Hoja III.11

SOLUCIÓN DE ALGÚN APARTADO DEL EJERCICIO 1

1-e E = { (x,y): la ejecución del programa P_x sobre el dato y termina, y al final de la misma las variables X2, X3 y X4 tienen el valor ε }

Vamos a demostrar que no es decidible por reducción al absurdo, viendo que si suponemos C decidible probaríamos la computabilidad del problema de parada.

Supongamos que $C \in \sum_0$. Partiendo de cualquier programa **x** y cualquier entrada **y**, vamos a tratar de utilizar este supuesto para averiguar si $P_{\mathbf{x}}$ converge sobre **y** o no. La idea será transformar $P_{\mathbf{x}}$ en otro programa $P_{\mathbf{z}}$ de forma que, preguntando si **z** está o no en C, averigüemos de paso si $P_{\mathbf{x}}(\mathbf{y})\downarrow$.

Para obtener $P_{\mathbf{z}}$ añadiremos instrucciones a $P_{\mathbf{x}}$ por detrás, que modifiquen el contenido de las variables **X2**, **X3** y **X4** con el valor ε cuando la ejecución de $P_{\mathbf{x}}$ termine. Su forma (esquema con macros) será la siguiente:

$\mathbf{P_z}$:
```
Px
X2 := ε;
X3 := ε;
X4 := ε;
```

De esta forma podremos conocer el comportamiento del programa **x** con la entrada **y** preguntando si $(\mathbf{z},\mathbf{y}) \in C$.

- Si $\varphi_{\mathbf{x}}(\mathbf{y})\downarrow$ entonces el programa $P_{\mathbf{z}}$ además de ejecutar $P_{\mathbf{x}}$ sobre el valor **y**, asignará a **X2**, **X3** y **X4** el valor ε.
- Por el contrario, si $\varphi_{\mathbf{x}}(\mathbf{y})\uparrow$, entonces el programa $P_{\mathbf{z}}$ tampoco termina y por lo que **X2**, **X3** y **X4** no terminan con ningún valor.

Por tanto, si ahora preguntamos si $(\mathbf{z}, \mathbf{y}) \in C$ la respuesta nos permitirá averiguar si $\varphi_{\mathbf{x}}(\mathbf{y})$ convergía o no, con lo que resolveríamos el problema de parada para **x** e **y** cualesquiera. Escribimos el programa que computa la función de parada $H(\mathbf{x},\mathbf{y})$ mediante el procedimiento de construir $P_{\mathbf{z}}$ a partir de $P_{\mathbf{x}}$ y comprobar si el resultado está o no en C:

Z:= X1; -- *Primero copiamos el código de* P_x

-- A continuación añadimos las tres asignaciones vacías detrás de P_x

Z:= haz_composición(Z, haz_vacía(2));

Z:= haz_composición(Z, haz_vacía(3));

Z:= haz_composición(Z, haz_vacía(4));

-- Para terminar averiguamos si (z, y) está o no en C

X0 := C_C(Z, X2);

Así pues de esta forma indirecta, y utilizando únicamente la hipótesis de decibilidad de C, hemos construido un programa que calcula H(**x**,**y**), lo cual es absurdo, pues su incomputabilidad está demostrada. Por ello es imposible que C sea decidible.

Hoja III.11

SOLUCIÓN DE ALGÚN APARTADO DEL EJERCICIO 3

3h -A = { $x \in \Sigma^*$: $|W_x| = 1$ }.

Vamos a demostrar que no es decidible probando que existe algún conjunto reducible a A que tampoco lo es. Concretamente probaremos que $K \leq A$. Para conseguir esto necesitamos encontrar una función computable y total h: $\sum^* \to \sum^*$ que verifique que $\forall x\ (x \in K \leftrightarrow h(x) \in A)$

o, lo que es lo mismo, que para cualquier programa **x** la función **h** obtenga otro programa **h**(**x**) que verifique:

Si $\varphi_{\mathbf{x}}(\mathbf{x})\downarrow$, entonces $|W_{h(\mathbf{x})}| = 1$

Si $\varphi_{\mathbf{x}}(\mathbf{x})\uparrow$, entonces $|W_{h(\mathbf{x})}| \neq 1$

Por tanto, la función **h** que necesitamos tomará un programa **x** y lo transformará en otro programa **h**(**x**), de forma que el comportamiento del segundo está relacionado con el del primero. Esto nos sugiere que el teorema s-m-n puede ser aplicable para justificar la computabilidad de **h** sin necesidad de escribir el programa que la computa. Con ese objetivo definimos la función

$$\psi(x,y) \cong \begin{cases} \mathbf{y} & \varphi_{\mathbf{x}}(\mathbf{x})\downarrow \wedge \mathbf{y} = 100 \\ \perp & \text{c.c.} \end{cases}$$

computada por el programa:

