

Reducibilidad. El teorema de Rice

Ejercicio 1.— Decidir si los siguientes predicados son 1-extensionales o no:

- $P_1(p) \equiv \forall x [\llbracket p \rrbracket(x) = \llbracket p \rrbracket(x + 25)]$.
- $P_2(p) \equiv$ El programa p es de tipo lineal.
- $P_3(p) \equiv$ El programa p calcula una función polinómica (en aridad 1).
- $P_4(p) \equiv$ La función $\llbracket p \rrbracket^{(1)}$ es total, pero $\llbracket p \rrbracket^{(2)}$ no lo es ¿Es 2-extensional?
- $P_5(p) \equiv$ La función $\llbracket p \rrbracket^{(1)}$ es constante.
- $P_6(p) \equiv$ La ejecución de p siempre para en 100 pasos
- $P_7(p) \equiv$ La función $\llbracket p \rrbracket^{(1)}$ está acotada ¿Es 2-extensional?

Ejercicio 2.— Con respecto al ejercicio anterior:

1. Demostrar la indecidibilidad de los predicados del ejercicio anterior que sean extensionales.
2. Probar que los predicados P_2 y P_6 son recursivos.

Ejercicio 3.— Usando el teorema de Rice, decidir la recursividad de los siguientes conjuntos:

- $INF = \{e \in \mathbb{N} : W_e \text{ es infinito}\}$ y $FIN = \{e \in \mathbb{N} : W_e \text{ es finito}\}$
- $IMP = \{e \in \mathbb{N} : \text{ existe } p \text{ tal que } |p| \text{ es impar y } \llbracket p \rrbracket = \varphi_e\}$
- $MON = \{e \in \mathbb{N} : \varphi_e \text{ es total y } (\forall x)[\varphi_e(x) \leq \varphi_e(x + 1)]\}$
- $D = \{e \in \mathbb{N} : \varphi_e \text{ es total y } \forall x[x \in \mathcal{K} \iff \varphi_e(x) = 0]\}$
- $PRED = \{e \in \mathbb{N} : \varphi_e \text{ es un predicado}\}$

Ejercicio 4.— Consideremos el predicado

$$P(x) \equiv W_x^{(1)} \neq \emptyset$$

Probar que:

1. El conjunto $A = \{x : W_x^{(1)} \neq \emptyset\}$ es recursivamente enumerable.
2. El predicado $P(x)$ es extensional.
3. El predicado $P(x)$ no es recursivo (*Ind.:* Usar el teorema de Rice).

Ejercicio 5.— Sea $P(x)$ el predicado $P(x) \equiv$ La función $\varphi_x^{(1)}$ es una función constante. Probar que:

1. El predicado $P(x)$ es extensional.
2. El predicado $P(x)$ **no** es recursivo.

Ejercicio 6.— Sea $n \in \mathbb{N}$, y consideremos el predicado $P_n(p) \equiv \llbracket p \rrbracket^{(1)}(n) \downarrow$. Probar que:

1. El predicado P_n es extensional.
2. El predicado P_n no es recursivo.

Ejercicio 7.— Consideremos el conjunto $A = \{x \in \mathbb{N} : x \in \text{rang}(\varphi_x^{(1)})\}$. Probar que:

1. A es r.e. (*Indicación:* Utilizar la forma normal).
2. A **no** es recursivo.

Ejercicio 8.— El objetivo es demostrar que no existe un procedimiento mecánico para reconocer los programas cuyo rango (datos de salida) es $\mathbb{N}$ (es decir, que $\llbracket p \rrbracket^{(1)}$ es sobreyectiva). Para ello consideremos el siguiente problema:

- Entrada: $x \in \mathbb{N}$
- Salida: 1, si $\text{rang}(\varphi_x^{(1)}) = \mathbb{N}$. 0 en otro caso.

Probar que:

1. El predicado asociado al problema es extensional.
2. El predicado no es recursivo.

Ejercicio 9.— El objetivo es demostrar que no existe un procedimiento mecánico para reconocer si dos programas tienen el mismo dominio (sobre $\mathbb{N}$). Para ello consideremos el siguiente problema:

- Entrada: $p, q \in \text{GOTO}_P$
- Salida: 1, si $\text{dom} \llbracket p \rrbracket^{(1)} = \text{dom} \llbracket q \rrbracket^{(1)}$. 0 en otro caso.

Probar que:

1. Fijado p , el predicado asociado es 1-extensional, con índices $A = \{y \in \mathbb{N}^2 : \text{dom} \llbracket p \rrbracket = W_y^{(1)}\}$
2. El predicado asociado al problema no es recursivo.

Indicación: Probar que, en caso contrario sería recursivo el conjunto $\{e : \varphi_e^{(1)} \text{ es la función vacía}\}$.

Ejercicio 10.— Consideremos el siguiente problema:

“ Dados $p, q \in \text{GOTO}_P$, decidir si existe $x \in \mathbb{N}$ tal que $\llbracket p \rrbracket(x) \downarrow$ y $\llbracket q \rrbracket(x) \downarrow$ ”

Con conjunto asociado $A = \{(x, y) : \text{dom}(\varphi_x^{(1)}) \cap \text{dom}(\varphi_y^{(1)}) \neq \emptyset\}$. Se pide:

1. Probar que el predicado asociado es r.e.
2. Probar que $\mathbb{N}^2 \setminus A$ no es recursivo (*Indicación:* Prefijar p tal que $\llbracket p \rrbracket^{(1)}$ sea total).
3. Demostrar que el problema es indecidible.

Ejercicio 11.— Diremos que una función $g \in \mathcal{P}^{(1)}$ es extendible si existe $f \in \mathcal{R}$ tal que

$$\forall x \in \mathbb{N}, [g(x) \downarrow \implies f(x) = g(x)]$$

Probar que el problema de determinar si, dado p , $\llbracket p \rrbracket^{(1)}$ es o no extendible no es decidible.