Ejercicios

II.6. LA FUNCION UNIVERSAL

PRERREQUISITOS:

- ❑ Conocer las definiciones de la función Φ y de los predicados T y E.
- ❑ Conocer el programa que computa la función universal.

PROBLEMAS:

A. PARA ENTENDER CON PRECISIÓN EL SIGNIFICADO DE Φ, T Y E

1. ¿Cuál de los siguientes conjuntos es el dominio de la función universal para las funciones while-computables unarias?

   a) Σ^*
   b) $\Sigma^* \times \Sigma^*$
   c) $\{ (x,y): \varphi_x(y)\downarrow \}$
   d) $\{ x: \varphi_x(x)\downarrow \}$

2. Define, bien usando el lenguaje natural, bien basándote en los conjuntos de índices conocidos (TOT, K, etc.), los conjuntos que se citan a continuación:

   a) $\{ x: \Phi(0,x)\downarrow \}$
   b) $\{ x: \exists y\ \Phi(x,y)\downarrow \}$
   c) $\{ x: \Phi(x,x) = \varphi_x(x) \}$
   d) $\{ x: \forall y\ \Phi(y,x)\downarrow \}$
   e) $\{ x: \exists y\ \Phi(y,x)\downarrow \}$
   f) $\{ x: \exists z\ \forall y\ \Phi(x,y) = z \}$
   g) $\{ x: \forall y\ \Phi(y,x)\downarrow \}$
   h) $\{ x: \forall y\ \Phi(y,x)\uparrow \}$

3. Sea u un índice de la función universal para las funciones while-computables unarias. Demuestra entonces que $\Phi^2(u,u) \cong \varepsilon$.

4. Define, bien usando el lenguaje natural, bien basándote en los conjuntos de índices conocidos (TOT, K, etc.), las funciones y predicados que se citan a continuación:

   a) $f(x) = \begin{cases} \text{true} & \exists p T(x,x,p) \\ \text{false} & \text{c.c.} \end{cases}$
   b) $g(x) = \begin{cases} \text{true} & \forall p T(x,x,p) \\ \text{false} & \text{c.c.} \end{cases}$

   c) $\psi(x,y) \cong \begin{cases} p & T(x,y,p) \wedge \neg T(x,y,p-1) \\ \perp & \text{c.c.} \end{cases}$

   d) $\chi(x,y,k) \cong \begin{cases} \text{true} & \forall p < k \exists w E(x,y,p,w) \\ \perp & \text{c.c.} \end{cases}$

   e) $h(x) = \begin{cases} \text{true} & \exists y \exists p E(y,x,p,x) \\ \text{false} & \text{c.c.} \end{cases}$

f) $\theta(x, y) \cong \begin{cases} x+2 & \exists p(T(x,x,p) \wedge \neg T(x,x,p+y)) \\ \varphi_x(y) & \text{c.c.} \end{cases}$

B. PARA APRENDER A DISEÑAR PROGRAMAS QUE NESECITAN SIMULAR LA EJECUCIÓN DE OTROS PROGRAMAS

5. Demuestra que las siguientes funciones son while-computables:

a) $\psi: W \times \Sigma^* \times N \longrightarrow B$

$$\psi(x,y,j) \cong \begin{cases} \text{true} & \text{al ejecutarse el programa } P_x \text{ sobre el dato } y \text{ la variable } XJ \\ & \quad \text{es modificada alguna vez} \\ \perp & \text{c.c.} \end{cases}$$

b) $\chi: W \times \Sigma^* \times N \longrightarrow B$

$$\chi(x,y,j) \cong \begin{cases} \text{true} & \text{al ejecutarse el programa } P_x \text{ sobre el dato } y \text{ el valor de la} \\ & \quad \text{variable } XJ \text{ es utilizado alguna vez} \\ \perp & \text{c.c.} \end{cases}$$

c) $\theta: W \times \Sigma^* \times N \times \Sigma^* \longrightarrow B$

$$\theta(x,y,j,z) \cong \begin{cases} \text{true} & \text{al ejecutarse el programa } P_x \text{ sobre el dato } y \text{ la variable } XJ \\ & \quad \text{contiene en algún momento el valor } z \\ \perp & \text{c.c.} \end{cases}$$

d) $\tau: W \times \Sigma^* \longrightarrow N$

$$\tau(x,y) \cong \begin{cases} m & \varphi_x(y)\downarrow \wedge \text{"al ejecutarse el programa } P_x \text{ sobre el dato } y \\ & \quad m \text{ es el máximo número de iteraciones que realiza} \\ & \quad \text{un bucle individual de } P_x\text{"} \\ \perp & \text{c.c.} \end{cases}$$

e) $\xi: W \times \Sigma^* \longrightarrow N$

$$\xi(x,y) \cong \begin{cases} m & \varphi_x(y)\downarrow \wedge \text{"al ejecutarse el programa } P_x \text{ sobre el dato } y \\ & \quad \text{se realiza un total de } m \text{ iteraciones entre todos} \\ & \quad \text{los bucles de } P_x \\ \perp & \text{c.c.} \end{cases}$$

f) $\sigma: W \times \Sigma^* \times N \longrightarrow \Sigma^*$

$$\sigma(x,y,j) \cong \begin{cases} z & \varphi_x(y)\downarrow \wedge \text{"al ejecutarse el programa } P_x \text{ sobre el dato } y \\ & \quad \text{el valor máximo alcanzado por la variable } XJ \text{ es } z\text{"} \\ \perp & \text{c.c.} \end{cases}$$

g) $\eta: W \times \Sigma^* \longrightarrow N$

$$\eta(x,y) \cong \begin{cases} j & \varphi_x(y)\downarrow \wedge \text{"al ejecutarse el programa } P_x \text{ sobre el dato y} \\ & \quad \text{XJ es la variable que más veces se actualiza} \\ & \quad \text{(en caso de empate XJ es la de menor índice)"} \\ \perp & \end{cases}$$

h) $\mu: W \times \Sigma^* \times N \longrightarrow B$

$$\mu(x,y,j) \cong \begin{cases} \text{true} & \text{al ejecutarse el programa } P_x \text{ sobre el dato y la primera} \\ & \quad \text{operación realizada sobre la variable XJ es} \\ & \quad \text{asignarle un valor} \\ \text{false} & \text{al ejecutarse el programa } P_x \text{ sobre el dato y la primera} \\ & \quad \text{operación realizada sobre la variable XJ es} \\ & \quad \text{consultar su valor} \\ \perp & \text{al ejecutarse el programa } P_x \text{ sobre el dato y la variable XJ} \\ & \quad \text{no es utilizada jamás} \end{cases}$$

i) $\delta: W \times \Sigma^* \times N \longrightarrow B$

$$\delta(x,y,n) \cong \begin{cases} \text{true} & \text{si durante la ejecución de } P_x\ (y) \\ & \text{la variable XN contiene en algún momento el valor 'abb'} \\ & \text{(independientemente de si converge o no)} \\ \perp & \text{c.c.} \end{cases}$$