Nota: Los ejercicios que tienen (S) son para una "Segunda vuelta" es decir conviene hacerlos una vez que ya se completó la guía haciendo los otros y ya se tiene mas madurez e intuición basica sobre los conceptos. Los que tienen (O) son opcionales por lo cual no se toman en los examenes.

El paradigma de Godel: Funciones Σ-recursivas

En esta guia y la siguiente desarrollaremos el modelo matematico del concepto de funcion Σ-efectivamente computable, dado por Godel. Dichas funciones seran llamadas Σ-recursivas. La idea es partir de un conjunto inicial de funciones muy simples y obviamente Σ-efectivamente computables y luego obtener nuevas funciones Σ-efectivamente computables usando constructores que preservan la computabilidad efectiva. Las funciones Σ-recursivas seran las que se obtienen iterando el uso de estos constructores, partiendo del conjunto inicial de funciones antes mencionado. Nos referiremos a este paradigma como el paradigma Godeliano o recursivo. A veces tambien lo llamaremos el paradigma funcional.

La familia de funciones simples y obviamente Σ-efectivamente computables de la que partiremos es la siguiente

$$\left\{Suc, Pred, C_0^{0,0}, C_\varepsilon^{0,0}\right\} \cup \{d_a : a \in \Sigma\} \cup \left\{p_j^{n,m} : 1 \leq j \leq n+m\right\}$$

Los constructores que usaremos son:

- Composicion
- Recursion primitiva
- Minimizacion de predicados totales

Estos constructores nos permiten dadas ciertas funciones construir o definir una nueva funcion y tienen la propiedad de preservar la computabilidad efectiva en el sentido que si las funciones iniciales son Σ-efectivamente computables, entonces la funcion obtenida tambien lo es. Un concepto fundamental es el de funcion Σ-recursiva primitiva. Estas funciones seran aquellas que se obtienen a partir de las del conjunto inicial usando solo los dos primeros constructores: composicion y recursion primitiva. Nuestro primer objetivo es definir el concepto de funcion Σ-recursiva primitiva para lo cual en las proximas dos secciones definiremos y estudiaremos los constructores de composicion y recursion primitiva. Luego definiremos el concepto de funcion Σ-recursiva primitiva y nos abocaremos a desarrollar este concepto fundamental. Recien despues ya en la Guia 6 estudiaremos el constructor de minimizacion y definiremos el concepto de funcion Σ-recursiva.

Composicion

Dadas funciones Σ-mixtas $f, f_1, ..., f_r$, con $r \geq 1$, diremos que la funcion $f \circ [f_1, ..., f_r]$ es *obtenida por composicion a partir de las funciones* $f, f_1, ..., f_r$. Un hecho que a priori no es obvio es que si $f, f_1, ..., f_r$ son Σ-mixtas, entonces $f \circ [f_1, ..., f_r]$ lo es. Esto es consecuencia del siguiente lema.

Lemma 1 *Supongamos que $f, f_1, ..., f_r$ son funciones Σ-mixtas, con $r \geq 1$. Supongamos ademas que $f \circ [f_1, ..., f_r] \neq \emptyset$. Entonces hay $n, m, k, l \in \omega$ y $s \in \{\#, *\}$ tales que*

- *$r = n + m$*
- *f es de tipo (n, m, s)*
- *f_i es de tipo $(k, l, \#)$, para cada $i = 1, ..., n$*
- *f_i es de tipo $(k, l, *)$, para cada $i = n+1, ..., n+m$*

Mas aun, en tal caso la funcion $f \circ [f_1, ..., f_{n+m}]$ es Σ-mixta de tipo (k, l, s) y:

$$D_{f \circ [f_1, ..., f_{n+m}]} = \left\{ (\vec{x}, \vec{\alpha}) \in \bigcap_{i=1}^{n+m} D_{f_i} : (f_1(\vec{x}, \vec{\alpha}), ..., f_{n+m}(\vec{x}, \vec{\alpha})) \in D_f \right\}$$

$$f \circ [f_1, ..., f_{n+m}](\vec{x}, \vec{\alpha}) = f(f_1(\vec{x}, \vec{\alpha}), ..., f_{n+m}(\vec{x}, \vec{\alpha})).$$

Ejercicio 1: Justifique con palabras la veracidad del lema anterior

Ahora si es facil probar que la composicion preserva la computabilidad efectiva. Mas formalmente:

Lemma 2 *Si $f, f_1, ..., f_r$, con $r \geq 1$, son Σ-efectivamente computables, entonces $f \circ [f_1, ..., f_r]$ lo es.*

Proof. Si $f \circ [f_1, ..., f_r] = \emptyset$, entonces claramente es Σ-efectivamente computable. Supongamos entonces que $f \circ [f_1, ..., f_r] \neq \emptyset$. Por el lema anterior hay $n, m, k, l \in \omega$ y $s \in \{\#, *\}$ tales que

- $r = n + m$
- f es de tipo (n, m, s)
- f_i es de tipo $(k, l, \#)$, para cada $i = 1, ..., n$
- f_i es de tipo $(k, l, *)$, para cada $i = n+1, ..., n+m$

Sean $\mathbb{P}, \mathbb{P}_1, ..., \mathbb{P}_{n+m}$ procedimientos efectivos los cuales computen las funciones $f, f_1, ..., f_{n+m}$, respectivamente. Usando estos procedimientos es facil definir un procedimiento efectivo el cual compute a $f \circ [f_1, ..., f_{n+m}]$ ■

Ejercicio 2: Complete la prueba anterior

Recursion primitiva

La recursion primitiva es un tipo muy particular de recursion. Mas adelante lo definiremos matematicamente pero antes daremos varios ejemplos para aproximarnos gradualmente a la definicion. Consideremos por ejemplo las siguientes ecuaciones:

(1) $R(0) = 1$

(2) $R(t + 1) = 1 + R(t) + R(t)^2$

Notese que hay una unica funcion $R : \omega \to \omega$ la cual cumple (1) y (2). Esto es ya que el valor de R en t esta determinado por sucesivas aplicaciones de las ecuaciones (1) y (2). Por ejemplo la ecuacion (1) nos dice que $R(0) = 1$ pero entonces la ecuacion (2) nos dice que $R(1) = 1 + 1 + 1^2 = 3$ por lo cual nuevamente la ecuacion (2) nos dice que $R(2) = 1 + 3 + 3^2 = 13$ y asi podemos notar facilmente que R esta determinada por dichas ecuaciones.

Se suele decir que las ecuaciones (1) y (2) definen recursivamente a la funcion R pero hay que tener cuidado porque esto es una manera de hablar ya que la funcion R podria en nuestro discurso ya haber sido definida de otra manera. Mas propio es pensar que dichas ecuaciones determinan a R en el sentido que R es la unica que las cumple. Por ejemplo las ecuaciones:

(a) $R(0) = 50$

(b) $R(t + 1) = R(t)$

definen recursivamente a la funcion $C_{50}^{1,0}$ pero esta claro que la definicion de $C_{50}^{1,0}$ en esta materia no fue dada de esta forma.

Ejercicio 3: Encuentre ecuaciones que "definan recursivamente" a la funcion $R = \lambda t[2^t]$

Hay casos de recursiones en las cuales el valor de $R(t + 1)$ no solo depende de $R(t)$ sino que tambien depende de t. Por ejemplo

(i) $R(0) = 1$

(ii) $R(t + 1) = t.R(t) + 1$

De todas maneras deberia quedar claro que las ecuaciones (i) y (ii) determinan una unica funcion $R : \omega \to \omega$ que las satisface.

Ejercicio 4: Encuentre ecuaciones que "definan recursivamente" a la funcion $R = \lambda t[t!]$

Tambien podemos generalizar pensando que la funcion R depende no solo de un parametro t sino que tiene otras variables. Por ejemplo

(p) $R(0, x_1, x_2, x_3) = x_1 + 2x_3$

(q) $R(t + 1, x_1, x_2, x_3) = t + x_1 + x_2 + x_3 + R(t, x_1, x_2, x_3)$

Ejercicio 5: Explique con palabras por que las ecuaciones (p) y (q) determinan una unica funcion $R : \omega^4 \to \omega$. Cuanto vale $R(3, 1, 2, 3)$?

Por supuesto la cantidad de variables extra puede ser cualquiera y no justo 3.

Ejercicio 6: Encuentre ecuaciones que "definan recursivamente" a la funcion $R = \lambda t x_1 [t + x_1]$, usando la funcion Suc.

