PRÁCTICA 3:

① PROBAR NO COMPUTABLES: $f_1(x,y) = \begin{cases} 1 & \text{si } \phi_x(y)\downarrow \\ 0 & \text{c.c.} \end{cases}$

Sabemos que $HALT(x,y) = \begin{cases} 1 & \text{si } \phi_y(x)\downarrow \\ 0 & \text{c.c.} \end{cases}$ No es computable.

En particular, f_1 es "Parecido" a HALT. (Por no decir que es HALT).

Supongamos f_1 computable.

Escaneado con CamScanner

NO COMPUTABLE.

Defino f: $f(x) = f_1(x,x) = \boxed{HALT(x,x)}$

Pero habíamos dicho que f_1 era computable. ABSURDO!

Entonces f_1 NO es computable.

$$f_2(x,y) = \begin{cases} 1 & \text{si } \phi_x(y) = 0 \\ 0 & \text{c.c.} \end{cases}$$

Supongamos f_2 computable.

Defino $\tilde{f}_2$ tq. $\tilde{f}_2(x) = f_2(x,x) = \begin{cases} 1 & \text{si } \phi_x(x) = 0 \\ 0 & \text{c.c.} \end{cases}$

Ahora entonces defino la función parcial computable h:

$$h(x) = \begin{cases} 0 & \text{si } \tilde{f}_2(x) = 0 \ (\phi_x(x)\uparrow \text{ o } \phi_x(x) \neq 0) \\ \uparrow & \text{c.c.} \end{cases}$$

Entonces: como h es p. computable, existe e: $\phi_e(x) = h(x)$

$\phi_e(x)\uparrow \iff \tilde{f}_2(x) = 1 \iff \phi_x(x) = 0$

x es una variable $\longrightarrow$ fijo su valor en e. $(x = e)$.

$\boxed{\phi_e(e)\uparrow} \iff \tilde{f}_2(e) = 1 \iff \boxed{\phi_e(e) = 0}$

ABSURDO!

La contradicción surge de suponer f_2 computable.

$\Longrightarrow$ f_2 NO es computable.

$$f_3(x,y,z) = \begin{cases} 1 & \text{si } \phi_x(y)\downarrow \text{ y } \phi_x(y) > z \\ 0 & \text{c.c.} \end{cases}$$

Supongamos f_3 computable.

Defino $\tilde{f}_3$ tq. $\tilde{f}_3(x) = f_3(x,x,0) = \begin{cases} 1 & \text{si } \phi_x(x)\downarrow \text{ y } \phi_x(x) > 0 \\ 0 & \text{c.c.} \end{cases}$

Sea h_3 la función parcial computable:

$$h_3(x) = \begin{cases} x & \text{si } \tilde{f}_3(x) = 0 \ (\phi_x(x)\uparrow \text{ o } \phi_x(x) = 0) \\ \uparrow & \text{si } \tilde{f}_3(x) = 1 \end{cases}$$

h_3 es una función parcial computable $\Rightarrow$ Existe un nro. e:

$$\boxed{\phi_e(x) = h_3(x)}$$

$\phi_e(x)\downarrow \wedge \phi_e(x) = e \Longleftrightarrow f(x) = 0 \Longleftrightarrow \phi_x(x) \cancel{\vee} \phi_x(x) \; 0$

x es una variable. Fijo x en e. ($x = e$).

$\rightarrow \underline{\phi_e(e)\downarrow} \wedge \phi_e(e) = e \Longleftrightarrow f(e) = 0 \Longleftrightarrow \underline{\phi_e(e)\uparrow \vee \phi_e(e) \; 0}$

ABSURDO.

La contradicción surge de suponer f_3 computable.

$\Rightarrow f_3$ no es computable.

$\rightarrow f_4(x) = \begin{cases} 1 & \text{si } \phi_x(x)\downarrow \text{ y } \phi_x(x) \neq x \\ 0 & \text{c.c.} \end{cases}$

Supongamos f_4 computable. Defino h_4 tq.

$$h_4(x) = \begin{cases} x+1 & \text{si } f_4(x) = 0 \; (\phi_x(x)\uparrow \text{ o } \phi_x(x) = x) \\ \uparrow & \text{c.c.} \end{cases}$$

Como h_4 es p-computable $\Rightarrow$ Existe un nro. e. tq: $\phi_e(x) = h_4(x)$.

