E.T.S.I. Informática. Parcial, Diciembre 2003

Lenguajes de Programación (tercer curso)

APELLIDOS: ______________________

NOMBRE: ______________________

PUNTOS:

1	2	3	4	5	6	7
1	1	3	1	3	1	1

□ si
□ no
} deseo que se publique mi calificación

Consideremos la lógica de Hoare estándar (sin bucles).

1 Define una regla para la regla *aborta* de forma que se verifique $\vdash_{\mathcal{H}} \{P\}aborta\{Q\} \equiv [P \equiv Falso]$, y da una interpretación de esta equivalencia.

SOL Una posible regla es:

$$\frac{[P \equiv Falso]}{\{P\}aborta\{Q\}}$$

La interpretación es: Para precondiciones idénticamente falsas, es posible inferir el triplete.

2 Interpreta la equivalencia: $\vdash_{\mathcal{H}} \{P\}S\{Falso\} \equiv [P \equiv Falso]$.

SOL La interpretamos como una doble implicación:

($\Rightarrow$) Si existiera un estado satisfaciendo P, entonces el programa S terminaría para tal estado inicial y en un estado final imposible.

($\Leftarrow$) Si no existe ningún estado satisfaciendo P, entonces la siguiente frase es correcta: "partiendo de un estado satisfaciendo P el ..."

3 Prueba la equivalencia anterior indicando la técnica utilizada.

Probaré ... la doble implicación.

y para ello utilizaré las técnicas:

1. Para la implicación $[P \equiv Falso] \quad \Rightarrow \quad \vdash_{\mathcal{H}} \{P\}S\{Falso\}$ basta aplicar la regla de refinamiento y la propiedad $\{Falso\}S\{Falso\}$, que se prueba por inducción sobre la sentencia (ver página 74 del libro de texto).

2. Para la implicación $\vdash_{\mathcal{H}} \{P\}S\{Falso\} \quad \Rightarrow \quad [P \equiv Falso]$, podemos **intentar aplicar inducción sobre la derivación, y veremos donde está el problema**. Los casos base son triviales: Si la última regla aplicada es (*aborta*), o (*nada*) o la asignación, es porque la precondición es "sintácticamente" igual a $Falso$. Veamos los pasos inductivos.

   a) Si el triple $\{P\}S;T\{Falso\}$ es consecuencia de la regla de la composición con antecedente:

   $\{P\}S\{X\} \wedge \{X\}T\{Falso\}$

   $\Rightarrow\because$ Hipótesis de inducción aplicada al segundo triple

   $\{P\}S\{X\} \wedge [X \equiv Falso]$

   $\Rightarrow\because$ regla refinamiento

   $\{P\}S\{Falso\}$

   $\Rightarrow\because$ ATENCION, aquí no podemos aplicar Hipótesis de inducción ya que el triplete $\{P\}S\{Falso\}$ NO NECESARIAMENTE APARECE en el árbol de derivación de la derivación considerada para el triplete $\{P\}S;T\{Falso\}$

   Por tanto modificamos ligeramente la implicación a demostrar, y probaremos esta otra:

   $$\vdash_{\mathcal{H}} \{P\}S\{Q\} \quad \Rightarrow \quad ([Q \equiv Falso] \Rightarrow [P \equiv Falso])$$

   y lo hacemos por inducción sobre la derivación (atención que la propiedad a probar por inducción es otra implicación). Los casos base son sencillos: Si la última regla aplicada es (*aborta*), o (*nada*) es inmediato. Si es la asignación, la propiedad es consecuencia de esta otra:

   $$[Q \equiv Falso] \quad \Rightarrow \quad [x := E.Q \equiv Falso]$$

   que a su vez es consecuencia de la regla esta otra propiedad:

   $$[Q \equiv Q'] \quad \Rightarrow \quad [(x := E.Q) \equiv (x := E.Q')]$$

   (que es consecuencia de la regla de Leibnitz: página 11 del libro de texto). Veamos los pasos inductivos. Supongamos en lo que sigue que $[Q \equiv Falso]$.

a) Si el triple $\{P\}S;T\{Q\}$ es consecuencia de la regla de la composición con antecedente:

$\{P\}S\{X\} \wedge \{X\}T\{Q\}$, además de $[Q \equiv Falso]$

$\Rightarrow \because$ Hipótesis de inducción aplicada al segundo triple

$\{P\}S\{X\} \wedge [X \equiv Falso]$

$\Rightarrow \because$ HI aplicada al primer triple (ÉSTE SÍ APARECE EN EL ÁRBOL DE DERIVACIÓN).

