1 Comentario preliminar

Usamos φ para denotar un elemento arbitrario del alfabeto $\{\#, *\}$. Usamos la notación $f \sim (n, m, \varphi)$ para decir "f es de tipo (n, m, φ)". En el paradigma de Turing, usamos $\mathbb{D}$ en vez de Des para denotar el conjunto de descripciones instanáneas.

Muchos de estos problemas están resueltos en inglés. Esto es una desgracia porque es poco democrático. La razón es que los ejercicios fueron hechos con Vim + VimTex, y los atajos de teclado de Vim y LaTex son muy imprácticos si el teclado está configurado en castellano. Si hubiera escrito estos problemas con la intención de compartirlos, habría resuelto esta incompatibilidad. Lo cierto es que los escribí sin anticipar que los haría públicos.

No asuma que los ejercicios son correctos. Allí donde lo sean, no asuma que la solución dada es la más elegante. Fueron hechos por un estudiante, nada más. Estimo que otros estudiantes podrán sacar provecho de ellos.

2 Guia 2 : Infinituplas

Problem 1 *Demostrá por inducción: Para todo $x \in \mathbb{N}$ hay una infinitupla única $\overrightarrow{s} \in \omega^{[\mathbb{N}]}$ tal que*

$$x = \prod_{i=1}^{\infty} pr(i)^{s_i}$$

El caso base es trivial. Supongamos que la afirmación se cumple para todo $n \leq k$. El teorema fundamental de la aritmética asegura que $k + 1 = p_1 \cdot \ldots \cdot p_m$, donde p_i es primo. Supongamos que la factorización anterior está ordenada (es decir, $p_{j+1} > p_j$ para todo $j \in [1, m]$). Entonces $k + 1 = p_m \cdot q$ con $q = p_1 \cdot \ldots \cdot p_{m-1}$.

Subdemostración. Demostraremos $k + 1 = p_m \cdot q \Rightarrow q \leq k$. Supongamos que la premisa se cumple y la consecuencia no. Dado que $q > k$, tenemos $q \cdot x > k + 1$ para todo $x > 1$. Entonces $q \cdot x > k + 1$ para todo x que sea primo. Entonces $q \cdot p_m \neq k + 1$, lo cual es una contradicción. Entonces, si $k + 1 = q \cdot p_m$, tenemos $q \leq k$. ■

Dado que $q \leq k$, mediante la hipótesis inductiva, q toma la forma productoria del teorema anterior. Entonces $k+1 = q \cdot pr(j)$ donde $pr(j) = p_m$. Entonces el teorema se cumple para todo $n \in \mathbb{N}$.

Problem 2 *Prove that $S = \{(x, @^x) : x \equiv 0 \mod 2\}$ is $\{@\}$-effectively enumerable.*

Prueba corta. Se hace demostrando que S es Σ-efectivamente computable. Esto es fácil: se da un procedimiento que verifica, dado un input (x, α), si x es par y si $\alpha = @^x$. Demostrando que es Σ-efectivamente computable, demostramos que es Σ-enumerable.

Prueba larga. Damos explícitamente el programa que enumera a S.

Lo hacemos notando que $*^{\leq}$ es Σ-efectivamente computable bajo cualquier orden $\leq$ de Σ, ya que la función es bastante algorítmica por naturaleza. (Si no se convence, escriba el procedimiento efectivo de esta función.) Sea $\mathbb{P}_{\text{número a palabra}}$ el procedimiento que, dado un valor $x \in \omega$, calcula $*^{\leq}(x)$. Entonces definimos $\mathbb{P}$ como el procedimiento que tomando un valor $x \in \omega$ hace lo siguiente:

(0) Computa $(x)_1, (x)_2$.
(1) Usa $\mathbb{P}_{\text{número a palabra}}$ para calcular $*^{\leq}(x_2)$ y lo guarda en α.
(2) Comprueba si $(x)_1$ es par; si lo es continúa, si no lo es va a *(5)*
(3) Comprueba si $\alpha = @^{(x)_1}$. Si lo es, continúa, si no lo es va a *(5)*
(4) Devuelve (x_1, α) y termina.
(5) Devuelve $(0, \varepsilon)$ y termina.

Ejemplo. Considera $\mathbb{P}(6)$. En *(0)*, esto asigna $x_1 = 1$, $x_2 = 1$. La primera palabra en Σ es $@$. El programa encuentra la tupla $(1, @$. Como 1 es impar va a *(5)* y devuelve $(0, \epsilon)$.