Tambien podriamos tener variables alfabeticas. Por ejemplo consideremos

(r) $R(0, x_1, x_2, \alpha_1, \alpha_2) = x_1 + |\alpha_1|^{x_2}$

(s) $R(t + 1, x_1, x_2, \alpha_1, \alpha_2) = t + x_1 + x_2 + |\alpha_1| + |\alpha_2| + R(t, x_1, x_2, \alpha_1, \alpha_2)$

Es claro aqui que las ecuaciones (r) y (s) determinan una unica funcion $R : \omega^3 \times \Sigma^{*2} \to \omega$ que las cumple. Esto se puede explicar de la siguiente manera:

- La ecuacion (r) determina los valores de R sobre el conjunto $\{0\} \times \omega \times \omega \times \Sigma^* \times \Sigma^*$. Pero una ves determinados estos valores, la ecuacion (s) tomada con $t = 0$, determina los valores de R sobre el conjunto $\{1\} \times \omega \times \omega \times \Sigma^* \times \Sigma^*$. Pero una ves determinados estos valores, la ecuacion (s) tomada con $t = 1$, determina los valores de R sobre el conjunto $\{2\} \times \omega \times \omega \times \Sigma^* \times \Sigma^*$, etc

El caso anterior podria generalizarse de la siguiente manera: Si tenemos dadas dos funciones

$$f : \omega^n \times \Sigma^{*m} \to \omega$$
$$g : \omega^{n+2} \times \Sigma^{*m} \to \omega$$

entonces las ecuaciones:

(a) $R(0, \vec{x}, \vec{\alpha}) = f(\vec{x}, \vec{\alpha})$

(b) $R(t + 1, \vec{x}, \vec{\alpha}) = g(R(t, \vec{x}, \vec{\alpha}), t, \vec{x}, \vec{\alpha})$

determinan una unica funcion $R : \omega^{n+1} \times \Sigma^{*m} \to \omega$ que las cumple. Notese que para el caso

$$n = m = 2$$
$$f = \lambda x_1 x_2 \alpha_1 \alpha_2 [x_1 + |\alpha_1|^{x_2}]$$
$$g = \lambda x t x_1 x_2 \alpha_1 \alpha_2 [t + x_1 + x_2 + |\alpha_1| + |\alpha_2| + x]$$

las ecuaciones (a) y (b) se transforman en las ecuaciones (r) y (s).

Conjuntos rectangulares

El primer caso de recursion primitiva que definiremos a continuacion engloba todos los ejemplos vistos recien dentro de un marco general. Para enunciarlo necesitaremos una definicion muy importante en la materia. Sea Σ un alfabeto finito. Un conjunto Σ-mixto S es llamado *rectangular* si es de la forma

$$S_1 \times ... \times S_n \times L_1 \times ... \times L_m$$

con cada $S_i \subseteq \omega$ y cada $L_i \subseteq \Sigma^*$. Notar que todo subconjunto de ω es rectangular (es el caso $n = 1$ y $m = 0$). Analogamente, todo subconjunto de Σ^* es rectangular (es el caso $n = 0$ y $m = 1$). Tambien $\{\Diamond\}$ es rectangular (es el caso $n = m = 0$). Otros ejemplos:

- $\mathbf{N} \times \{1, 2\} \times \{@@, \varepsilon\}$ es rectangular (aqui $n = 2$ y $m = 1$)
- $\{!!!, !!\} \times \{@@, \varepsilon\}$ es rectangular (aqui $n = 0$ y $m = 2$)

Tambien notese que $\emptyset = \emptyset \times \emptyset$ por lo cual $\emptyset$ es un conjunto rectangular.

El concepto de conjunto rectangular es muy importante en nuestro enfoque. Aunque en general no habra restricciones acerca del dominio de las funciones y predicados, nuestra filosofia sera tratar en lo posible que los dominios de las funciones que utilicemos para hacer nuestro analisis de recursividad de los distintos paradigmas, sean rectangulares.

Aunque en principio puede parecer que todos los conjuntos son rectangulares, el siguiente lema mostrara cuan ingenua es esta vision. Lo aceptaremos sin demostracion aunque es facil de probar.

Lemma 3 *Sea $S \subseteq \omega \times \Sigma^*$. Entonces S es rectangular si y solo si se cumple la siguiente propiedad:*

(R) Si $(x, \alpha), (y, \beta) \in S$, entonces $(x, \beta) \in S$

Ejercicio 7: Supongamos $\Sigma = \{\#, \blacktriangle, \%\}$. Use el lema anterior para probar que

$$\{(0, \#\#), (1, \%\%\%)\} \text{ y } \{(x, \alpha) \in \omega \times \Sigma^* : |\alpha| = x\}$$

no son rectangulares

Recursion primitiva sobre variable numerica con valores numericos

Ahora si daremos el primer caso del constructor de recursion primitiva. Supongamos tenemos dadas funciones

$$f : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \omega$$
$$g : \omega \times \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \omega$$

con $S_1,...,S_n \subseteq \omega$ y $L_1,...,L_m \subseteq \Sigma^*$ conjuntos no vacios. Usando el razonamiento inductivo usado en los ejemplos anteriores, se puede probar que hay una unica funcion

$$R : \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \omega$$

la cual cumple las ecuaciones

- $R(0, \vec{x}, \vec{\alpha}) = f(\vec{x}, \vec{\alpha})$
- $R(t+1, \vec{x}, \vec{\alpha}) = g(R(t, \vec{x}, \vec{\alpha}), t, \vec{x}, \vec{\alpha})$

LLamaremos $R(f,g)$ a esta unica funcion que cumple las ecuaciones anteriores. Resumiendo, diremos que las ecuaciones

(1) $R(f,g)(0, \vec{x}, \vec{\alpha}) = f(\vec{x}, \vec{\alpha})$

(2) $R(f,g)(t+1, \vec{x}, \vec{\alpha}) = g(R(f,g)(t, \vec{x}, \vec{\alpha}), t, \vec{x}, \vec{\alpha})$

definen recursivamente a la funcion $R(f,g)$. Tambien diremos que $R(f,g)$ es obtenida por *recursion primitiva* a partir de f y g.

NOTA IMPOTANTE: No confundirse y pensar que $R(f,g)$ es el resultado de aplicar una funcion R al par (f,g), de hecho hasta el momento no hemos definido ninguna funcion R cuyo dominio sea cierto conjunto de pares ordenados de funciones!

Ejercicio 8: Justifique con palabras que la funcion $R(f,g)$ esta bien definida, es decir que dada una $(1+n+m)$-upla $(t, \vec{x}, \vec{\alpha})$ perteneciente a $\omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m$, las ecuaciones de (1) y (2) determinan el valor $R(f,g)(t, \vec{x}, \vec{\alpha})$

Notese que cuando $n = m = 0$, se tiene que $D_f = \{\Diamond\}$ y (1) y (2) se transforman en

(1) $R(f,g)(0) = f(\Diamond)$

(2) $R(f,g)(t+1) = g(R(f,g)(t), t)$

Veamos algunos ejemplos

E_1 Tomemos $f = p_1^{1,0}$ y $g = Suc \circ p_1^{3,0}$. De la definicion de $R(f,g)$, obtenemos que su dominio es ω^2 y

$$R(f,g)(0, x_1) = p_1^{1,0}(x_1) = x_1$$
$$R(f,g)(t+1, x_1) = \left(Suc \circ p_1^{3,0}\right)(R(f,g)(t, x_1), t, x_1) = R(f,g)(t, x_1) + 1$$

Es facil notar que la unica funcion que cumple estas dos ecuaciones es $\lambda t x_1 [t + x_1]$, lo cual implica que $\lambda t x_1 [t + x_1] = R\left(p_1^{1,0}, Suc \circ p_1^{3,0}\right)$

E$_2$ Sean $f = C_0^{0,0}$ y $g = p_1^{2,0}$. De la definicion de $R(f,g)$, obtenemos que su dominio es ω y

$$R(f,g)(0) = C_0^{0,0}(\Diamond) = 0$$
$$R(f,g)(t+1) = p_1^{2,0}(R(f,g)(t),t) = R(f,g)(t)$$

Es facil notar que la unica funcion que cumple estas dos ecuaciones es $C_0^{1,0}$ lo cual implica que $C_0^{1,0} = R\left(C_0^{0,0}, p_1^{2,0}\right)$

Nota importante: En los dos ejemplos anteriores y en todos los casos que manejaremos en la Guia 5, en las aplicaciones del constructor de recursion primitiva (en sus cuatro formas) las funciones iniciales seran Σ-totales (es decir $S_1 = ... = S_n = \omega$ y $L_1 = ... = L_m = \Sigma^*$). Solo a partir de la Guia 6 veremos aplicaciones con funciones no Σ-totales

Ejercicio 9: Sean $n, m, k \in \omega$ y supongamos $n \geq 1$. Encuentre f y g tales que $R(f,g) = C_k^{n,m}$

Ejercicio 10: Encuentre f y g tales que $R(f,g) = \lambda tx_1[t.x_1]$. Idem para $\lambda tx_1\alpha_1\alpha_2[t.x_1]$

Recordemos que por definicion teniamos que $0^0 = 1$. Esto nos dice que $D_{\lambda xy[x^y]} = \omega \times \omega$.