$\phi_e(x)\uparrow \vee \phi_e(x) = x \Longleftrightarrow f_4(x) = 1 \Longleftrightarrow \phi_x(x)\downarrow \wedge \phi_x(x) \neq x$

x es una variable. Fijo x en e ($x = e$).

$\underline{\phi_e(e)\uparrow \vee \phi_e(e) = e} \Longleftrightarrow f_4(e) = 1 \Longleftrightarrow \underline{\phi_e(e)\downarrow \wedge \phi_e(e) \neq e}$

ABSURDO!

$\Rightarrow f_4$ no es computable.

"PE\$O
PLOMA"

2 PROBAR NO COMPUTABLE POR REDUCCIÓN:

$\rightarrow g_1(x,y) = \begin{cases} 1 & \text{si } \phi_x(y)\uparrow \\ 0 & \text{c.c.} \end{cases}$

Escaneado con CamScanner

Supongamos g_2 computable $\Rightarrow$ Existe algún programa P que la computa.

$\rightarrow$ $g_1(x,y) = \psi_p(x,y) = \alpha(f_1(x,y))$ ↯ PERO f_1 NO ERA COMPUTABLE.

ABSURDO!

$\Rightarrow$ g_1 no computable.

$\rightarrow$ $g_2(x,y,z,w) = \begin{cases} 1 & \text{si } \phi_x(z)\downarrow \wedge \phi_y(w)\downarrow \wedge \phi_x(z) > \phi_y(w) \\ 0 & \text{c.c.} \end{cases}$

Veo que puedo intentar reducir a f_3.

Supongamos g_2 computable. Defino $\tilde{g}_2$:

$$\tilde{g}_2(x,y,z) = g_2(x,e,y,z) = \begin{cases} 1 & \text{si } \phi_x(y)\downarrow \wedge \phi_x(y) > z \\ 0 & \text{c.c.} \end{cases}$$

NRO. DE PROGRAMA QUE COMPUTA LA IDENTIDAD.

$$\lightning \rightarrow \tilde{g}_2(x,y,z) = \begin{cases} 1 & \text{si } f_3(x,y,z) \\ 0 & \text{c.c.} \end{cases}$$

$\Leftrightarrow$ $\tilde{g}_2(x,y,z) = f_3(x,y,z)$, PERO f_3 NO ES COMPUTABLE.

ABSURDO!

$\Rightarrow$ g_2 no es computable.

$\rightarrow$ $g_3(x,y,z) = \begin{cases} z+1 & \text{si } \phi_x(y)\downarrow \wedge \phi_x(y) \neq z \\ 0 & \text{c.c.} \end{cases}$

Veo la reducción a f_4.

$\rightarrow$ $\tilde{g}_3(x) = \lfloor g_3(x,x,x) \dot{-} x \rfloor$

$\Rightarrow$ $\tilde{g}_3(x) = f_4(x) \rightarrow f_4$ NO COMPUTABLE.

$\Rightarrow$ g_3 no es computable.

Papel de fibra de caña de azúcar.

Escaneado con CamScanner

③ PROBAR NO COMPUTABLE A PARTIR DE f_4:

$$g_3'(x,y,z) = \begin{cases} z & \text{si } \phi_x(y)\downarrow \wedge \phi_x(y) \neq z \\ 0 & \text{c.c.} \end{cases}$$

Supongamos g_3' computable. Definimos:

$$g_3''(x) = g_3'(x,x,x) = \begin{cases} x & \text{si } \phi_x(x)\downarrow \wedge \phi_x(x) \neq x \\ 0 & \text{c.c.} \end{cases}$$

Queremos que el caso $g_3''(0)$ esté bien definido.

En $f_4 \rightarrow f_4(0) = 0$, ya que $\phi_0(0) = 0 \leftarrow$ Programa nulo.