Considera la tupla $(2, @@@@)$. Sabemos que existe algún $x \in \omega$ tal que $\mathbb{P}(x) = (2, @@@@)$ (aquí uso la notación matemática de manera flexible). Dado que $@@@@$ es la cuarta palabra en Σ, x es tal que $x = \langle 2, 4, (x)_3, (x)_4, \ldots\ldots \rangle$. Por ejemplo, $2^2 + 3^4 = 85$ o $2^2 + 3^4 + 5^{17} = 762939453210$ satisfarán esto.

3 Guia 4

Problem 3 *Diga verdadero o falso.*

(1) Si $d \in \mathbb{D}$ entonces d es una 3-UPLA.

Recuerde que una descripción instantánea es una palabra de la forma $\alpha q \beta$, con $q \in Q$ y $[\beta]_{|\beta|} \neq B$. La proposición es falsa.

(2) Si $d \in \mathbb{D}$, entonces $St(d) = d \cap Q$.

Ya dijimos que d es una palabra. La operación binaria $\cap$ está definida cuando los términos operantes son conjuntos. La expresión $d \cap Q$ carece de significado.

(3) $\forall d \in \mathbb{D} : d \vdash d$.

Evidentemente falso. Si fuera verdadero ninguna máquina podría realizar ninguna operación. Si quiere probarlo formalmente, halle un contraejemplo (esto es trivial).

(4) Si $\alpha p \beta \nvdash d$ para toda $d \in \mathbb{D}$, entonces $\delta(p, [\beta B]_1) = \emptyset$.

La proposición es falsa. Considere como contraejemplo la descripción instantánea $\varepsilon p \beta$, es decir una descripción que indica que el cabezal está al inicio de la cinta, y asuma $\beta = a\beta'$ con $a \in \Sigma$. Si $\delta(p, a) = (q, w, L)$ para cualesquiera $q \in Q, w \in \Sigma$, tenemos que $\varepsilon q \beta \nvdash d$ para todo d, pues el símbolo $\vdash$ con instrucciones que van hacia la izquierda (L) no está definidio cuando la palabra la izquierda del cabezal es ε.

(5) Si $(p, a, L) \in \delta(p, a)$, entonces $pa \nvdash d$ para toda $d \in \mathbb{D}$.

Verdadero por el mismo principio que nos dio el contraejemplo del caso anterior. El símbolo $\vdash$ no está definido si la dirección es L (izquierda) y el cabezal está al comienzo de la cinta.

(6) Dadas $d, d' \in \mathbb{D}$, si $d \vdash d'$ entonces $|d| \leq |d'| + 1$.

Es fácil ver que esto es falso con un contraejemplo.

Problem 4 *Find a Turing machine that accepts by reach of final state the language* $\mathscr{L} = \left\{w \in \{a, b\}^* : |w|_a = 2|w|_b\right\}$.

4 Guías 5 y 6

Problem 5 *Encuentre funciones que definan recursivamente a* $R = \lambda t \left[2^t\right]$.

Seamos claros con los tipos. Pues $R \sim (1, 0, \#)$ y la recursión se hará claramente sobre una variable numérica, debemos encontrar $f \sim (0, 0, \#), g \sim (2, 0, \#)$ tales que $R(0) = f$ y $R(t+1) = g\,(R(t), t)$. Evidentemente $R(0) = 1 \Rightarrow f = C_1^{0,0}$. Puesto que $R(t+1) = 2^{t+1} = 2^t \times 2$ tenemos que $g = \lambda x\,[2 \cdot x] \circ \left[p_1^{2,0}\right]$.

Observación. Aunque g involucra, a fines prácticos, una sola variable numérica, la definimos de modo tal que su dominio es ω^2. Esto es para respetar los tipos exigidos por la recursión primitiva.

Problem 6 *Lo mismo para* $R = \lambda t\,[t!]$.

Los tipos de f y g serán igual que en el ejercicio anterior. Pero como esta recursión sí involucra al factor t, el segundo argumento de g ya no será superfluo. Es fácil ver que $f = C_1^{0,0}$. Dado que $R(t+1) = t!(t+1)$ tenemos que

$$g = \lambda xy\,[x \cdot y] \circ \left[p_1^{2,0}, Suc \circ p_2^{2,0}\right]$$

(Recuerde que para recursión de función numérica sobre variable numérica, requerimos $R(t+1, \vec{x}, \vec{\alpha}) = g(R(t), t, \vec{x}, \vec{\alpha})$).

Problem 7 *Lo mismo para* $R = \lambda t x_1 \alpha_1 \alpha_2\,[t \cdot x_1]$.