R := Φ(X1, X1);

if X2 = 100 **then** X0 := X2; **else** X0 := ⊥; **end if;**

Dado que ψ es computable, el teorema s-m-n nos asegura que existe una función h: $\sum^* \to \sum^*$ computable y total que cumple $\forall \mathbf{x} \forall \mathbf{y}\ \psi(\mathbf{x},\mathbf{y}) \cong \varphi_{h(\mathbf{x})}(\mathbf{y})$. Comprobaremos que esta función es precisamente la que necesitamos para reducir K a A. Sea **x** un programa cualquiera:

$$\mathbf{x} \in K \Rightarrow \varphi_{\mathbf{x}}(\mathbf{x})\downarrow \Rightarrow \psi(\mathbf{x},\mathbf{y}) = \begin{cases} \mathbf{y} & \mathbf{y} = 100 \\ \perp & \mathbf{c.c.} \end{cases} \Rightarrow \varphi_{h(\mathbf{x})}(\mathbf{y}) = \begin{cases} \mathbf{y} & \mathbf{y} = 100 \\ \perp & \mathbf{c.c.} \end{cases} \Rightarrow$$

$$\Rightarrow W_{h(\mathbf{x})} = \{100\} \Rightarrow |W_{h(\mathbf{x})}| = 1 \Rightarrow h(\mathbf{x}) \in A$$

$$\mathbf{x} \notin K \Rightarrow \varphi_{\mathbf{x}}(\mathbf{x})\uparrow \Rightarrow \forall y\ \psi(\mathbf{x},\mathbf{y})\uparrow \Rightarrow \forall y\ \varphi_{h(\mathbf{x})}(\mathbf{y})\uparrow \Rightarrow W_{h(\mathbf{x})} = \varnothing \Rightarrow |W_{h(\mathbf{x})}| = 0 \Rightarrow$$

$$\Rightarrow |W_{h(\mathbf{x})}| \neq 1 \Rightarrow h(\mathbf{x}) \notin A$$

Así hemos cumplido nuestro propósito, demostrando que $K \leq A$ por medio de h, y como sabemos que $K \notin \sum 0$, podemos concluir que $A \notin \sum 0$.

(Nótese que no es la única manera de definir la función ψ. Podríamos sustituir la condición y = 100 por cualquier otra que nos asegure que la función solo va a converger para un valor de y. Además el resultado de la función podría ser cualquiera.)

Hoja III.11

SOLUCIÓN DE ALGÚN APARTADO DEL EJERCICIO 4

4-c Demostrar utilizando el teorema s-m-n que el conjunto C no es semidecidible

$$C = \{ \mathbf{x} \in \Sigma^* : | \overline{W_x} | = 3 \}$$

Para hacer la demostración utilizaremos el conjunto $\overline{K}$, que sabemos que no es semidecidible ($\overline{K} \notin \Sigma_1$).

Para empezar, definimos la función computable $\psi: \Sigma^* \times \Sigma^* \longrightarrow \Sigma^*$

$$\psi(x, y) \cong \begin{cases} 20 & \neg T(x, x, y) \wedge y \geq 3 \\ \perp & \text{c.c.} \end{cases}$$

La función ψ es computable, y el programa que lo demuestra:

if not T(X1, X1, X2) and X2 > 2 **then** X0 := 20; **end if**;

Dado que ψ es computable, el teorema s_m_n nos asegura que existe una función total y computable h: $\Sigma^* \to \Sigma^*$ tal que, $\psi(x, y) \cong \varphi_{h(x)}(y)$

Con esta función h podemos demostrar que $\overline{K} \leq C$, es decir, $x \in \overline{K} \Leftrightarrow h(x) \in C$

$$\mathbf{x} \in \overline{K} \Rightarrow \varphi_{\mathbf{x}}(\mathbf{x})\uparrow \Rightarrow \forall \mathbf{y} \in \Sigma^* \neg T(\mathbf{x},\mathbf{x},\mathbf{y}) \Rightarrow \forall \mathbf{y} \in \Sigma^* (\psi(\mathbf{x}, \mathbf{y}) = 20 \leftrightarrow \mathbf{y} > 2) \Rightarrow$$

$$\forall \mathbf{y} \in \Sigma^* (\varphi_{h(\mathbf{x})}(\mathbf{y}) = 20 \leftrightarrow \mathbf{y} > 2) \Rightarrow \left| \overline{\mathbf{W}_{\mathbf{h}(\mathbf{x})}} \right| = 3 \Rightarrow h(\mathbf{x}) \in C$$

$$\mathbf{x} \notin \overline{K} \Rightarrow \varphi_{\mathbf{x}}(\mathbf{x})\downarrow \Rightarrow \exists \mathbf{y} \in \Sigma^* T(\mathbf{x},\mathbf{x},\mathbf{y}) \Rightarrow \exists \mathbf{y} \in \Sigma^* \forall \mathbf{z}(\mathbf{z} \geq \mathbf{y} \to T(\mathbf{x},\mathbf{x},\mathbf{z})) \wedge$$