Ejercicio 11: Encuentre f y g tales que $R(f,g) = \lambda tx_1[x_1^t]$

Ejercicio 12: Encuentre f y g tales que $R(f,g) = \lambda t[t!]$

Ejercicio 13: Explique la forma en la que aplicando los constructores de composicion y recursion primitiva a las funciones del conjunto inicial se puede obtener la funcion $\lambda x_1x_2\alpha_1[x_1!]$

Ejercicio 14: V o F o I, justifique.

(a) $\lambda tx_1 [t + x_1] = R\left(p_1^{1,0}, Suc \circ p_1^{2,0}\right)$

(b) $R(\lambda xy[0], p_2^{4,0}) = p_1^{3,0}$

(c) Si $f : \omega^2 \to \omega$ y $g : \omega^4 \to \omega$, entonces para cada $(x,y) \in \omega^2$, se tiene que $R(f,g)(2,x,y) = g \circ (g \circ [f \circ [p_2^{3,0}, p_3^{3,0}], p_1^{3,0}, p_2^{3,0}, p_3^{3,0}])(0,x,y)$

Como era de esperar, este caso del constructor de recursion primitiva preserva la computabilidad efectiva

Lemma 4 *Si f y g son Σ-efectivamente computables, entonces $R(f,g)$ lo es.*

Ejercicio 15: Suponga que tiene procedimientos efectivos $\mathbb{P}_f$ y $\mathbb{P}_g$ que computan a f y g, respectivamente. Construya un procedimiento efectivo que compute a $R(f,g)$. Concluya que el lema anterior es cierto.

Recursion primitiva sobre variable numerica con valores alfabeticos

Ahora haremos el caso en el que la funcion definida recursivamente tiene imagen contenida en Σ^*. Es claro que entonces f y g tambien deberan tener imagen contenida en Σ^*. El unico detalle a tener en cuenta en la definicion de este caso es que si solo hicieramos estos cambios y pusieramos las mismas ecuaciones la funcion g no resultaria Σ-mixta en general. Para que la g de la recursion siga siendo Σ-mixta deberemos modificar levemente su dominio en relacion al caso ya hecho

Supongamos Σ es un alfabeto finito. Sean

$$\begin{aligned} &f : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \Sigma^* \\ &g : \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \to \Sigma^* \end{aligned}$$

con $S_1, ..., S_n \subseteq \omega$ y $L_1, ..., L_m \subseteq \Sigma^*$ conjuntos no vacios. Definamos

$$R(f, g) : \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \Sigma^*$$

de la siguiente manera

(1) $R(f, g)(0, \vec{x}, \vec{\alpha}) = f(\vec{x}, \vec{\alpha})$

(2) $R(f, g)(t + 1, \vec{x}, \vec{\alpha}) = g(t, \vec{x}, \vec{\alpha}, R(f, g)(t, \vec{x}, \vec{\alpha}))$

Diremos que $R(f, g)$ es obtenida por *recursion primitiva* a partir de f y g. Notese que cuando $m = n = 0$, se tiene que $D_f = \{\Diamond\}$ y (1) y (2) se transforman en

(1) $R(f, g)(0) = f(\Diamond)$

(2) $R(f, g)(t + 1) = g(t, R(f, g)(t))$

Veamos algunos ejemplos

E$_1$ Tomemos $f = C_\varepsilon^{0,1}$ y $g = \lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{1,2}, p_2^{1,2}\right]$. De la definicion de $R(f, g)$, obtenemos que

$$\begin{aligned} R(f, g)(0, \alpha_1) &= C_\varepsilon^{0,1}(\alpha_1) = \varepsilon \\ R(f, g)(t + 1, \alpha_1) &= \lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{1,2}, p_2^{1,2}\right](t, \alpha_1, R(f, g)(t, \alpha_1)) = R(f, g)(t, \alpha_1)\alpha_1 \end{aligned}$$

Es facil notar que la unica funcion que cumple estas dos ecuaciones es $\lambda t\alpha_1\,[{\alpha_1}^t]$, lo cual implica que $\lambda t\alpha_1\,[{\alpha_1}^t] = R\left(C_\varepsilon^{0,1}, \lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{1,2}, p_2^{1,2}\right]\right)$

E$_2$ Sean $f = C_\varepsilon^{0,0}$ y $g = p_2^{1,1}$. De la definicion de $R(f, g)$, obtenemos que

$$\begin{aligned} R(f, g)(0) &= C_\varepsilon^{0,0}(\Diamond) = \varepsilon \\ R(f, g)(t + 1) &= p_2^{1,1}(t, R(f, g)(t)) = R(f, g)(t) \end{aligned}$$

Es facil notar que la unica funcion que cumple estas dos ecuaciones es $C_\varepsilon^{1,0}$ lo cual implica que $C_\varepsilon^{1,0} = R\left(C_\varepsilon^{0,0}, p_2^{1,1}\right)$

Ejercicio 16: Sea $\Sigma = \{\%, @, ?\}$. Encuentre f y g tales que $\lambda t x_1[\%@\%\%\%\%?^t] = R(f,g)$

Ejercicio 17: V o F o I, justifique.

(a) $C_\varepsilon^{2,,2} = R\left(C_\varepsilon^{1,2}, C_\varepsilon^{2,3}\right)$

(b) $R\left(C_\varepsilon^{1,1}, C_\varepsilon^{1,1}\right) = C_\varepsilon^{1,1}$

(c) Si f, g son funciones Σ-mixtas tales que $R(f,g)$ esta definida y es de tipo $(1+n, m, *)$, entonces f es de tipo $(n, m, *)$ y g es de tipo $(n, m+1, *)$

La prueba del siguiente lema es completamente analoga a la del lema anterior que fue dejada como ejercicio.

Lemma 5 *Si f y g son Σ-efectivamente computables, entonces $R(f,g)$ lo es.*

Recursion primitiva sobre variable alfabetica con valores numericos

Ya vimos dos casos de recursion donde el parametro o variable que comanda la recursion es numerico. Daremos a continuacion un ejemplo de recursion en el cual el parametro principal es alfabetico. Sea $\Sigma = \{\%, @, ?\}$ y consideremos las siguientes ecuaciones:

(1) $R(\varepsilon) = 15$

(2) $R(\alpha\%) = R(\alpha) + 1$

(3) $R(\alpha@) = R(\alpha).5$

(4) $R(\alpha?) = R(\alpha)^{20}$

Notese que las ecuaciones anteriores determinan una funcion $R : \Sigma^* \to \omega$. Esto es ya que R en ε debe valer 15 y sabiendo esto las ecuaciones (2), (3) y (4) (con $\alpha = \varepsilon$) nos dicen que

$$R(\%) = 16$$
$$R(@) = 75$$
$$R(?) = 15^{20}$$

por lo cual podemos aplicarlas nuevamente a dichas ecuaciones (con $\alpha \in \{\%, @, ?\}$) para calcular R en todas las palabras de longitud 2; y asi sucesivamente.

Daremos otro ejemplo un poco mas complicado para seguir aproximandonos al caso general. Nuevamente supongamos que $\Sigma = \{\%, @, ?\}$ y supongamos tenemos una funcion

$$f : \omega \times \Sigma^* \to \omega$$

y tres funciones

$$\mathcal{G}_{\%} : \omega \times \omega \times \Sigma^* \times \Sigma^* \to \omega$$
$$\mathcal{G}_{@} : \omega \times \omega \times \Sigma^* \times \Sigma^* \to \omega$$
$$\mathcal{G}_{?} : \omega \times \omega \times \Sigma^* \times \Sigma^* \to \omega$$

Entonces hay una unica funcion $R : \omega \times \Sigma^* \times \Sigma^* \to \omega$ la cual cumple las siguientes ecuaciones

(1) $R(x_1, \alpha_1, \varepsilon) = f(x_1, \alpha_1)$

(2) $R(x_1, \alpha_1, \alpha\%) = \mathcal{G}_{\%}(R(x_1, \alpha_1, \alpha), x_1, \alpha_1, \alpha)$

(3) $R(x_1, \alpha_1, \alpha@) = \mathcal{G}_{@}(R(x_1, \alpha_1, \alpha), x_1, \alpha_1, \alpha)$

(4) $R(x_1, \alpha_1, \alpha?) = \mathcal{G}_{?}(R(x_1, \alpha_1, \alpha), x_1, \alpha_1, \alpha)$

Ejercicio 18: Justifique que las ecuaciones anteriores determinan a la funcion R

Ejercicio 19: Por que el parametro α de la recursion es la ultima coordenada de R?