Seamos rigurosos con los dominios y observemos que la f y la g buscadas son tal que $f \sim (1, 2, \#), g \sim (3, 2, \#)$. Es evidente entonces que $f = C_0^{1,2}$. Pues $R(t+1, x, \alpha, \beta) = t \cdot x_1$ tenemos simplemente que

$$g = \lambda xyz\alpha\beta\,[x \cdot y] \circ \left[p_2^{3,2}, p_3^{3,2}\right]$$

Problem 8 *Sea* $\Sigma = \{@, !, ?\}$. *Encuentre* f, g *tales que* $R(f, g) = \lambda t x_1 \left[!@!!!!?^t\right]$.

Pues hacemos recursión sobre una variable numérica de $R(f, g) \sim (2, 0, *)$, requerimos que $f \sim (1, 0, *), g \sim (2, 1, *)$. Observe que $R(f, g)(0, x_1) = !@!!!!\varepsilon$. Luego $f = C_{!@!!!!}^{1,0}$. Observe que $R(t+1, x) = !@!!!!?^t?^{t+1} = R(t)?^{t+1}$. Luego

$$g = \lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{2,1}, \lambda\,[\alpha^x] \circ \left[C_?^{2,1}, p_1^{2,1}\right]\right]$$

Es fácil observar, reemplazando las variables, que

$$\lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{2,1}, \lambda x\alpha\,[\alpha^x] \circ \left[Suc \circ p_1^{2,1}, C_?^{2,1},\right]\right](t, x, R(t)) = R(t)?^{t+1}$$

Problem 9 *Si* $\Sigma = \{@, !, ?\}$*, encuentre* $f, \mathcal{G}$ *tales que* $R(f, \mathcal{G}) = \lambda\alpha_1\alpha\,[|\alpha|_1 + |\alpha|_@]$

Otra vez seamos explícitos con los dominios. Pues $R(f, g) \sim (0, 2, \#)$ tenemos $f \sim (0, 1, \#), g \sim (1, 2, \#)$.

Es evidente que $R(f, \mathcal{G}), \alpha_1, \epsilon) = |\alpha|$. Luego $f = \lambda\alpha\,[|\alpha|]$. Veamos que

$$R(\alpha_1, \alpha a) = \begin{cases} R(\alpha_1, \alpha) & a \neq @ \\ R(\alpha_1, \alpha) + 1 & a = @ \end{cases}$$

Tomando la familia indexada de funciones $\mathcal{G} = \left\{(!, p_1^{1,2}), (?, p_1^{1,2}), (@, Suc \circ p_1^{1,2})\right\}$, obtenemos efectivamente que $R(f, \mathcal{G})$.

Problem 10 *Encuentre* $f, \mathcal{G}$ *tales que* $R(f, \mathcal{G}) = \lambda\alpha_1\alpha\,[\alpha_1\alpha]$

Evidentemente, $f = \lambda\alpha\,[\alpha]$. Pues $R(f, \mathcal{G})(\alpha_1, \alpha a) = \alpha_1\alpha a$, observamos que $\mathcal{G} = \left\{a \in \Sigma : (a, d_a \circ p_3^{0,3})\right\}$. Entonces, es evidente que

$$\begin{aligned} R(f, \mathcal{G})(\alpha_1, \alpha a) &= \mathcal{G}_a\,(\alpha_1, \alpha, R(f, \mathcal{G})(\alpha_1, \alpha)) \\ &= (d_a \circ p_3^{3,0)})\,(\alpha_1, \alpha, R(f, \mathcal{G})(\alpha_1, \alpha)) \\ &= R(f, \mathcal{G})(\alpha_1, \alpha)a \end{aligned}$$

Por ejemplo, $R(f, \mathcal{G})(!?!, ?@) = R(f, \mathcal{G})(!?!, ?)@ = (R(f, \mathcal{G})(!?!, \varepsilon)?)@ = ((!?!)?)@ = !?!?@$.

tal como deseábamos.

Problem 11 *Demuestre que* $\mathcal{F} = \lambda xy\alpha\beta\,[\alpha^x = \beta]$ *es* Σ*-p.r.*

Cuidado con los dominios: $\mathcal{D}_{\mathcal{F}} = \omega^2 \times \Sigma^{*2}$, aunque la variable y de la expresión lambda no sea utilizada. Es fácil ver que

$$\mathcal{F} = \lambda\alpha\beta\,[\alpha = \beta] \circ \left[\lambda x\alpha\,[\alpha^x] \circ \left[p_1^{2,2}, p_3^{2,2}\right], p_4^{2,2}\right]$$

Problem 12 *Demuestre que el conjunto* $S = \left[(x, y, \alpha, \beta, \gamma) \in \omega^2 \times \Sigma^{*3} : x \leq |\gamma|\right]$ *es* Σ*-p.r.*

Observe que

$$\chi_S^{\omega^2 \times \Sigma^{*3}} = \lambda xy\,[x \leq y] \circ \left[p_1^{2,3}, \lambda\alpha\,[|\alpha|] \circ p_5^{2,3}\right]$$

Puesto que $\lambda xy\,[x \leq y]$ es Σ-p.r. y $\lambda\alpha\,[|\alpha|]$ también, $\chi_S^{\omega^2 \times \Sigma^{*3}}$ es Σ-p.r.

