge \neg R) \wedge (R[a/x] \Rightarrow \neg S[a/x]) \Rightarrow (\exists x. \neg(P \wedge S)).$$

ESERCIZIO 3

Utilizzando il calcolo del primo ordine si formalizzi il seguente enunciato dichiarativo, indicando esplicitamente l'interpretazione intesa:

“I docenti del Corso di Studio in Informatica (CdS-Inf) insegnano almeno un corso del CdS-Inf, ma non tutti insegnano solo corsi del CdS-Inf.”

ESERCIZIO 4

Assumendo **a**, **b**: **array [0, n) of nat** e **c**: **array [0, m) of nat**, si formalizzi il seguente enunciato:

“Il numero degli elementi pari di **a** che non compaiono in posizione dispari di **b** è maggiore del numero di elementi di **c** che sono quadrato di un naturale.”

ESERCIZIO 5

Si verifichi la seguente tripla di Hoare (assumendo **a**: **array [0, n) of nat**):

```
{ k ∈ dom(a) ∧ k > 0 ∧ (∀i. i ∈ (0, k) ⇒ a[i - 1] < a[i]) }
  if a[k] > a[k - 1] then skip
  else a[k] := a[k - 1] + 1 fi
{ (∀i. i ∈ (0, k) ⇒ a[i - 1] < a[i]) }
```

ESERCIZIO 6

Si consideri il seguente programma annotato:

```
{ y = 0 ∧ z = 0 }
{ Inv : y = z · w + (Σi : i ∈ [0, z) . 2 · i) ∧ z ∈ [0, m] } { t: m - z }
while z < m do
  z, y := z + 1, y + w + 2 * z
endw
{ y = m · (w + m - 1) }
```

1. Scrivere le ipotesi di invarianza, di progresso e di terminazione.
2. Dimostrare l'ipotesi di invarianza.