El ejemplo anterior nos muestra que para hacer recursion sobre parametro alfabetico nos hace falta "una funcion g por cada simbolo de Σ". Esto motiva la siguiente definicion. Dado un alfabeto Σ, una *familia Σ-indexada de funciones* sera una funcion $\mathcal{G}$ tal que $D_{\mathcal{G}} = \Sigma$ y para cada $a \in D_{\mathcal{G}}$ se tiene que $\mathcal{G}(a)$ es una funcion. Algunos ejemplos:

E$_1$ Sea $\mathcal{G}$ dada por

$$\begin{array}{rcl} \mathcal{G} : \{\Box, \%, \blacktriangle\} & \to & \{Suc, Pred\} \\ \Box & \to & Suc \\ \% & \to & Suc \\ \blacktriangle & \to & Pred \end{array}$$

Claramente $\mathcal{G}$ es una familia $\{\Box, \%, \blacktriangle\}$-indexada de funciones. Notar que

$$\mathcal{G} = \{(\Box, Suc), (\%, Suc), (\blacktriangle, Pred)\}$$

Se tiene tambien por ejemplo que $\mathcal{G}(\%) = Suc$ por lo cual tambien es cierto que $\mathcal{G}(\%)(22) = 23$, etc.

E$_2$ Si Σ es un alfabeto no vacio, la funcion

$$\begin{array}{rcl} \mathcal{G} : \Sigma & \to & \{f : f \text{ es una funcion de } \Sigma^* \text{ en } \Sigma^*\} \\ a & \to & d_a \end{array}$$

es una familia Σ-indexada de funciones. Notar que

$$\mathcal{G} = \{(a, d_a) : a \in \Sigma\}$$

E$_3$ Sea $\Sigma = \{\Box, \%, \blacktriangle\}$. Entonces $\{(\Box, Suc), (\%, p_3^{2,4}), (\blacktriangle, \emptyset)\}$ es una familia $\{\Box, \%, \blacktriangle\}$-indexada de funciones.

NOTACION: Si $\mathcal{G}$ es una familia Σ-indexada de funciones, entonces para $a \in \Sigma$, escribiremos $\mathcal{G}_a$ en lugar de $\mathcal{G}(a)$.

Ahora si podemos dar la definicion matematica precisa del primero de los dos casos de recursion primitiva sobre parametro alfabetico. Sea

$$f : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \omega$$

con $S_1, ..., S_n \subseteq \omega$ y $L_1, ..., L_m \subseteq \Sigma^*$ conjuntos no vacios y sea $\mathcal{G}$ una familia Σ-indexada de funciones tal que

$$\mathcal{G}_a : \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \to \omega$$

para cada $a \in \Sigma$. Definamos

$$R(f, \mathcal{G}) : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \to \omega$$

de la siguiente manera

(1) $R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \varepsilon) = f(\vec{x}, \vec{\alpha})$

(2) $R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \alpha a) = \mathcal{G}_a(R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \alpha), \vec{x}, \vec{\alpha}, \alpha)$

Diremos que $R(f, \mathcal{G})$ es obtenida por *recursion primitiva* a partir de f y $\mathcal{G}$.

Notese que cuando $n = m = 0$, se tiene que $D_f = \{\Diamond\}$ y (1) y (2) se transforman en

(1) $R(f, \mathcal{G})(\varepsilon) = f(\Diamond)$

(2) $R(f, \mathcal{G})(\alpha a) = \mathcal{G}_a(R(f, \mathcal{G})(\alpha), \alpha)$

Ejercicio 20: Sea $\Sigma = \{\%, @, ?\}$. Encuentre f y $\mathcal{G}$ tales que $\lambda\alpha[|\alpha|] = R(f, \mathcal{G})$

Ejercicio 21: Sea $\Sigma = \{\%, @, ?\}$. Encuentre f y $\mathcal{G}$ tales que $\lambda\alpha_1\alpha[|\alpha_1| + |\alpha|_@] = R(f, \mathcal{G})$

Ejercicio 22: V o F o I, justifique.

(a) Sea $\Sigma = \{@, \&\}$. Se tiene que $\lambda\alpha\,[|\alpha|] = R\left(C_0^{0,0}, \{Suc \circ p_1^{1,1}, Suc \circ p_1^{1,1}\}\right)$

(b) $R\left(p_1^{2,0}, \{(@, p_1^{3,1}), (\&, p_2^{3,1})\}\right) = p_1^{2,1}$

Lemma 6 *Si f y cada $\mathcal{G}_a$ son Σ-efectivamente computables, entonces $R(f, \mathcal{G})$ lo es.*

Ejercicio 23: Haga la prueba del lema anterior para el caso $\Sigma = \{@, \blacktriangle\}$

Recursion primitiva sobre variable alfabetica con valores alfabeticos

Supongamos Σ es un alfabeto finito. Sea

$$f : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \rightarrow \Sigma^*$$

con $S_1, ..., S_n \subseteq \omega$ y $L_1, ..., L_m \subseteq \Sigma^*$ conjuntos no vacios y sea $\mathcal{G}$ una familia Σ-indexada de funciones tal que

$$\mathcal{G}_a : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \times \Sigma^* \rightarrow \Sigma^*$$

para cada $a \in \Sigma$. Definamos

$$R(f, \mathcal{G}) : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \rightarrow \Sigma^*$$

de la siguiente manera

(1) $R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \varepsilon) = f(\vec{x}, \vec{\alpha})$

(2) $R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \alpha a) = \mathcal{G}_a(\vec{x}, \vec{\alpha}, \alpha, R(f, \mathcal{G})(\vec{x}, \vec{\alpha}, \alpha))$.

Diremos que $R(f, \mathcal{G})$ es obtenida por *recursion primitiva* a partir de f y $\mathcal{G}$. Notese que cuando $n = m = 0$, se tiene que $D_f = \{\Diamond\}$ y (1) y (2) se transforman en

(1) $R(f, \mathcal{G})(\varepsilon) = f(\Diamond)$

(2) $R(f, \mathcal{G})(\alpha a) = \mathcal{G}_a(\alpha, R(f, \mathcal{G})(\alpha))$

Ejercicio 24: Encuentre f y $\mathcal{G}$ tales que $\lambda \alpha_1 \alpha [\alpha_1 \alpha] = R(f, \mathcal{G})$

Ejercicio 25: Sea $\Sigma = \{\triangle, \blacktriangle\}$. Diga que funcion conocida es $R(C_\varepsilon^{0,1}, \mathcal{G})$, donde $\mathcal{G}$ es dada por $\mathcal{G}_\triangle = d_\triangle \circ p_3^{0,3}$ y $\mathcal{G}_\blacktriangle = d_\blacktriangle \circ p_3^{0,3}$

Sea Σ un alfabeto finito. Dada $\gamma \in \Sigma^*$, definamos

$$\gamma^R = \begin{cases} [\gamma]_{|\gamma|}[\gamma]_{|\gamma|-1}...[\gamma]_1 & \text{si } |\gamma| \geq 1 \\ \varepsilon & \text{caso contrario} \end{cases}$$

La palabra γ^R es llamada la *resiproca* de γ. Para $a \in \Sigma$, definamos la funcion

$$\begin{array}{rcl} I_a : \Sigma^* & \rightarrow & \Sigma^* \\ \alpha & \rightarrow & a\alpha \end{array}$$

Recordemos que $\alpha^0 = \varepsilon$, para cada $\alpha \in \Sigma^*$, por lo cual tenemos que $D_{\lambda x \alpha [\alpha^x]} = \omega \times \Sigma^*$.

Ejercicio 26: Sea $\Sigma = \{\triangle, \blacktriangle\}$. Explique la forma en la que aplicando los constructores de composicion y recursion primitiva a las funciones del conjunto inicial se pueden obtener las siguientes funciones

(a) I_a

(b) $\lambda\alpha[\alpha^R]$

(c) $\lambda t\alpha[\alpha^t]$

Ejercicio 27: V o F o I, justifique.

(a) Sea $\Sigma = \{\triangle, \blacktriangle\}$. Entonces $R(p_1^{0,1}, \{(\triangle, p_3^{0,3}), (\blacktriangle, d_\blacktriangle \circ p_3^{0,3})\})(\triangle\blacktriangle, \triangle\blacktriangle) = \triangle\blacktriangle\blacktriangle\blacktriangle$

(b) $R(p_1^{0,1}, d_\alpha \circ p_3^{0,3}) = \lambda\alpha_1\alpha[\alpha_1\alpha]$

La prueba del siguiente lema es completamente analoga a la del lema anterior que fue dejada como ejercicio.

Lemma 7 *Si f y cada $\mathcal{G}_a$ son Σ-efectivamente computables, entonces $R(f, \mathcal{G})$ lo es.*

Funciones Σ-recursivas primitivas

Intuitivamente hablando ya sabemos que una funcion es Σ-recursiva primitiva si se puede obtener de las iniciales usando los constructores de composicion y recursion primitiva. Daremos ahora una definicion matematica de este concepto. Definamos los conjuntos $\mathrm{PR}_0^\Sigma \subseteq \mathrm{PR}_1^\Sigma \subseteq \mathrm{PR}_2^\Sigma \subseteq ... \subseteq \mathrm{PR}^\Sigma$ de la siguiente manera

$$\begin{array}{lcl}
\mathrm{PR}_0^\Sigma & = & \left\{Suc, Pred, C_0^{0,0}, C_\varepsilon^{0,0}\right\} \cup \{d_a : a \in \Sigma\} \cup \left\{p_j^{n,m} : 1 \leq j \leq n+m\right\} \\
\mathrm{PR}_{k+1}^\Sigma & = & \mathrm{PR}_k^\Sigma \cup \left\{f \circ [f_1, ..., f_r] : f, f_1, ..., f_r \in \mathrm{PR}_k^\Sigma,\ r \geq 1\right\} \cup \\
 & & \left\{R(f, \mathcal{G}) : f \text{ y cada } \mathcal{G}_a \text{ pertenecen a } \mathrm{PR}_k^\Sigma\right\} \cup \\
 & & \left\{R(f, g) : f, g \in \mathrm{PR}_k^\Sigma\right\} \\
\mathrm{PR}^\Sigma & = & \bigcup_{k \geq 0} \mathrm{PR}_k^\Sigma
\end{array}$$

Una funcion es llamada Σ-*recursiva primitiva* (Σ-*p.r.*) si pertenece a PR^Σ.