$\therefore$ S es Σ-p.r.

Problem 13 *Sea* $\Sigma = [@, ?]$. *Demuestre que*

$$f : \{(x, y, \alpha) : x \leq y\} \mapsto \omega$$
$$(x, y, \alpha) \mapsto \begin{cases} x^2 & |\alpha| \leq y \\ 0 & |\alpha| > y \end{cases}$$

es Σ*-p.r.*

Sean

$$S_1 = \left\{(x, y, \alpha) \in \omega^2 \times \Sigma^* : x \leq y \wedge |\alpha| \leq y\right\}$$
$$S_2 = \left\{(x, y, \alpha) \in \omega^2 \times \Sigma^* : x \leq y \wedge |\alpha| > y\right\}$$

Evidentemente, $S_1 \cap S_2 = \emptyset$. Es claro que cada conjunto corresponde a uno de los casos de f, y que $S_1 \cup S_2 = \mathcal{D}_f$.

Ahora bien, la función $f_1 := \lambda xy\alpha\left[x^2\right]$ es evidentemente Σ-p.r. Lo mismo aplica a la función $f_2 := C_0^{2,1}$. Más aún, es fácil probar que S_1, S_2 son Σ-p.r. (esto lo dejamos). Luego, puesto que la restricción de una función Σ-p.r. a un dominio Σ-p.r. es a su vez una función Σ-p.r., tenemos que $f_{1|S_1}, f_{2|S_2}$ son Σ-p.r. Luego $f = f_{1|S_1} \cup f_{2|S_2}$ es Σ-p.r.

Problem 14 *Pruebe que la función* $\lambda xx_1\left[\sum_{t=1}^{t=x} Pred(x_1)^t\right]$ *es* Σ*-p.r.*

(1) Evidentemente, $\lambda xy\,[Pred(x)^y] = \lambda xy\,[x^y] \circ \left[Pred \circ p_1^{2,0}, p_2^{2,0}\right]$ es Σ-p.r.

(2) Considere la función $G := \lambda xyx_1\left[\sum_{t=x}^{t=y} Pred(x_1)^t\right]$. Pues $Pred(x_1)^t$ es Σ-p.r. sabemos que G es Σ-p.r. Evidentemente, la función del ejercicio es

$$G \circ \left[C_1^{2,0}, p_1^{2,0}, p_2^{2,0}\right]$$

Luego es Σ-p.r.

Problem 15 *Lo mismo para* $\mathcal{F} := \lambda xyz\alpha\beta\left[\subset_{t=3}^{t=z+5} \alpha^{Pred(z)\cdot t}\beta^{Pred(Pred(|\alpha|))}\right]$

Observe que $\mathcal{D}_{\mathcal{F}} = \left\{(x, y, z, \alpha, \beta) \in \omega^3 \times \Sigma^{*2} : z \geq 1 \wedge |\alpha| \geq 2\right\}$, pues la función $Pred$ no está definida para el valor cero.

(1) Observe que

$$f_1 := \lambda xy\alpha\beta\left[\alpha^{Pred(x)y}\right] = \lambda x\alpha\left[\alpha^x\right] \circ \left[\lambda xy\left[Pred(x).y\right] \circ [p_1^{2,2}, p_2], p_3^{2,2}\right]$$
$$f_2 := \lambda xy\alpha\beta\left[\beta^{Pred(Pred(|\alpha|))}\right] = \lambda x\alpha\left[\alpha^x\right] \circ \left[Pred \circ \left[Pred \circ \left[\lambda\alpha\left[|\alpha|\right] \circ p_3^{2,2}\right]\right], p_4^{2,2}\right]$$

Luego

$$f := \lambda xy\alpha\beta\left[f_1(x, y, \alpha, \beta)f_2(x, y, \alpha, \beta)\right] = \lambda\alpha\beta\left[\alpha\beta\right] \circ \left[f_1, f_2\right]$$

es Σ-p.r. Esta es la función que está dentro de la concatenación.