Por lo tanto, queremos $\underline{g_3''(0) = 0}$

~~¿Qué pasa si yo hago $\alpha(\alpha(g_3''(x)))$? Me devuelve 1? (en el caso de que sea x la respuesta).~~

<u>IDEA</u>: escribir f_4 como $\alpha(\alpha(g_3''(x)))$ y luego llegar a un absurdo.

La función α es computable, y composición de funciones computables es computable. $\Rightarrow$ $\underline{\alpha(\alpha(g_3''(x)))\ \text{computable}}$.

Pero $\alpha(\alpha(g_3''(x))) = f_4(x) \rightarrow$ NO COMPUTABLE.

¡ABSURDO!

La contradicción surge de suponer g_3 computable.

$\Rightarrow g_3''$ <u>no es computable</u>.

④ FUNCIÓN PARCIAL COMPUTABLE QUE NO SEA EXTENSIBLE.

Sea f una función parcial computable:

$$f(x) = \min_t \ STP(x,x,t)$$

Sea g una función computable extensión de f:

$$g(x) = \begin{cases} 1 & \text{si } x \in \text{Dom} f \\ 0 & \text{c.c.} \end{cases}$$

Pero g no es computable $\rightarrow f$ no es extensible.

Escaneado con CamScanner

5) PROBAR USANDO TEO. DEL PARÁMETRO QUE NO SON COMPUTABLES.

→ $g_1(x) = \begin{cases} 1 & \text{si} \quad HALT(1337, x) \\ 0 & \text{c.c.} \end{cases}$

La idea es aprovechar que $S(x,e)$ es computable, entonces si probamos que $g_1(S(x,e))$ no es computable es xq g_1 no lo es.

Sea h parcial computable:

$$h(x,y) = \begin{cases} 1 & \text{si} \quad HALT(x,x) = 1 \\ \uparrow & \text{c.c.} \end{cases}$$

Sea e el número de programa que computa h:

$$\phi_e(x,y) = h(x,y) \xrightarrow[\text{TEO. DEL PARÁMETRO}]{} \boxed{\phi_{S(x,e)}(y)}$$

→ Por definición de HALT: $HALT(1337, x) = 1 \iff \underline{\phi_x(1337)\downarrow}$

$$\Longrightarrow g_1(x) = \begin{cases} 1 & \text{si} \quad \phi_x(1337)\downarrow \\ 0 & \text{c.c.} \end{cases}$$

como $S(x,e)$ es computable y g_1 también (es nuestra suposición), analizamos $g_1(S(x,e))$.

$$g_1(S(x,e)) = 1 \iff \phi_{S(x,e)}(1337)\downarrow \iff h(x,1337) = 1$$
$$\iff \underline{HALT(x,x) = 1}$$

Pero HALT no es computable. ¡ABSURDO!

$\Longrightarrow g_1$ NO es computable.

→ $g_2(x,y,z) = \begin{cases} 1 & \text{si} \quad \phi_x(z)\downarrow \wedge \phi_y(z)\downarrow \wedge \phi_x(z) > \phi_y(z) \\ 0 & \text{c.c.} \end{cases}$

Escaneado con CamScanner

Supongamos g_2 computable.
En este ejercicio voy a intentar reducir a f_3. Para ello, fijo y en 0, siendo 0 el número de programa que computa la función nula.

Defino g_2': $g_2'(x,z) = g_2(x,0,z) = \begin{cases} 1 & \text{si } \Phi_x(z)\downarrow \wedge \Phi_x(z) > 0 \\ 0 & \text{c.c.} \end{cases}$

Quiero plantear el teo. del parámetro sobre x, ya que funciona como número de programa en g_2.
Sea h parcial computable:

$$h(x,z) = \begin{cases} 1 & \text{si } f_3(x,z,0) = 1 \\ \uparrow & \text{c.c.} \end{cases}$$

y sea e el número que la computa. $\Rightarrow \Phi_e(x,z) = h(x,z)$
Por teo. del parámetro. $\boxed{\Phi_{S(x,e)}(z) = \Phi_e(x,z)}$

Entonces:

"MISMO PROGRAMA"

$$g_2'(S(x,e),z) = 1 \iff \Phi_{S(x,e)}(z)\downarrow \wedge \Phi_{S(x,e)}(z) > 0 \iff h(x,z) = 1$$
$$\iff f_3(x,z,0) = 1$$

Pero f_3 NO es computable. ¡ABSURDO!
$\Rightarrow g_2$ NO es "

$$g_3(x) = \begin{cases} 13 & \text{si } \Phi_x \text{ es la constante } 7 \\ 0 & \text{c.c.} \end{cases}$$

"$(\forall t)(\Phi_x(t) = 7)$"

No veo ninguna f que sea notable, así que voy a intentar reducir a HALT.
Supongamos g_3 computable.
Defino una función h parcial computable:

$$h(x,y)=\begin{cases} 7 & \text{si } HALT(x,x)=1 \\ \uparrow & \text{c.c.} \end{cases}$$

Sea e el Nro. de Prog. que computa a h: $\phi_e(x,y) = h(x,y)$.

Por teo. del Parámetro $\longrightarrow$ $\phi_{S(x,e)}(y) = \phi_e(x,y)$

$g_3(S(x,e)) = 13 \Longleftrightarrow (\forall t)\ \phi_{S(x,e)}(t) = 7 \Longleftrightarrow h(x,t) = 7 \Longleftrightarrow HALT(x,x)=1$

ABSURDO.

$\Longrightarrow g_3$ NO es computable.

$$g_4(x,y)=\begin{cases} 1 & \text{si } \phi_x(y)\downarrow \wedge \phi_y(x)\downarrow \wedge \phi_x(y) \neq \phi_y(x) \\ 0 & \text{c.c.} \end{cases}$$

Voy a intentar reducir a f_4.

Supongamos g_4 computable: Defino una función h parcial computable:

$$h(x,y)=\begin{cases} 1 & \text{si } f_4(x)=1 \\ \uparrow & \text{c.c.} \end{cases}$$

Sea e el Nro. de Programa que computa a h: $\phi_e(x,y) = h(x,y)$

Por teo. del Parámetro: $\phi_{S(x,e)}(y) = \phi_e(x,y)$.

Veo que se me complica la situación xq. en g_4, x funciona como parámetro de programa y número de prog.

Fijo Y en 0. ($y=0$).

$$g_4'(x) = g_4(x,0) = \begin{cases} 1 & \text{si } \phi_x(0)\downarrow \wedge \phi_x(0) \neq 0 \\ 0 & \text{c.c.} \end{cases}$$

CONSULTAR.

$g_4(S(x,e)) = 1 \Longleftrightarrow \phi_{S(x,e)}(0)\downarrow \wedge \phi_{S(x,0)}(0) \neq 0 \Longleftrightarrow h(x,y) = 1$

$\Longleftrightarrow f_4(x) = 1$

Pero f_4 NO es computable. ¡ABSURDO!

$\Longrightarrow g_4$ NO es computable. ☒

⑥ DEMOSTRAR QUE EXISTE UN PROGRAMA P TQ. $\psi_P(x)\downarrow$ SII $x = \#P$.

Me suena que es con teo. de la recursión.

Sea g parcial computable. Existe un e tq:

$$\phi_e(x) = g(e,x) = \begin{cases} 1 & \text{si } e = x \\ \uparrow & \text{c.c.} \end{cases}$$

¿LISTO?

⑦ PROBAR NO COMPU. CON TEO. DE LA RECURSIÓN.

$$h_1(x) = \begin{cases} 1 & \text{si } x \in \mathrm{Im}\,\phi_x \quad (\Leftarrow \phi_x(z) = x \text{ PARA ALGÚN } z) \\ 0 & \text{c.c.} \end{cases}$$

Supongamos h_1 computable. Defino $\tilde{h}_1$ parcial computable:

$$\tilde{h}_1(x,y) = \begin{cases} x & \text{si } h_1(x) = 0 \\ \uparrow & \text{c.c.} \end{cases}$$