Proposition 8 *Si $f \in \mathrm{PR}^\Sigma$, entonces f es Σ-efectivamente computable.*

Ejercicio 28: Explique con palabras por que es cierta la proposicion anterior

Algunas funciones Σ-recursivas primitivas

En los siguientes cuatro lemas se prueba bien formalmente que varias funciones bien conocidas son Σ-primitivas recursivas. La mayoria de estas funciones ya fueron obtenidas usando los constructores de composicion y recursion primitiva, en los desarrollos anteriores o en los ejercicios.

Lemma 9 *Sea Σ un alfabeto finito.*

(1) $\emptyset \in \mathrm{PR}^{\Sigma}$.

(2) $\lambda xy\,[x+y] \in \mathrm{PR}^{\Sigma}$.

(3) $\lambda xy\,[x.y] \in \mathrm{PR}^{\Sigma}$.

(4) $\lambda x\,[x!] \in \mathrm{PR}^{\Sigma}$.

Proof. (1) Notese que $\emptyset = Pred \circ C_0^{0,0} \in \mathrm{PR}_1^{\Sigma}$

(2) Notar que

$$\begin{aligned}
\lambda xy\,[x+y]\,(0,x_1) &= x_1 = p_1^{1,0}(x_1) \\
\lambda xy\,[x+y]\,(t+1,x_1) &= \lambda xy\,[x+y]\,(t,x_1)+1 \\
&= \left(Suc \circ p_1^{3,0}\right)(\lambda xy\,[x+y]\,(t,x_1),t,x_1)
\end{aligned}$$

lo cual implica que $\lambda xy\,[x+y] = R\left(p_1^{1,0}, Suc \circ p_1^{3,0}\right) \in \mathrm{PR}_2^{\Sigma}$.

(3) Primero note que

$$\begin{aligned}
C_0^{1,0}(0) &= C_0^{0,0}(\Diamond) \\
C_0^{1,0}(t+1) &= C_0^{1,0}(t)
\end{aligned}$$

lo cual implica que $C_0^{1,0} = R\left(C_0^{0,0}, p_1^{2,0}\right) \in \mathrm{PR}_1^{\Sigma}$. Tambien note que

$$\lambda tx\,[t.x] = R\left(C_0^{1,0}, \lambda xy\,[x+y] \circ \left[p_1^{3,0}, p_3^{3,0}\right]\right),$$

lo cual por (2) implica que $\lambda tx\,[t.x] \in \mathrm{PR}_4^{\Sigma}$.

(4) Note que

$$\begin{aligned}
\lambda x\,[x!]\,(0) &= 1 = C_1^{0,0}(\Diamond) \\
\lambda x\,[x!]\,(t+1) &= \lambda x\,[x!]\,(t).(t+1),
\end{aligned}$$

lo cual implica que

$$\lambda x\,[x!] = R\left(C_1^{0,0}, \lambda xy\,[x.y] \circ \left[p_1^{2,0}, Suc \circ p_2^{2,0}\right]\right).$$

Ya que $C_1^{0,0} = Suc \circ C_0^{0,0}$, tenemos que $C_1^{0,0} \in \mathrm{PR}_1^\Sigma$. Por (3), tenemos que

$$\lambda xy\,[x.y] \circ \left[p_1^{2,0}, Suc \circ p_2^{2,0}\right] \in \mathrm{PR}_5^\Sigma,$$

obteniendo que $\lambda x\,[x!] \in \mathrm{PR}_6^\Sigma$. ■

Ahora consideraremos dos funciones las cuales son obtenidas naturalmente por recursion primitiva sobre variable alfabetica.

Lemma 10 *Supongamos Σ es un alfabeto finito.*

(a) $\lambda\alpha\beta\,[\alpha\beta] \in \mathrm{PR}^\Sigma$

(b) $\lambda\alpha\,[|\alpha|] \in \mathrm{PR}^\Sigma$

Proof. (a) Ya que

$$\begin{aligned} \lambda\alpha\beta\,[\alpha\beta]\,(\alpha_1,\varepsilon) &= \alpha_1 = p_1^{0,1}(\alpha_1) \\ \lambda\alpha\beta\,[\alpha\beta]\,(\alpha_1,\alpha a) &= d_a(\lambda\alpha\beta\,[\alpha\beta]\,(\alpha_1,\alpha)), a \in \Sigma \end{aligned}$$

tenemos que $\lambda\alpha\beta\,[\alpha\beta] = R\left(p_1^{0,1}, \mathcal{G}\right)$, donde $\mathcal{G}_a = d_a \circ p_3^{0,3}$, para cada $a \in \Sigma$.

(b) Ya que

$$\begin{aligned} \lambda\alpha\,[|\alpha|]\,(\varepsilon) &= 0 = C_0^{0,0}(\Diamond) \\ \lambda\alpha\,[|\alpha|]\,(\alpha a) &= \lambda\alpha\,[|\alpha|]\,(\alpha) + 1 \end{aligned}$$

tenemos que $\lambda\alpha\,[|\alpha|] = R\left(C_0^{0,0}, \mathcal{G}\right)$, donde $\mathcal{G}_a = Suc \circ p_1^{1,1}$, para cada $a \in \Sigma$..
■

Lemma 11 *Sea Σ un alfabeto finito. Entonces $C_k^{n,m}, C_\alpha^{n,m} \in \mathrm{PR}^\Sigma$, para cada $n,m,k \geq 0$ y $\alpha \in \Sigma^*$.*

Proof. (a) Note que $C_{k+1}^{0,0} = Suc \circ C_k^{0,0}$, lo cual implica $C_k^{0,0} \in \mathrm{PR}_k^\Sigma$, para $k \geq 0$. Tambien note que $C_{\alpha a}^{0,0} = d_a \circ C_\alpha^{0,0}$, lo cual dice que $C_\alpha^{0,0} \in \mathrm{PR}^\Sigma$, para $\alpha \in \Sigma^*$. Para ver que $C_k^{0,1} \in \mathrm{PR}^\Sigma$ notar que

$$\begin{aligned} C_k^{0,1}(\varepsilon) &= k = C_k^{0,0}(\Diamond) \\ C_k^{0,1}(\alpha a) &= C_k^{0,1}(\alpha) = p_1^{1,1}\left(C_k^{0,1}(\alpha), \alpha\right) \end{aligned}$$

lo cual implica que $C_k^{0,1} = R\left(C_k^{0,0}, \mathcal{G}\right)$, con $\mathcal{G}_a = p_1^{1,1}$, $a \in \Sigma$. En forma similar podemos ver que $C_k^{1,0}, C_\alpha^{1,0}, C_\alpha^{0,1} \in \mathrm{PR}^\Sigma$. Supongamos ahora que $m > 0$. Entonces

$$\begin{aligned} C_k^{n,m} &= C_k^{0,1} \circ p_{n+1}^{n,m} \\ C_\alpha^{n,m} &= C_\alpha^{0,1} \circ p_{n+1}^{n,m} \end{aligned}$$

de lo cual obtenemos que $C_k^{n,m}, C_\alpha^{n,m} \in \mathrm{PR}^\Sigma$. El caso $n > 0$ es similar. ■

Lemma 12 *Sea Σ un alfabeto finito.*

(a) $\lambda xy\,[x^y] \in \mathrm{PR}^\Sigma$.

(b) $\lambda t\alpha\,[\alpha^t] \in \mathrm{PR}^\Sigma$.

Proof. (a) Note que

$$\lambda tx\left[x^t\right] = R\left(C_1^{1,0}, \lambda xy\,[x.y] \circ \left[p_1^{3,0}, p_3^{3,0}\right]\right) \in \mathrm{PR}^\Sigma.$$

O sea que $\lambda xy\,[x^y] = \lambda tx\,[x^t] \circ \left[p_2^{2,0}, p_1^{2,0}\right] \in \mathrm{PR}^\Sigma$.