(2) Sea $G := \lambda xyz\alpha\beta\left[\subset_{t=x}^{t=y} f(z, t, \alpha, \beta)\right]$. Sabemos que, dado que f es Σ-p.r., G es Σ-p.r. Ahora bien,

$$\mathcal{F} = G \circ \left[C_3^{3,2}, \lambda x\left[x + 5\right] \circ p_3^{3,2}, p3^{3,2}, p_4^{3,2}, p_5^{3,2}\right]$$

Luego $\mathcal{F}$ es Σ-p.r.

Problem 16 *Use que* $x \in \mathbb{N}$ *es primo si y solo si* $x > 1 \wedge ((\forall t \in \omega)_{t \leq x}\ t = x \vee \neg(t \mid x))$ *para demostrar que* $\lambda x\left[x \text{ es primo }\right]$ *es* Σ*-p.r.*

Definamos $P_1 = \lambda\left[x > 1\right], P_2 = \lambda x\left[(\forall t \in \omega)_{t \leq x}\, t = x \vee \neg\,(t \mid x)\right]$. Observe que el predicado $P' = \lambda tx\left[t = x \vee \neg(t \mid x)\right]$ es Σ-p.r. (se deja al lector). Pues P' es Σ-p.r. tenemos que $P_2 = \lambda x\left[(\forall t \in \omega)_{t \leq x} P'(t, x)\right]$ es Σ-p.r. Dado que $\mathcal{D}_{P_1} = \mathcal{D}_{P_2}$ podemos tomar $P = P_1 \wedge P_2$ y P es Σ-p.r. Es evidente que $P = \lambda x\left[x \text{ es primo}\right]$.

Problem 17 *Pruebe que* $L = \left\{(x, \alpha, \beta) \in \omega \times \Sigma^{*2} : (\exists t \in \omega)\ \alpha^x = \beta^t\right\}$ *es* Σ*-p.r.*

El predicado siendo cuantificado es trivialmente Σ-p.r. y así lo es a su vez ω (pues $\chi_\omega^\omega = C_1^{1,0}$). Fijemos un elemento arbitrario $(x, \alpha, \beta) \in L$. Pues $\alpha^x = \beta^t$, se sigue que $|\alpha|x = |\beta|t$. Si $t > |\alpha|x$ es imposible que la igualdad se cumpla, asumiendo que $|\beta| \neq 0$. Si $\beta = \varepsilon$ entonces o bien $x = 0$ o $\alpha = \varepsilon$, en cuyos casos la igualdad vale para cualquier $t \in \omega$, incluyendo $t = 0$. Se sigue que en todo caso $t \leq |\alpha|x$.

Ahora que dimos con una cota para t, observe que $f = \lambda x\alpha\left[|\alpha|x\right]$ es Σ-p.r. (se deja al lector). Luego

$$\chi_L^{\omega\times\Sigma^{*2}} = \lambda x x_0 \alpha\beta\left[(\exists t \in \omega)_{t\leq x}\, \alpha^{x_0} = \beta^t\right] \circ \left[f \circ [p_1^{1,2}, p_2^{1,2}]\,, p_1^{1,2}, p_2^{1,2}, p_3^{1,2}\right]$$

que es Σ-p.r.

Problem 18 *Sea* $\Sigma = \{@, ?\}$. *Demuestre que*

$$L = \left\{ (x, \alpha, \beta) \in \mathbb{N} \times \Sigma^* \times \Sigma^+ : (\exists \gamma \in \Sigma^*) \; @\beta@ = \gamma?\alpha?\gamma^R \right\}$$

es Σ*-p.r.*

(1) Sea $P_0 = \lambda\alpha\beta\gamma \left[@\beta@ = \gamma?\alpha?\gamma^R \right]$. Para demostrar que es Σ-p.r. observe que $\lambda\alpha \left[\alpha^R\right] = R\left(C_\varepsilon^{0,2}, \left\{a \in \Sigma : \left(a, \lambda\alpha\beta\left[\alpha\beta\right] \circ \left[p_3^{0,4}, p_4^{0,4}\right]\right)\right\}\right)$. Pues tomar la recíproca de una palabra es una función Σ-p.r. se sigue fácilmente que P_0 es Σ-p.r.

(2) Sea $(x, \alpha, \beta) \in L$ un elemento arbitrario. Considere $\gamma \in \Sigma^*$ t.q. $@\beta@ = \gamma?\alpha?\gamma^R$. Evidentemente,

$$\begin{aligned} |@\beta@| = |\gamma?\alpha?\gamma^R| &\Rightarrow |\beta| + 2 = 2 + 2|\gamma| + |\alpha| \\ &\Rightarrow |\beta| - |\alpha| = 2|\gamma| \end{aligned}$$

Si $|\beta| - |\alpha|$ es par obtenemos que $(|\beta| - |\alpha|)/2$ es una cota. Si es impar entonces $(|\beta| - |\alpha| + 1)/2$ es una cota. Como este último valor es superior a $|\gamma|$ en ambos casos, lo tomamos como la cota de t. Es trivial observar que $f := \lambda\alpha\beta\left[(|\alpha| - |\beta| + 1)/2\right]$ es Σ-p.r., pues la división entera es Σ-p.r.