Por teo. de la recursión, $\phi_e(y) = \underline{\tilde{h}_1(e,y)}$

$$\rightarrow \tilde{h}_1(e,y) = \begin{cases} e & \text{si } h_1(e) = 0 \quad (e \notin \mathrm{Im}\,\phi_e) \\ \uparrow & \text{c.c.} \end{cases}$$

- $h_1(e) = 1 \Longrightarrow e \in \mathrm{Im}\,\phi_e \Longrightarrow \phi_e(y)\uparrow \;\forall y \Longrightarrow \mathrm{Im}\,\phi_e = \emptyset$
- $h_1(e) = 0 \Longrightarrow e \notin \mathrm{Im}\,\phi_e \Longrightarrow \underbrace{\phi_e(y) = e \;\forall y}_{\text{¡ABSURDO!}}$

La contradicción surge de suponer h_1 computable.

$\Longrightarrow$ <u>h_1 NO es computable.</u>

$$h_2(x,y) = \begin{cases} 1 & \text{si } \phi_x(y)\downarrow \text{ y } \phi_x(y) > x \\ 0 & \text{c.c.} \end{cases}$$

Sup. h_2 computable. Defino g_2 parcial computable:

$$g_2(x,y) = \begin{cases} x+1 & \text{si } h_2(x,y) = 0 \\ \uparrow & \text{c.c.} \end{cases}$$

Por teo. de la recursión, $\exists e : \phi_e(y) = g_2(e,y)$

$$\rightarrow g_2(e,y) = \begin{cases} e+1 & \text{si } h_2(e,y)=0 \ (\phi_e(y)\uparrow \text{ ó } \phi_e(y)\leq e) \\ \uparrow & \text{c.c.} \end{cases}$$

- $h_2(e,y)=1 \Longrightarrow \phi_e(y)\downarrow \wedge \phi_e(y)>e \Longrightarrow g_2(e,y)\uparrow = \underline{\phi_e(y)\uparrow} \ \forall y.$
- $h_2(e,y)=0 \Longrightarrow \phi_e(y)\uparrow \vee \phi_e(y)\leq e \Longrightarrow \underline{\phi_e(y) = e+1} \ \forall y$

$$\Longrightarrow \phi_e(y)\downarrow \wedge \phi_e(y)>e \ \forall y$$

¡ABSURDO!

La contradicción surge de suponer h_2 computable.

$\Longrightarrow h_2$ <u>no es computable.</u>

$$\rightarrow h_3(x) = \begin{cases} 1 & \text{si } \mathrm{Im}\,\phi_x \text{ es infinita} \\ 0 & \text{c.c.} \end{cases}$$

Sup. h_3 computable. Defino g_3 parcial computable:

$$g_3(x,y) = \begin{cases} y & \text{si } h_3(x)=0 \\ \uparrow & \text{si } h_3(x)=1 \end{cases}$$

Por el teo. de la recursión, $\exists e \in \mathbb{N}$: $\underline{\phi_e(y) = g_3(e,y)}$.

$$\rightarrow g_3(e,y) = \begin{cases} y & \text{si } h_3(e)=0 \ (\mathrm{Im}\,\phi_e \text{ es finita}) \\ \uparrow & \text{c.c.} \end{cases}$$

- $h_3(e)=1 \Longrightarrow \mathrm{Im}\,\phi_e$ es infinita $\Longrightarrow \phi_e(y)\uparrow \ \forall y \Longrightarrow \underline{\mathrm{Im}\,\phi_e = \emptyset}$
  NO ES INF.
- $h_3(e)=0 \Longrightarrow \mathrm{Im}\,\phi_e$ es finita $\Longrightarrow \phi_e(y)=y \ \forall y \Longrightarrow \underline{\mathrm{Im}\,\phi_e = \mathbb{N}}$
  ES INFINITA

¡ABSURDO!