(b) Note que

$$\lambda t\alpha\left[\alpha^t\right] = R\left(C_\varepsilon^{0,1}, \lambda\alpha\beta\,[\alpha\beta] \circ \left[p_3^{1,2}, p_2^{1,2}\right]\right) \in \mathrm{PR}^\Sigma.$$

■

Ejercicio 29: Si $\leq$ es un orden total sobre un alfabeto no vacio Σ, entonces $s^{\leq}$, $\#^{\leq}$ y $*^{\leq}$ pertenecen a PR^Σ

Ejercicio 30: Sea $\Sigma = \{\$, ?, @, \forall, \rightarrow, (\}$ y sea $S = \{\$, ?\}^*$. Pruebe que

$$\begin{array}{rcl} \chi_S^{\Sigma^*} : \Sigma^* & \rightarrow & \omega \\ \alpha & \rightarrow & \left\{ \begin{array}{ll} 1 & \text{si } \alpha \in S \\ 0 & \text{si } \alpha \notin S \end{array} \right. \end{array}$$

es Σ-p.r.

Dados $x, y \in \omega$, definamos

$$x \dot{-} y = \max(x - y, 0).$$

Lemma 13 *(a)* $\lambda xy\left[x \dot{-} y\right] \in \mathrm{PR}^\Sigma$

(b) $\lambda xy\,[\max(x, y)] \in \mathrm{PR}^\Sigma$

(c) $\lambda xy\,[x = y] \in \mathrm{PR}^\Sigma$

(d) $\lambda xy\,[x \leq y] \in \mathrm{PR}^\Sigma$

(e) $\lambda\alpha\beta\,[\alpha = \beta] \in \mathrm{PR}^\Sigma$

Proof. (a) Primero notar que $\lambda x\left[x \dot{-} 1\right] = R\left(C_0^{0,0}, p_2^{2,0}\right) \in \mathrm{PR}^\Sigma$. Tambien note que

$$\lambda tx\left[x \dot{-} t\right] = R\left(p_1^{1,0}, \lambda x\left[x \dot{-} 1\right] \circ p_1^{3,0}\right) \in \mathrm{PR}^\Sigma.$$

O sea que $\lambda xy\left[x \dot{-} y\right] = \lambda tx\left[x \dot{-} t\right] \circ \left[p_2^{2,0}, p_1^{2,0}\right] \in \mathrm{PR}^\Sigma$.

(b) Note que $\lambda xy\left[\max(x,y)\right] = \lambda xy\left[x + (y \dot{-} x)\right]$.
(c) Note que $\lambda xy\left[x = y\right] = \lambda xy\left[1 \dot{-} ((x \dot{-} y) + (y \dot{-} x))\right]$.
(d) Note que $\lambda xy\left[x \leq y\right] = \lambda xy\left[1 \dot{-} (x \dot{-} y)\right]$.
(e) Sea $\leq$ un orden total sobre Σ. Ya que

$$\alpha = \beta \text{ sii } \#^{\leq}(\alpha) = \#^{\leq}(\beta)$$

tenemos que

$$\lambda\alpha\beta\left[\alpha = \beta\right] = \lambda xy\left[x = y\right] \circ \left[\#^{\leq} \circ p_1^{0,2}, \#^{\leq} \circ p_2^{0,2}\right]$$

lo cual nos dice que $\lambda\alpha\beta\left[\alpha = \beta\right]$ es Σ-p.r. ■

Ejercicio 31: Complete las pruebas de (b),(c),(d) y (e) del lema anterior

Ejercicio 32: Sea Σ un alfabeto finito.

(a) $\lambda x[x \text{ es par}]$ es Σ-p.r..
(b) $\lambda xyz\alpha\beta\gamma[x.y + \max(x, |\alpha|)^{|\beta|}]$ es Σ-p.r.
(c) $\lambda x\alpha\left[x = |\alpha|\right]$ es Σ-p.r..
(d) $\lambda xy\alpha\beta\left[\alpha^x = \beta\right]$ es Σ-p.r..

Operaciones logicas entre predicados

Dados predicados $P : S \subseteq \omega^n \times \Sigma^{*m} \to \omega$ y $Q : S \subseteq \omega^n \times \Sigma^{*m} \to \omega$, con el mismo dominio, definamos nuevos predicados $(P \vee Q)$, $(P \wedge Q)$ y $\neg P$ de la siguiente manera

$$\begin{array}{rcl} (P \vee Q) : S & \to & \omega \\ (\vec{x}, \vec{\alpha}) & \to & \left\{ \begin{array}{ll} 1 & \text{si } P(\vec{x}, \vec{\alpha}) = 1 \text{ o } Q(\vec{x}, \vec{\alpha}) = 1 \\ 0 & \text{caso contrario} \end{array} \right. \\ (P \wedge Q) : S & \to & \omega \\ (\vec{x}, \vec{\alpha}) & \to & \left\{ \begin{array}{ll} 1 & \text{si } P(\vec{x}, \vec{\alpha}) = 1 \text{ y } Q(\vec{x}, \vec{\alpha}) = 1 \\ 0 & \text{caso contrario} \end{array} \right. \end{array}$$

$$\begin{array}{rcl} \neg P : S & \to & \omega \\ (\vec{x}, \vec{\alpha}) & \to & \left\{ \begin{array}{ll} 1 & \text{si } P(\vec{x}, \vec{\alpha}) = 0 \\ 0 & \text{si } P(\vec{x}, \vec{\alpha}) = 1 \end{array} \right. \end{array}$$

Lemma 14 *Si $P : S \subseteq \omega^n \times \Sigma^{*m} \to \omega$ y $Q : S \subseteq \omega^n \times \Sigma^{*m} \to \omega$ son predicados Σ-p.r., entonces $(P \vee Q)$, $(P \wedge Q)$ y $\neg P$ lo son tambien.*

Proof. Note que

$$\begin{aligned}\neg P &= \lambda xy\left[x\dot{-}y\right]\circ[C_1^{n,m},P]\\(P\wedge Q) &= \lambda xy\,[x.y]\circ[P,Q]\\(P\vee Q) &= \neg(\neg P\wedge\neg Q)\end{aligned}$$

∎

Ejercicio 33: V o F o I, justifique.

(a) Si P_1, P_2, P_3 son predicados Σ-p.r. y $D_{P_1} = D_{P_2} = D_{P_3}$, entonces el predicado $(P_1 \vee P_2 \wedge P_3)$ es Σ-p.r.

(b) Sea Σ un alfabeto finito. Entonces $\lambda x\alpha\beta\,[x = |\alpha| \wedge \alpha = \beta] = (\lambda x\alpha\,[x = |\alpha|] \wedge \lambda\alpha\beta\,[\alpha = \beta])$

(c) Sea Σ un alfabeto finito. Entonces $(\lambda x\,[x = 1] \wedge \lambda\alpha\,[\alpha = \varepsilon])(2, \varepsilon) = 0$

(d) Si $S, T \subseteq \omega$, entonces $\chi^{\omega\times\omega}_{S\times T} = (\chi^{\omega}_{S} \wedge \chi^{\omega}_{T})$

Conjuntos Σ-recursivos primitivos

Un conjunto Σ-mixto $S \subseteq \omega^n \times \Sigma^{*m}$ es llamado Σ-*recursivo primitivo* si su funcion caracteristica $\chi_S^{\omega^n\times\Sigma^{*m}}$ es Σ-p.r.. Notese que $\chi_S^{\omega^n\times\Sigma^{*m}} = \lambda\vec{x}\vec{\alpha}[(\vec{x},\vec{\alpha}) \in S]$.

Ejercicio 34: Sea $\Sigma = \{@, !\}$. Pruebe que los siguientes conjuntos son Σ-p.r.

(a) ω

(b) Σ^*

(c) $\{(x, y) \in \omega^2 : x = y\}$

(d) $\{(x, \alpha) \in \omega \times \Sigma^* : x = |\alpha|\}$

(e) $\{x \in \omega : x \text{ es par}\}$

(f) $\{(x, y, \alpha, \beta, \gamma) \in \omega^2 \times \Sigma^{*3} : x \leq |\gamma|\}$

Lemma 15 *Si $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ son Σ-p.r., entonces $S_1 \cup S_2$, $S_1 \cap S_2$ y $S_1 - S_2$ lo son.*

Proof. Note que

$$\begin{aligned}\chi^{\omega^n\times\Sigma^{*m}}_{S_1\cup S_2} &= (\chi^{\omega^n\times\Sigma^{*m}}_{S_1} \vee \chi^{\omega^n\times\Sigma^{*m}}_{S_2})\\\chi^{\omega^n\times\Sigma^{*m}}_{S_1\cap S_2} &= (\chi^{\omega^n\times\Sigma^{*m}}_{S_1} \wedge \chi^{\omega^n\times\Sigma^{*m}}_{S_2})\\\chi^{\omega^n\times\Sigma^{*m}}_{S_1- S_2} &= \lambda xy\left[x\dot{-}y\right]\circ\left[\chi^{\omega^n\times\Sigma^{*m}}_{S_1}, \chi^{\omega^n\times\Sigma^{*m}}_{S_2}\right]\end{aligned}$$

∎

Ejercicio 35: Si $S \subseteq \omega^n \times \Sigma^{*m}$ es finito, entonces S es Σ-p.r. Haga el caso $n = m = 1$. (Hint: haga el caso en que S tiene un solo elemento y luego aplique el lema anterior).