(3) Tenemos entonces que

$$\begin{aligned} \chi_L^{\omega\times\Sigma^{*2}} = \lambda xy\alpha\beta &\left[(\exists\gamma \in \Sigma^*)_{|\gamma| \le x} \;\; @\beta@ = \gamma?\alpha?\gamma^R \right] \\ &\circ \left[f \circ \left[p_2^{1,2}, p_3^{1,2}\right], p_1^{1,2}, p_2^{1,2}, p_3^{1,2} \right] \\ &\wedge \lambda x\alpha \left[x \neq 0 \wedge \alpha \neq \epsilon \right] \circ \left[p_1^{1,2}, p_3^{1,2} \right] \end{aligned}$$

Observe que ambos predicados tienen el mismo dominio y en consecuencia están bien definidos. El segundo predicado asegura que respetemos que los elementos de L son de $\mathbb{N}\times\Sigma^*\times\Sigma^+$. Pues la función dada arriba es claramente Σ-p.r., hemos probado que L es Σ-p.r.

Problem 19 *Pruebe que*

$$L = \left\{ (x, \alpha, \beta) \in \omega \times \Sigma^{*2} : (\exists t \in Im(pr)) \;\; \alpha^{Pred(Pred(x))\cdot Pred(|\alpha|)} = \beta^t \right\}$$

es Σ*-p.r.*

(1) Sea $P_0 = \lambda x_0 x_1 \alpha\beta \left[\alpha^{Pred(Pred(x_0))\cdot Pred(|\alpha|)} = \beta^{x_1}\right]$. Salteamos la prueba de que P_0 es Σ-p.r. porque es mecánica.

(2) Sabemos que $\chi_L^{\omega\times\Sigma^{*2}}$ es el predicado $P = \lambda x_0\alpha\beta\left[(\exists t \in Im(pr))\ P_0(x_0, t, \alpha, \beta)\right]$. Sea $(x_0, \alpha, \beta) \in L$ un elemento arbitrario y $t \in Im(pr)$.

$$\alpha^{(x_0-2)(|\alpha|-1)} = \beta^t \Rightarrow |\alpha|(x_0-2)(|\alpha|-1) = |\beta|t$$

Un razonamiento similar al de un ejercicio anterior nos lleva a concluir que $t \leq |\alpha|(x_0-2)(|\alpha|-1)$. Es muy fácil ver que $f := \lambda x\alpha\left[(x-2)(|\alpha|-1)\right]$ es Σ-p.r. (de hecho, si demostró que P_0 es Σ-p.r. ya demostró que f es Σ-p.r.).

(3) Se sigue de todo lo anterior que

$$\begin{aligned}\chi_L^{\omega\times\Sigma^{*2}} = \lambda xy\alpha\beta &\left[(\exists t \in Im(pr))_{t\leq x}\, \alpha^{(2-y)(|\alpha|-1)} = \beta^t\right] \\ &\circ \left[f \circ \left[p_1^{1,2}, p_2^{1,2}\right], p_1^{1,2}, p_2^{1,2}, p_3^{1,2}\right]\end{aligned}$$

(4) Como última observación, vea que $Im(pr)$ es Σ-computable. En efecto,

$$\chi_{Im(pr)}^{\omega} = \lambda x\left[\neg\left((\exists t \in \omega)_{t\leq x}\ t \mid x\right)\right]$$

Pues ω es Σ-p.r. y $t \mid x$ es Σ-p.r. tenemos que $\chi_{Im(pr)^{\omega}}$ es Σ-p.r. Luego, $\chi_L^{\omega\times\Sigma^{*2}}$ es Σ-p.r.

Problem 20 *Use the* ***U rule*** *to find a predicate P s.t.* $M(P) = \lambda x[\text{integer part of } \sqrt{x}]$
.

Let $f(x)$ denote the integer part of $\sqrt{x}$. If $f(x) = y$ then $y^2 \leq x \wedge (y+1)^2 > x$. Then letting $P = \lambda xy\left[y^2 \leq x \wedge (y+1)^2 > x\right]$ ensures that $M(P(x,y)) = f(x)$.