La contra. surge de suponer h_3 compu. $\Longrightarrow h_3$ <u>no es computable</u>

$$\rightarrow h_4(x) = \begin{cases} 1 & \text{si } |\mathrm{Dom}\,\phi_x| = x \\ 0 & \text{c.c.} \end{cases}$$

Supongamos h_4 computable. Defino g_4 parcial computable:

$$g_4(x,y) = \begin{cases} 1 & \text{si } h_4(x)=0 \wedge y\leq x \\ \uparrow & \text{si } h_4(x)=1 \end{cases}$$

SIGUE.

g_4 es parcial computable $\Longrightarrow$ existe e, nro. de programa tq:

Por teo. de la recursión $\longrightarrow$ $\underline{\underline{\Phi_e(y) = g_4(e,y)}}$.

$$g_4(e,y) = \begin{cases} 1 & \text{si } h_4(e) = 0 \wedge y \leq e \\ \uparrow & \text{c.c.} \end{cases}$$

Entonces:

- $h_4(e) = 0 \Longrightarrow |\text{Dom } \Phi_e| \neq e \Longrightarrow g_4(e,y) = \Phi_e(y) = 1 \ (\forall y)_{\leq e}$

$\Longrightarrow |\text{Dom } \Phi_e| = e$

• $h_4(e) = 1 \Rightarrow |\text{Dom } \phi_e| = e \Rightarrow g_4(e, y) = \phi_e(y)\uparrow \ \forall y \rightarrow \text{Dom } \phi_e = \emptyset$

¡ABSURDO!

La contradicción surge de suponer h_4 computable.

$\Rightarrow$ h_4 no es computable.

8

A $C = \{x : \phi_x \in \bigcap_{i=1}^{k} C_i\}$

3 Me mataste no soy de por acá.

9

(1 $\Rightarrow$ 2). P y Q computan la misma función. D es un conjunto de índices $\Rightarrow$ #P, #Q $\in$ D.

(2 $\Rightarrow$ 1). Como sucede que #P, #Q $\in$ D, y #P y #Q son números de programas, podemos decir que D es un conj. de índices.

10 DEMOSTRAR NO COMPUTABLES POR RICE.

$$g_1(x) = \begin{cases} 1 & \text{si Dom } \phi_x = \emptyset \\ 0 & \text{c.c.} \end{cases}$$

Suponemos g_1 computable, con g_1 función característica.

Tenemos que definir el conj. de índices de g_1 y la clase de funciones al que pertenece.

$C_{g_1} = \{x \mid \text{Dom } \phi_x = \emptyset\}$ conjunto de índices.

$\mathcal{C} = \{g \mid \text{Dom } g = \emptyset\}$

Quiero aplicar Rice sobre C_{g_1}. Para ello, quiero probar que es un conjunto no trivial.

• $0 \notin C_{g_1}$.

• $e \in C_{g_1}$, con e número de programa que se indefine para toda entrada.

Escaneado con CamScanner

$\Longrightarrow \quad \emptyset \neq C_{g_1} \neq \mathbb{N}$.

Por Rice, C_{g_1} no es computable. $\Longrightarrow g_1$ <u>no es computable</u>.

$$g_2(x,y) = \begin{cases} 1 & \text{si } \operatorname{Dom} \phi_x \cup \operatorname{Dom} \phi_y = \mathbb{N} \\ 0 & \text{c.c.} \end{cases}$$

Supongamos g_2 computable.

$$g_2'(x) = g_2(x,x) = \begin{cases} 1 & \text{si } \operatorname{Dom} \phi_x = \mathbb{N} \\ 0 & \text{c.c.} \end{cases}$$

Defino el conjunto de índices de la función característica g_2':

$$C_{g_2'} = \{ x \mid \operatorname{Dom} \phi_x = \mathbb{N} \}$$

cuya clase de funciones es:

$$\mathcal{C} = \{ g \mid \operatorname{Dom} g = \mathbb{N} \}$$

Como quiero aplicar el teo. de Rice, tengo que probar que $C_{g_2'}$ es un conjunto no trivial.

- $0 \in C_{g_2'}$

Sea e el número de programa tq $\operatorname{Dom} \phi_x = \emptyset$.

- $e \notin C_{g_2'}$.