Ejercicio 36: Sea Σ un alfabeto finito.

(a) Pruebe que Σ es Σ-p.r.

(b) Pruebe que $\Sigma^* - \{\varepsilon\}$ es Σ-p.r.

(c) Pruebe que $\Sigma^* - (\{\varepsilon\} \cup \Sigma)$ es Σ-p.r.

(d) Pruebe que $\omega - \{0, 1\}$ es Σ-p.r.

(e) Pruebe que $\{(x, y, \alpha, \beta, \gamma) \in \omega^2 \times \Sigma^{*3} : \alpha \neq \varepsilon \vee x \leq |\gamma|\}$ es Σ-p.r.

(f) Pruebe que $\{(x, \alpha, \beta) : |\alpha| > 6\}$ es Σ-p.r.

El siguiente lema caracteriza cuando un conjunto rectangular es Σ-p.r..

Lemma 16 *Supongamos $S_1, ..., S_n \subseteq \omega$, $L_1, ..., L_m \subseteq \Sigma^*$ son conjuntos no vacios. Entonces $S_1 \times ... \times S_n \times L_1 \times ... \times L_m$ es Σ-p.r. sii $S_1, ..., S_n, L_1, ..., L_m$ son Σ-p.r.*

Ejercicio 37: (S) Haga la prueba del lema anterior para el caso de $n = m = 1$ (en el apunte esta la prueba general)

Dada una funcion f y un conjunto $S \subseteq D_f$, usaremos $f|_S$ para denotar la *restriccion* de f al conjunto S, i.e. $f|_S = f \cap (S \times I_f)$. Notese que $f|_S$ es la funcion dada por

$$\begin{aligned} D_{f|_S} &= S \\ f|_S(e) &= f(e), \text{ para cada } e \in S \end{aligned}$$

Notese que cualesquiera sea la funcion f tenemos que $f|_\emptyset = \emptyset$ y $f|_{D_f} = f$.

Lemma 17 *Supongamos $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ es Σ-p.r., donde $O \in \{\omega, \Sigma^*\}$. Si $S \subseteq D_f$ es Σ-p.r., entonces $f|_S$ es Σ-p.r..*

Proof. Supongamos $O = \Sigma^*$. Entonces

$$f|_S = \lambda x\alpha\, [\alpha^x] \circ \left[Suc \circ Pred \circ \chi_S^{\omega^n \times \Sigma^{*m}}, f \right]$$

lo cual nos dice que $f|_S$ es Σ-p.r.. El caso $O = \omega$ es similar usando $\lambda xy\, [x^y]$ en lugar de $\lambda x\alpha\, [\alpha^x]$. ■

Usando el lema anterior en combinacion con el Lema 14 podemos ver que muchos predicados usuales son Σ-p.r.. Por ejemplo sea

$$P = \lambda x\alpha\beta\gamma \left[x = |\gamma| \wedge \alpha = \gamma^{Pred(|\beta|)}\right].$$

Notese que

$$D_P = \omega \times \Sigma^* \times (\Sigma^* - \{\varepsilon\}) \times \Sigma^*$$

Ademas D_P es Σ-p.r. ya que

$$\chi_{D_P}^{\omega\times\Sigma^{*3}} = \neg\lambda\alpha\beta\left[\alpha = \beta\right] \circ \left[p_3^{1,3}, C_\varepsilon^{1,3}\right]$$

Tambien note que los predicados

$$P_1 = \lambda x\alpha\beta\gamma\left[x = |\gamma|\right]$$
$$P_2 = \lambda x\alpha\beta\gamma\left[\alpha = \gamma^{Pred(|\beta|)}\right]$$

son Σ-p.r. ya que pueden obtenerse componiendo funciones Σ-p.r.. Un error seria pensar que $P = (P_1 \wedge P_2)$ ya que P_1 y P_2 tienen dominios distintos por lo cual no esta definido $(P_1 \wedge P_2)$. Sin envargo tenemos que $P = (P_1|_{D_P} \wedge P_2)$, lo cual nos dice que P es Σ-p.r. ya que $P_1|_{D_P}$ es Σ-p.r. por el Lema 17 y por lo tanto podemos aplicar el Lema 14

Ejercicio 38: Sea $\Sigma = \{@, !\}$. Sea $P = \lambda xy\alpha\beta\gamma\left[Pred(Pred(|\beta|)) \neq 6 \wedge \alpha^x = \beta\right]$

(a) Encuentre por definicion de notacion lambda el dominio de P

(b) Pruebe que P es Σ-p.r.

Aceptaremos sin prueba el siguiente resultado (ver el apunte por una prueba)

Lemma 18 *Sean $O \in \{\omega, \Sigma^*\}$ y $n, m \in \omega$. Si $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ es Σ-p.r., entonces existe una funcion Σ-p.r. $\bar{f} : \omega^n \times \Sigma^{*m} \to O$, tal que $f = \bar{f}|_{D_f}$.*

Ahora podemos probar una proposicion muy importante.

Proposition 19 *Un conjunto S es Σ-p.r. sii S es el dominio de alguna funcion Σ-p.r..*

Proof. Supongamos que $S \subseteq \omega^n \times \Sigma^{*m}$.

($\Rightarrow$) Note que $S = D_{Pred\circ\chi_S^{\omega^n\times\Sigma^{*m}}}$.

($\Leftarrow$) Probaremos por induccion en k que D_F es Σ-p.r., para cada $F \in \mathrm{PR}_k^\Sigma$. El caso $k = 0$ es facil. Supongamos el resultado vale para un k fijo y supongamos

$F \in \mathrm{PR}^{\Sigma}_{k+1}$. Veremos entonces que D_F es Σ-p.r.. Hay varios casos. Consideremos primero el caso en que $F = R(f,g)$, donde

$$f : S_1 \times ... \times S_n \times L_1 \times ... \times L_m \to \Sigma^*$$
$$g : \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m \times \Sigma^* \to \Sigma^*,$$

con $S_1, ..., S_n \subseteq \omega$ y $L_1, ..., L_m \subseteq \Sigma^*$ conjuntos no vacios y $f, g \in \mathrm{PR}^{\Sigma}_{k}$. Notese que por definicion de $R(f,g)$, tenemos que

$$D_F = \omega \times S_1 \times ... \times S_n \times L_1 \times ... \times L_m.$$

Por hipotesis inductiva tenemos que $D_f = S_1 \times ... \times S_n \times L_1 \times ... \times L_m$ es Σ-p.r., lo cual por el Lema 16 nos dice que los conjuntos $S_1, ..., S_n$, $L_1, ..., L_m$ son Σ-p.r.. Ya que ω es Σ-p.r., el Lema 16 nos dice que D_F es Σ-p.r..

Los otros casos de recursion primitiva son dejados al lector.

Supongamos ahora que $F = g \circ [g_1, ..., g_r]$ con $g, g_1, ..., g_r \in \mathrm{PR}^{\Sigma}_{k}$. Si $F = \emptyset$, entonces es claro que $D_F = \emptyset$ es Σ-p.r.. Supongamos entonces que F no es la funcion $\emptyset$. Tenemos entonces que r es de la forma $n + m$ y

$$g : D_g \subseteq \omega^n \times \Sigma^{*m} \to O$$
$$g_i : D_{g_i} \subseteq \omega^k \times \Sigma^{*l} \to \omega,\ i = 1, ..., n$$
$$g_i : D_{g_i} \subseteq \omega^k \times \Sigma^{*l} \to \Sigma^*, i = n+1, ..., n+m$$

con $O \in \{\omega, \Sigma^*\}$ y $k, l \in \omega$. Por Lema 18, hay funciones Σ-p.r. $\bar{g}_1, ..., \bar{g}_{n+m}$ las cuales son Σ-totales y cumplen

$$g_i = \bar{g}_i|_{D_{g_i}}, \text{ para } i = 1, ..., n+m.$$

Por hipotesis inductiva los conjuntos D_g, D_{g_i}, $i = 1, ..., n+m$, son Σ-p.r. y por lo tanto

$$S = \bigcap_{i=1}^{n+m} D_{g_i}$$

lo es. Notese que

$$\chi^{\omega^k \times \Sigma^{*l}}_{D_F} = (\chi^{\omega^n \times \Sigma^{*m}}_{D_g} \circ [\bar{g}_1, ..., \bar{g}_{n+m}] \wedge \chi^{\omega^k \times \Sigma^{*l}}_{S})$$

lo cual nos dice que D_F es Σ-p.r.. ■

Lema de division por casos para funciones Σ-p.r.