Problem 21 *Is is true that* $Ins^{\Sigma} \cap Pro^{\Sigma} = \emptyset$*? And is it true that* $\lambda i\mathcal{P}[I_i^{\mathcal{P}}]$ *has domain* $\left\{(i,\mathcal{P}) \in \mathbb{N}\times Pro^{\Sigma} : i \leq n(\mathcal{P})\right\}$*?*

Both statements are false. A single instruction in Ins^{Σ} can be a program (as long as it is not a GOTO statement to a non-existent label). Furthermore, $\lambda i\mathcal{P}[I_i^{\mathcal{P}}]$ is defined for $i = 0$ (it maps to ε) and for $i \geq n(\mathcal{P})$ (it also maps to ε).

Problem 22 *Prove: If* $\mathcal{P}_1, \mathcal{P}_2 \in Pro^{\Sigma}$ *then* $\mathcal{P}_1\mathcal{P}_1 = \mathcal{P}_2\mathcal{P}_2 \Rightarrow \mathcal{P}_1 = \mathcal{P}_2$.

This follows from the theorem that guarantees that any program $\mathcal{P} \in Pro^{\Sigma}$ is a *unique* concatenation of instructions. Let $\mathcal{P}_1 = I_1^{\mathcal{P}_1} \dots I_{n(\mathcal{P}_1)}^{\mathcal{P}_1}$ and $\mathcal{P}_2 = I_1^{\mathcal{P}_2} \dots I_{n(\mathcal{P}_2)}^{\mathcal{P}_2}$. Assume $\mathcal{P}_1\mathcal{P}_1 = \mathcal{P}_2\mathcal{P}_2$. Then

$$I_1^{\mathcal{P}_1} \dots I_{n(\mathcal{P}_1)}^{\mathcal{P}_1} I_1^{\mathcal{P}_1} \dots I_{n(\mathcal{P}_1)}^{\mathcal{P}_1} = I_2^{\mathcal{P}_2} \dots I_{n(\mathcal{P}_2)}^{\mathcal{P}_2} I_2^{\mathcal{P}_2} \dots I_{n(\mathcal{P}_2)}^{\mathcal{P}_2}$$

Then, it is a theorem that $I_k^{\mathcal{P}_1} = i_k^{\mathcal{P}_2}$. From this follows directly that $\mathcal{P}_1 = \mathcal{P}_2$. ■

Problem 23 *Give true or false for the following statements.*

Statement 1: If $S_{\mathcal{P}}(i, \overrightarrow{s}, \overrightarrow{\alpha}) = (i, \overrightarrow{s}, \overrightarrow{\alpha})$ then $i \notin [1, n(\mathcal{P})]$. The statement is false. It could be the case that $i \notin [1, n(\mathcal{P})]$, in which case we would say the program halted. However, consider the program

$$L1 \; GOTO \; L1$$

Evidently, $S_{\mathcal{P}}(1, \overrightarrow{s}, \overrightarrow{\alpha}) = (1, \overrightarrow{s}, \overrightarrow{\alpha})$, and $1 \leq 1 \leq n(\mathcal{P})$.

Statement 2. Let $\mathcal{P} \in Pro^{\Sigma}$ and d an instantaneous description whose first coordinate is i. If $I_i^{\mathcal{P}} = N_2 \leftarrow N_2 + 1$, then

$$S_{\mathcal{P}}(d) = (i + 1, (N_1, Suc(N_2), N_3, \dots), (P_1, P_2, P_3, \dots))$$

The statement is true via direct application of the $S_{\mathcal{P}}$ function.

Statement 3. Let $\mathcal{P} \in Pro^{\Sigma}$ and $(i, \overrightarrow{s}, \overrightarrow{\sigma})$ an instantaneous description. If $Bas(I_i^{\mathcal{P}}) = IF\ P_3\ BEGINS\ a\ GOTO\ L_6$ and $[P_3]_1 = a$, then $S_{\mathcal{P}}(i, \overrightarrow{s}, \overrightarrow{\sigma}) = (j, \overrightarrow{s}, \overrightarrow{\sigma})$, where j is the least number l s.t. $I_l^{\mathcal{P}}$ has label L_6.

Because $[P_3]_1 = a$, the value of $S_{\mathcal{P}}(i, \overrightarrow{s}, \overrightarrow{\sigma})$ must indeed contain the instruction that has label L_6. This instruction is the jth instruction for some j, etc. The statement is true.