$\Longrightarrow \quad \emptyset \neq C_{g_2'} \neq \mathbb{N}$

Por Rice, $C_{g_2'}$ no es computable. $\Longrightarrow g_1$ <u>no es computable</u>.

$$g_3(x,y) = \begin{cases} 1 & \text{si } y \in \operatorname{Dom} \phi_x \\ 0 & \text{c.c.} \end{cases}$$

Supongamos g_3 computable. Defino g_3' tq:

$$g_3'(x) = \begin{cases} 1 & \text{si } g_3(x,38) = 1 \\ 0 & \text{c.c.} \end{cases}$$

Escaneado con CamScanner

Como g_3 es computable, también lo es.

Defino el conj. de índices de g_3':

$$C_{g_3'} = \{ x \mid \phi_x(38)\downarrow \}$$

que se corresponde con la clase de funciones $\mathcal{C}$ tq:

$$\mathcal{C} = \{ g \mid 38 \in \mathrm{Dom}\, g \}$$

Probamos que $C_{g_3'}$ no es un conj. trivial:

- $0 \in C_{g_3'}$

Sea e el nro. de programa tq. $\phi_e(38)\uparrow$:

- $e \notin C_{g_3'}$

$\Longrightarrow \emptyset \neq C_{g_3'} \neq \mathbb{N}$.

Por Rice, $C_{g_3'}$ no es computable. $\Longrightarrow$ g_3' no es computable.

$$g_4(x,y) = \begin{cases} \phi_x(\phi_y(72)) & \text{si } \phi_x \circ \phi_y \text{ es total} \\ 73 & \text{c.c.} \end{cases}$$

Supongamos g_4 computable.

Sea id el número de programa que computa la identidad.

Definimos g_4':

$$g_4'(x) = \begin{cases} 1 & \text{si } g_4(x, id) = \phi_x(72) \\ 0 & \text{c.c} \end{cases}$$

como id es la función identidad, ya sabemos que es total computable (es p.r.). Tenemos que ver que $\phi_x \circ \phi_{id}$ es total $\Longleftrightarrow$ ϕ_x es total. $\leftarrow$ LO QUE QUIERO VER.

Defino el conjunto de índices de g_4':

$$C_{g_4'} = \{ x \mid \phi_x \text{ es TOTAL} \}$$

que se corresponde con la clase de funciones $\mathcal{C}$ tq:

$$\mathcal{C} = \{ g \mid g \text{ es TOTAL} \}$$

Escaneado con CamScanner

Veamos que $C_{g_4'}$ no es un conj. trivial.

- $0 \in C_{g_4'}$
- $e \notin C_{g_4'}$

$\Longrightarrow \emptyset \neq C_{g_4'} \neq \mathbb{N}$

Con e nro. de programa tq. $\forall y \ \phi_e(y)\uparrow$.

Por Rice, $C_{g_4'}$ no es computable. $\Longrightarrow g_4$ no es computable.

11

A) Sale x def. de c.e. Está en la teórica.

B) K es c.e. pero no es computable. Tampoco K^c.

C) K es c.e. pero K^c no. Sino, K sería computable.

12

$C_1 = \{ x : \phi_x(x) = 2x \}$

Supongamos C_1 computable.

// No sé si se puede usar Rice en este...

Defino la función característica de C_1:

$$g_1(x) = \begin{cases} 1 & \text{si } \phi_x(x) = 2x \\ 0 & \text{c.c.} \end{cases}$$

Defino la función parcial computable h_1:

$$h_1(x,y) = \begin{cases} 2x & \text{si } g_1(x) = 0 \\ \uparrow & \text{c.c.} \end{cases}$$

Por teo. de la recursión, existe un $e \in \mathbb{N}$: $\phi_e(y) = h_1(e,y)$.

$$h_1(e,y) = \begin{cases} 2e & \text{si } g_1(e) = 0 \\ \uparrow & \text{c.c.} \end{cases}$$

- $g_1(e) = 1 \Longrightarrow \phi_e(e) = 2e \Longrightarrow h_1(e,y)\uparrow = \phi_e(y)\uparrow \ \forall y$

Si se indefine $\forall y$, también en $y = e$.