$$\forall \mathbf{z}(\mathbf{z} < \mathbf{y} \to \neg T(\mathbf{x},\mathbf{x},\mathbf{z})) \Rightarrow \exists \mathbf{y} \in \Sigma^* \forall \mathbf{z}(\mathbf{z} \geq \mathbf{y} \to \psi(\mathbf{x}, \mathbf{y})\uparrow) \wedge \forall \mathbf{z}(\mathbf{z} < \mathbf{y} \to \psi(\mathbf{x}, \mathbf{y}) = 20)$$

$$\Rightarrow \exists \mathbf{y} \in \Sigma^* \forall \mathbf{z}(\mathbf{z} \geq \mathbf{y} \to \varphi_{h(\mathbf{x})}(\mathbf{y})\uparrow) \wedge \forall \mathbf{z}(\mathbf{z} < \mathbf{y} \to \varphi_{h(\mathbf{x})}(\mathbf{y}) = 20) \Rightarrow \left| \overline{\mathbf{W}_{\mathbf{h}(\mathbf{x})}} \right| > 3 \Rightarrow$$

$$h(\mathbf{x}) \notin C$$

Una vez demostrado que $\overline{K} \leq C$, como sabemos que el conjunto $\overline{K}$ no es semidecidible ($\overline{K} \notin \Sigma_1$), entonces podemos concluir que $C \notin \Sigma_1$

4-f Demostrar utilizando el teorema s-m-n que el conjunto $\overline{FIN}$ no es semidecidible

Para hacer la demostración utilizaremos el conjunto TOT, que sabemos que no es semidecidible ($TOT \notin \Sigma_1$).

Para empezar, definimos la función computable $\psi: \Sigma^* \times \Sigma^* \dashrightarrow \Sigma^*$

$$\psi(x, y) \cong \begin{cases} \varepsilon & \forall z \le y \ \varphi_x(z)\downarrow \\ \perp & \text{c.c.} \end{cases}$$

La función ψ es computable, y el programa que lo demuestra:

```
VAR := 0;
while VAR ≤ X2 loop
    X0 := Φ(X1, VAR); VAR := VAR + 1;
end loop;
X0 := ε;
```

Dado que ψ es computable, el teorema s_m_n nos asegura que existe una función total y computable h: $\Sigma^* \to \Sigma^*$ tal que, $\psi(x, y) \cong \varphi_{h(x)}(y)$

Con esta función h podemos demostrar que $TOT \le \overline{FIN}$, es decir, $x \in TOT \Leftrightarrow h(x) \in \overline{FIN}$

$$\mathbf{x} \in TOT \Rightarrow \forall \mathbf{y} \in \Sigma^* \ \varphi_{\mathbf{x}}(\mathbf{y})\downarrow \Rightarrow \forall \mathbf{y} \in \Sigma^* \ (\forall \mathbf{z} \le \mathbf{y} \ \varphi_{\mathbf{x}}(\mathbf{z})\downarrow) \Rightarrow \forall \mathbf{y} \in \Sigma^* \ \psi(\mathbf{x}, \mathbf{y}) = \varepsilon$$

$$\Rightarrow \forall \mathbf{y} \in \Sigma^* \ \varphi_{h(\mathbf{x})}(\mathbf{y}) = \varepsilon \Rightarrow \varphi_{h(\mathbf{x})} \text{ función infinita} \Rightarrow h(\mathbf{x}) \in \overline{FIN}$$

$$\mathbf{x} \notin TOT \Rightarrow \exists \mathbf{y} \in \Sigma^* \ \varphi_{\mathbf{x}}(\mathbf{y})\uparrow \Rightarrow \exists \mathbf{y} \in \Sigma^* \neg(\forall \mathbf{z} \le \mathbf{y} \ \varphi_{\mathbf{x}}(\mathbf{z})\downarrow) \Rightarrow \exists \mathbf{y} \in \Sigma^* \ \forall k \ (k \ge \mathbf{y}$$

$$\neg(\forall \mathbf{z} \le k \ \varphi_{\mathbf{x}}(\mathbf{z})\downarrow) \ ) \Rightarrow \exists \mathbf{y} \in \Sigma^* \ \forall \mathbf{k} \ (\mathbf{k} \ge \mathbf{y} \ \psi(\mathbf{x}, k)\uparrow) \Rightarrow \exists \mathbf{y} \in \Sigma^* \ \forall k \ (k \ge \mathbf{y} \ \varphi_{h(\mathbf{x})}(k)\uparrow)$$

$$\Rightarrow \exists \mathbf{y} \in \Sigma^* \ |W_{h(x)}| < y \Rightarrow \varphi_{h(\mathbf{x})}(\mathbf{y}) \text{ finita} \Rightarrow h(\mathbf{x}) \notin \overline{FIN}$$

Una vez demostrado que $TOT \le \overline{FIN}$, como sabemos que el conjunto TOT no es semidecidible ($TOT \notin \Sigma_1$), entonces podemos concluir que $\overline{FIN} \notin \Sigma_1$