Problem 24 *Let $\Sigma = \{@, !\}$. Give a program that computes $f : \{0, 1, 2\} \mapsto \omega$ given by $f(0) = f(1) = 0, f(2) = 5$.*

Evidently $f \sim (1, 0, \#)$ and so we must find some $\mathcal{P} \in Pro^{\Sigma}$ s.t. $\Psi_{\mathcal{P}}^{1,0,\#}(x) = f(x)$. The program must let N_1 hold the value 0 if the starting state is either [[0]] or [[1]], and the value 5 if the starting state is [[2]]. In all other cases, it must not halt, to ensure that the domain of $\Psi_{\mathcal{P}}^{1,0,\#}$ is the same as that of f. The desired program is

$$
\begin{array}{ll}
 & N_2 \leftarrow N_1 \\
 & N_2 \leftarrow N_2 - 1 \\
 & IF\ N_2 \neq 0\ GOTO\ L_1 \\
 & GOTO\ L_4 \\
L_1 & N_2 \leftarrow N_2 - 1 \\
 & IF\ N_2 \neq 0\ GOTO\ L_2 \\
 & GOTOL_3 \\
L_2 & GOTO\ L_2 \\
L_3 & N_1 \leftarrow N_1 + 1 \\
 & N_1 \leftarrow N_1 + 1 \\
 & N_1 \leftarrow N_1 + 1 \\
 & GOTO\ L_5 \\
L_4 & N_1 \leftarrow 0 \\
L_5 & SKIP
\end{array}
$$

If $\mathcal{P}$ denotes this program, it is evident that $\mathcal{P}$ only halts for starting states $[[x_1]]$ with $x_1 \in \{0, 1, 2\}$. Thus, the domain of $\Psi^{1,0,\#}_{\mathcal{P}}$ is precisely $\mathcal{D}_f$. It is easy to verify that, more generally, $\Psi^{1,0,\#}_{\mathcal{P}} = f$.

Problem 25 *Using the same alphabet as in the previous problem, find $\mathcal{P} \in Pro^{\Sigma}$ that computes $\lambda xy[x + y]$.*

The desired program is

$$
\begin{array}{ll}
L_1 & IF\ N_2 = 0\ GOTO\ L_3 \\
 & N_1 \leftarrow N_1 + 1 \\
 & N_2 \leftarrow N_2 - 1 \\
 & GOTO\ L_1 \\
L_3 & SKIP
\end{array}
$$

Problem 26 *Same for $C_0^{1,1}{}_{|\{0,1\}\times\Sigma^*}$*

Since the domain of the constant function is restricted t o $\{0, 1\} \times \Sigma^*$, we must ensure the program only halts for states $[[x_1, x_2, \alpha]]$ s.t. $x_1, x_2 \in \{0, 1\}$. Thus, the program is

$$
\begin{aligned}
& N_1 \leftarrow N_1 - 1 \\
& N_2 \leftarrow N_2 - 1 \\
& IFN_2 \neq 0\, GOTO\ L_1 \\
& IFN_1 \neq 0\, GOTO\ L_1 \\
& GOTO\ L_2 \\
L_1\ & GOTO\ L_1 \\
L_2\ & SKIP
\end{aligned}
$$

Problem 27 *Same for $\lambda i\alpha[[\alpha]_i]$ (same alphabet).*

$$
\begin{aligned}
& IF\ N_0 \neq 0\, GOTO\ L_1 \\
& P_1 \leftarrow \varepsilon \\
& GOTO\ L_{100} \\
L_1\ & N_1 \leftarrow N_1 - 1 \\
L_2\ & N_1 \leftarrow N_1 - 1 \\
& P_1 \leftarrow {}^{\frown}P_1 \\
& IF\ N_1 \neq 0\, GOTO\ L_2 \\
& IF\ P_1\ STARTSWITH\ @\ GOTO\ L_2 \\
& IF\ P_1\ STARTSWITH\ !\ GOTOL_3 \\
& GOTOL_{100} \\
L_3\ & P_1 \leftarrow\ ! \\
L_2\ & P_1 \leftarrow @ \\
L_{100}\ & SKIP
\end{aligned}
$$

Example. Let $\alpha = @!!@@$. Assume we give $[[4, \alpha]]$. Since $4 \neq 0$ we go to L_1 immediately. Here N_1 is set to three. Then N_1 is set to two and P_1 is set to $!!@@$. Since $N_1 \neq 0$, N_1 is now set to 1 and P_1 to $!@@$. Once more, N_1 is now set to 0 and P_1 to $@@$. Since now $N_1 = 0$, we know the starting character of P_1 is the one we looked for. We set P_1 to be its first character (if $P_1 = \varepsilon$ it has no first character and nothings needs to be done, because this means the input $[[x_1, \alpha]]$ had $x_1 > |\alpha|$). The other cases also work.