- $g_1(e) = 0 \Longrightarrow \phi_e(e) \neq 2e \Longrightarrow h_1(e,y) = \phi_e(y) = 2e \ \forall y$

ABSURDO. $\Longrightarrow C_1$ NO ES COMPUTABLE.

Papel de fibra de caña de azúcar.

Escaneado con CamScanner

Sea P un Programa tq:

$Z_1 \leftarrow \Phi_{x_1}(x_1)$

[L] IF $Z_1 \neq 2x_1$ GOTO L

$\longrightarrow \psi_P(x) = \begin{cases} 0 & \text{si } x \in C_1 \\ \uparrow & \text{c.c.} \end{cases}$

C_1 es c.e.

→ $C_2 = \{x : 1 \in Dom\ \Phi_x\}$

Supongamos C_2 computable.

Sea: $\mathcal{C} = \{g : 1 \in Dom\ g\}$ la clase de funciones a la que pertenece el conj. de índices C_2.

Quiero aplicar Rice:

- $0 \in C_2$
- $e \notin C_2$, con $e: \Phi_e(1)\uparrow$.

$\Rightarrow \emptyset \neq C_2 \neq \mathbb{N}$.

Por Rice, C_2 no es computable.

Sea P un Programa tq:

$Z_1 \leftarrow \Phi_{x_1}(1)$ $\longrightarrow \psi_P(x) = \begin{cases} 0 & \text{si } 1 \in Dom\ \Phi_x \\ \uparrow & \text{c.c.} \end{cases}$

C_2 es c.e.

→ $C_3 = \{x : Dom\ \Phi_x \subseteq \{0, \dots, x\}\}$

Supongamos C_3 computable, y sea g su función característica:

$$g(x) = \begin{cases} 1 & \text{si } Dom\ \Phi_x \subseteq \{0, \dots, x\} \\ 0 & \text{c.c.} \end{cases}$$

Defino h_3 parcial computable:

$$h_3(x, y) = \begin{cases} 1 & \text{si } g(x) = 0 \wedge y \leq x \\ \uparrow & \text{si } g(x) = 1 \vee y > x. \end{cases}$$

Por teo. de la recursión $\longrightarrow \Phi_e(y) = h_3(e, y)$.

Escaneado con CamScanner

CONSULTAR.

$$h_3(e,y) = \begin{cases} 1 & \text{si} \quad g_3(e)=0 \wedge y \leq e \\ \uparrow & \text{c.c.} \end{cases}$$

→ $C_4 = \{ \langle x,y \rangle : \forall z \in (\mathrm{Dom}\, \phi_x \cap \mathrm{Dom}\, \phi_y) \ \ \phi_x(z) < \phi_y(z) \}$

Supongamos C_4 computable.

$$C_4' = \{ x : \langle x, id_1 \rangle \in C_4 \}$$

con id_1 = número de programa tq: $\forall z \quad \phi_{id_1}(z) = z+1$.

Defino la clase de funciones:

$$\mathcal{C} = \{ g : \forall x \ \ g(x) < x+1 \}$$

- $0 \in C_4'$
- $id_1 \notin C_4'$

$\Longrightarrow \emptyset \neq C_4' \neq \mathbb{N}$.

Por Rice, C_4 no es computable.

→ $C_5 = \{ \langle x,y,t,i \rangle : \exists \sigma \ \ SNAP(x,y,t) = \langle i, \sigma \rangle \}$

CONSULTAR.

Yo creo que siempre va a existir ese σ, ya que le estamos dando un tiempo e instrucción por parámetro.
No importa que el programa se indefina para la entrada x, de = forma va a estar haciendo algo.

→ $C_6 = \{ \langle x,y,i \rangle : \exists \sigma, t \ \ SNAP(x,y,t) = \langle i, \sigma \rangle \}$

~~CONSULTAR.~~ NO P.R. ⟹ NO COMP.

No estoy seguro xq capaz le paso una instrucción en un tiempo que no se ejecuta.

↳ En ese caso no ∈ al conjunto. C_5 es p.r.

Papel de fibra de caña de azúcar.

Escaneado con CamScanner