$[P \equiv Falso]$.

b) Si el triple $\{P\}if\ b\ then\ S\ else\ T\{Q\}$ es consecuencia de la regla de la composición con antecedente:

$\{P \wedge b\}S\{Q\} \wedge \{P \wedge \neg b\}T\{Q\}$, además de $[Q \equiv Falso]$

$\Rightarrow \because$ Hipótesis de inducción dos veces

$[P \wedge b \equiv Falso] \wedge [P \wedge \neg b \equiv Falso]$

$\Rightarrow \because$ CP (tercio excluido)

$[P \equiv Falso]$

Sea el programa :

$$
\begin{array}{ll}
 & q_1, q_2, \ldots, q_6 := Q_1, Q_2, \ldots, Q_6; \\
*[\![& q_1 > q_2 \rightarrow q_1, q_2 := q_2, q_1 \\
\square & q_2 > q_3 \rightarrow q_2, q_3 := q_3, q_2 \\
\square & q_2 > q_4 \rightarrow q_2, q_4 := q_4, q_2 \\
\square & q_2 > q_5 \rightarrow q_2, q_5 := q_5, q_2 \\
\square & q_2 > q_6 \rightarrow q_2, q_6 := q_6, q_2\,]\!]
\end{array}
$$

$\boxed{\mathbf{4}}$ Prueba que el predicado $I \doteq (q_1, \ldots, q_6) \in \mathcal{P}er(Q_1, \ldots, Q_6)$ es un invariante.

$\boxed{SOL}$ *Hay que probar* $\forall i : 1 \leq i \leq 5 : [I \wedge b_i \Rightarrow S_i.I]$. Veamos una de ellas

$[I \wedge q_1 > q_2 \qquad \Rightarrow \qquad q_1, q_2 := q_2, q_1.I]$

$= \because$ Definición de I, y def. sustitución

$[(q_1, q_2, q_3, q_4, q_5, q_6) \in \mathcal{P}er(Q_1, Q_2, Q_3, Q_4, Q_5, Q_6) \wedge q_1 > q_2 \qquad \Rightarrow$

$(q_2, q_1, q_3, q_4, q_5, q_6) \in \mathcal{P}er(Q_1, Q_2, Q_3, Q_4, Q_5, Q_6)]$

$= \because$ prop. de las permutaciones

Cierto

$\boxed{\mathbf{5}}$ Prueba que $t \doteq \delta_{1,2} + \delta_{1,3} + \cdots + \delta_{1,6} + \delta_{2,3} + \cdots + \delta_{2,6}$ es un contador, donde $\delta_{x,y} = \begin{cases} 1, & \text{si } q_x > q_y \\ 0, & \text{si } q_x \leq q_y \end{cases}$

$\boxed{SOL}$ *Hay que probar* $\forall i :: [I \wedge b_i \Rightarrow wdec(S_i \mid t]$. Veamos dos de ellas; *ptle*

$wdec(q_1, q_2 := q_2, q_1 \mid t)$

$= \because$ Lema 6.46 (página 107 del libro de texto)

$(q_1, q_2 := q_2, q_1.t) < t$

$= \because$ Definición de t, sustitución

$\delta_{2,1} + \delta_{2,3} + \delta_{2,4} + \delta_{2,5} + \delta_{2,6} + \delta_{1,3} + \delta_{1,4} + \delta_{1,5} + \delta_{1,6} < \delta_{1,2} + \delta_{1,3} + \delta_{1,4} + \delta_{1,5} + \delta_{1,6} + \delta_{2,3} + \delta_{2,4} + \delta_{2,5} + \delta_{2,6}$

$= \because$ Simplificamos términos iguales

$\delta_{2,1} < \delta_{1,2}$

$\Leftarrow \because$ Definición de δ

$q_1 > q_2$

$wdec(q_2, q_4 := q_4, q_2 \mid t)$

$= \because$ Lema 6.46 (página 107 del libro de texto), definición de t, sustitución

$\delta_{1,4} + \delta_{1,3} + \delta_{1,2} + \delta_{1,5} + \delta_{1,6} + \delta_{4,3} + \delta_{4,2} + \delta_{4,5} + \delta_{4,6} < \delta_{1,2} + \delta_{1,3} + \delta_{1,4} + \delta_{1,5} + \delta_{1,6} + \delta_{2,3} + \delta_{2,4} + \delta_{2,5} + \delta_{2,6}$

$= \because$ Simplificamos términos iguales, y consideramos $q_2 > q_4$

$\delta_{4,3} + \delta_{4,5} + \delta_{4,6} < \delta_{2,3} + 1 + \delta_{2,5} + \delta_{2,6}$

Si los tres sumandos de la izquierda son nulos, la desigualdad es trivial; si por ejemplo el primero es $1 (= \delta_{4,3})$ es porque $q_4 > q_3$, y al ser $q_2 > q_4$ entonces $q_2 > q_3$ de donde $\delta_{2,3} = 1$. En definitiva, para cada sumando igual a 1 a la izquierda hay uno a la derecha que vale 1.

$\boxed{\mathbf{6}}$ Aplica el teorema de los contadores para concluir que el programa calcula el 2º menor elemento con un máximo de 9 intercambios.

$\boxed{SOL}$ *El Teorema de los contadores dice que ...* (véase Teorema 6.38:página 105). Por tanto, ya que antes del bucle se satisface I, y ya que t es un contador relativo al invariante I, el bucle termina satisfaciendo $(q_1, \ldots, q_6) \in \mathcal{P}er(Q_1, \ldots, Q_6) \wedge q_1 \leq q_2 \leq q_3, q_4, q_5, q_6$, de donde q_2 es el segundo menor elemento del conjunto $\{Q_1, \ldots, Q_6\}$.