MODELOS DE COMPUTACIÓN Y COMPLEJIDAD

Grado en Ingeniería Informática. Tecnologías Informáticas

ETS Ingeniería Informática. Universidad de Sevilla (Curso 2021-2022)

Problemas de FUNCIONES GOTO-COMPUTABLES

EJERCICIO 7.

Usando el teorema de inducción, demostrar que se verifica lo siguiente: para cada número natural k, cualquier programa `GOTO` de longitud k que carece de instrucciones condicionales, **para** sobre **todo** número natural y siempre devuelve un número menor o igual que k.

SOLUCIÓN:

Vamos a demostrar que:

$$\forall k \ \forall P \ (P \text{ programa } \texttt{GOTO} \text{ sin condicionales } \wedge |P| = k \Rightarrow \forall n \in \mathbb{N}(P(n)\downarrow \wedge P(n) \leq k)$$

Probémoslo por inducción sobre k.

$\boxed{\textbf{Caso base: } \mathbf{k = 0}}$. Hemos de probar que el resultado es cierto para $k = 0$.

Para ello, sea P un programa `GOTO` sin condicionales y de longitud 0. Entonces P es **el programa vacío** y, por tanto, para cada número natural $n \in \mathbb{N}$ se verifica que la computación $P(n)$ es de parada y, además, $P(n) = 0$. Por tanto, se verifica que $P(n) \leq 0$. Esto prueba el caso base.

$\boxed{\textbf{Paso inductivo:}}$ Sea $k \in \mathbb{N}$ tal que suponemos cierto el resultado para k; es decir, supongamos por hipótesis de inducción que para cada programa `GOTO`, P, sin condicionales y de longitud k se verifica lo siguiente: para cada número natural n la computación $P(n)$ es de parada y, además, $P(n) \leq k$. Hemos de probar que el resultado también es cierto para $k+1$; es decir, hemos de ver que si Q es un programa `GOTO` sin condicionales y de longitud $k+1$, entonces para cada número natural n la computación $Q(n)$ es de parada y, además, $Q(n) \leq k+1$.

Pues bien, para ello, sea Q un programa `GOTO` sin condicionales y de longitud $k+1$. Vamos a probar que para cada número natural n se tiene que la computación $Q(n)$ es de parada y, además, $Q(n) \leq k+1$.

En efecto: consideremos el programa Q' obtenido de Q eliminando la última instrucción I_{k+1} de Q.

$$\mathbf{Q} \equiv \boxed{\begin{array}{c} I_1 \\ \vdots \\ I_k \\ I_{k+1} \end{array}} \qquad \mathbf{Q'} \equiv \boxed{\begin{array}{c} I_1 \\ \vdots \\ I_k \end{array}} \qquad \mathbf{Q} \equiv \boxed{\begin{array}{c} \mathbf{Q'} \\ \\ I_{k+1} \end{array}}$$

Entonces Q' es un programa `GOTO` sin instrucciones condicionales y de longitud k. Por tanto, de la hipótesis de inducción resulta que para cada número natural n, la computación $Q'(n)$ será de parada y, además, $Q'(n) \leq k$.

Ahora bien, la computación $Q(n)$ se obtiene realizando la computación $Q'(n)$ (que es de parada) y, finalmente, ejecutando la instrucción I_{k+1} a la configuración de salida de $Q'(n)$. Teniendo presente que I_{k+1} **no** es una **instrucción condicional**, resultará que la computación $Q(n)$ será de parada. Por otra parte, al ejecutar I_{k+1} a la configuración de salida de $Q'(n)$, el valor de la computación $Q'(n)$ aumentará, a lo sumo, en una unidad (y eso sucedería en el caso en que I_{k+1} fuese $Y \longleftarrow Y + 1$). En consecuencia, se verificaría lo siguiente:

$$Q(n) \leq Q'(n) + 1 \overset{h.i.}{\leq} k + 1$$

Esto prueba el paso inductivo y, por tanto, concluye la prueba por inducción.

− −