Una observacion interesante es que si $f_i : D_{f_i} \to O$, $i = 1, ..., k$, son funciones tales que $D_{f_i} \cap D_{f_j} = \emptyset$ para $i \neq j$, entonces $f_1 \cup ... \cup f_k$ es la funcion

$$\begin{array}{rcl} D_{f_1} \cup ... \cup D_{f_k} & \to & O \\ e & \to & \left\{ \begin{array}{ll} f_1(e) & \text{si } e \in D_{f_1} \\ \vdots & \vdots \\ f_k(e) & \text{si } e \in D_{f_k} \end{array} \right. \end{array}$$

Lemma 20 *Sean $O \in \{\omega, \Sigma^*\}$ y $n, m \in \omega$. Supongamos $f_i : D_{f_i} \subseteq \omega^n \times \Sigma^{*m} \to O$, $i = 1, ..., k$, son funciones Σ-p.r. tales que $D_{f_i} \cap D_{f_j} = \emptyset$ para $i \neq j$. Entonces $f_1 \cup ... \cup f_k$ es Σ-p.r..*

Proof. Supongamos $O = \Sigma^*$ y $k = 2$. Sean

$$\bar{f}_i : \omega^n \times \Sigma^{*m} \to \Sigma^*, i = 1, 2,$$

funciones Σ-p.r. tales que $\bar{f}_i|_{D_{f_i}} = f_i$, $i = 1, 2$ (Lema 18). Por Lema 19 los conjuntos D_{f_1} y D_{f_2} son Σ-p.r. y por lo tanto lo es $D_{f_1} \cup D_{f_2}$. Ya que

$$f_1 \cup f_2 = \left(\lambda\alpha\beta\left[\alpha\beta\right] \circ \left[\lambda x\alpha\left[\alpha^x\right] \circ \left[\chi_{D_{f_1}}^{\omega^n \times \Sigma^{*m}}, \bar{f}_1\right], \lambda x\alpha\left[\alpha^x\right] \circ \left[\chi_{D_{f_2}}^{\omega^n \times \Sigma^{*m}}, \bar{f}_2\right]\right]\right)|_{D_{f_1} \cup D_{f_2}}$$

tenemos que $f_1 \cup f_2$ es Σ-p.r..

El caso $k > 2$ puede probarse por induccion ya que

$$f_1 \cup ... \cup f_k = (f_1 \cup ... \cup f_{k-1}) \cup f_k.$$

■

CONSEJO IMPORTANTE: Si uno quiere usar el lema de division por casos para probar que una funcion f es Σ-p.r., entonces lo primero que hay que hacer, antes de ver que algo sea Σ-p.r. o no, es (a lo mariposa) definir correctamente funciones $f_1, ..., f_k$ tales que $D_{f_i} \cap D_{f_j} = \emptyset$ para $i \neq j$ y ademas $f_1 \cup ... \cup f_k = f$. Consejos para encontrar dichas funciones:

1. Determinar el k, es decir, k es justamente la cantidad de "casos" en la descripcion de f
2. Para cada "caso" de la descripcion de f, asociar un subconjunto del dominio de f el cual sea justamente definido por la propiedad correspondiente a ese caso. Ojo que dijimos subconjunto de D_f, no confundir los tipos!! (a veces los casos se describen usando no todas las variables de las cuales depende la funcion)
3. Notar que los subconjuntos $S_1, ..., S_k$ asi definidos deben ser disjuntos de a pares y unidos deben dar el dominio de f
4. Para cada i defina f_i de la siguiente manera:
   (a) dominio de $f_i = S_i$
   (b) regla de f_i dada por la regla que describe f para el caso i-esimo
5. En general suele suceder que f_i es la restriccion a S_i de una funcion con dominio mas amplio y se prueba entonces que tanto dicha funcion como S_i son Σ-p.r., resultando asi que f_i es Σ-p.r.

Ejercicio 38,3: Sea $\Sigma = \{@, \$\}$. Sea

$$\begin{array}{rcl} f : \omega \times \Sigma^* & \rightarrow & \omega \\ (x,\alpha) & \rightarrow & \left\{ \begin{array}{ll} |\alpha| .x^2 & \text{si } x + |\alpha| \text{ es impar} \\ 0 & \text{si } x + |\alpha| \text{ es par} \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r..

Ejercicio 38,6: Sea $\Sigma = \{@, \$\}$. Sea

$$\begin{array}{rcl} f : \{10, 11, 17\} \times \Sigma^+ & \rightarrow & \omega \\ (x,\alpha) & \rightarrow & \left\{ \begin{array}{ll} Pred(x) & \text{si } x \text{ es impar} \\ |\alpha| & \text{si } x \text{ es par} \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r..

Ejercicio 39: Sea $\Sigma = \{@, \$\}$. Sea

$$\begin{array}{rcl} f : \mathbf{N} \times \Sigma^+ & \rightarrow & \omega \\ (x,\alpha) & \rightarrow & \left\{ \begin{array}{ll} x^2 & \text{si } x + |\alpha| \text{ es par} \\ 0 & \text{si } x + |\alpha| \text{ es impar} \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r..

Ejercicio 40: Sea $\Sigma = \{@, \$\}$. Sea

$$\begin{array}{rcl} f : \{(x,y,\alpha) : x \leq y\} & \rightarrow & \omega \\ (x,y,\alpha) & \rightarrow & \left\{ \begin{array}{ll} x^2 & \text{si } |\alpha| \leq y \\ 0 & \text{si } |\alpha| > y \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r.. (Explicado en video colgado en granlogico.com)

Ejercicio 41: Sea $\Sigma = \{@, !, \%\}$. Sea

$$\begin{array}{rcl} f : \{1,2,3,4,5\} \times \mathbf{N} \times \{@, \%\}^* & \rightarrow & \Sigma^* \\ (x,y,\alpha) & \rightarrow & \left\{ \begin{array}{ll} \alpha^2 & \text{si } \alpha = @@ \\ !!! & \text{si } \alpha \neq @@ \wedge |\alpha| > y \\ \alpha^{x+y} & \text{si } \alpha \neq @@ \wedge |\alpha| \leq y \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r..

Ejercicio 41,5: Sea $\Sigma = \{@, !, \%\}$. Sea

$$\begin{array}{rcl} f : \mathbf{N} \times \{@, \%\}^* \times \{@, !, \%\}^+ & \rightarrow & \omega \\ (x,\alpha,\beta) & \rightarrow & \left\{ \begin{array}{ll} Pred(|\alpha|) & \text{si } |\alpha| > 2 \text{ y } x \geq 1 \\ |\beta| & \text{si } |\alpha| \leq 2 \text{ o } x = 0 \end{array} \right. \end{array}$$

Pruebe que f es Σ-p.r..

Ejercicio 42: Sea $F : \omega \to \omega$ dada por

$$\begin{aligned}
F(0) &= 2 \\
F(1) &= 2^2 \\
F(2) &= (2^2)^3 \\
F(3) &= ((2^2)^3)^2 \\
F(4) &= (((2^2)^3)^2)^3 \\
&\vdots
\end{aligned}$$

Pruebe que F es Σ-p.r..

Usaremos el lema de division por casos para probar que la funcion $\lambda i\alpha\,[[\alpha]_i]$ es Σ-p.r.. Recordemos que dados $i \in \omega$ y $\alpha \in \Sigma^*$, definimos

$$[\alpha]_i = \begin{cases} i\text{-esimo elemento de } \alpha & \text{si } 1 \leq i \leq |\alpha| \\ \varepsilon & \text{caso contrario} \end{cases}$$

Notese que $D_{\lambda i\alpha[[\alpha]_i]} = \omega \times \Sigma^*$.

Lemma 21 *$\lambda i\alpha\,[[\alpha]_i]$ es Σ-p.r..*

Proof. Supongamos $\Sigma = \{@, !\}$. Note que

$$\begin{aligned}
[\varepsilon]_i &= \varepsilon \\
[\alpha@]_i &= \begin{cases} [\alpha]_i & \text{si } i \neq |\alpha| + 1 \\ @ & \text{si } i = |\alpha| + 1 \end{cases} \\
[\alpha!]_i &= \begin{cases} [\alpha]_i & \text{si } i \neq |\alpha| + 1 \\ ! & \text{si } i = |\alpha| + 1 \end{cases}
\end{aligned}$$

lo cual dice que $\lambda i\alpha\,[[\alpha]_i] = R\left(C_\varepsilon^{1,0}, \mathcal{G}\right)$, donde $\mathcal{G}_a : \omega \times \Sigma^* \times \Sigma^* \to \Sigma^*$ es dada por

$$\mathcal{G}_a(i, \alpha, \zeta) = \begin{cases} \zeta & \text{si } i \neq |\alpha| + 1 \\ a & \text{si } i = |\alpha| + 1 \end{cases}$$

para cada $a \in \Sigma$. O sea que solo resta probar que cada $\mathcal{G}_a$ es Σ-p.r.. Veamos que $\mathcal{G}_@$ es Σ-p.r.. Primero note que los conjuntos

$$\begin{aligned}
S_1 &= \{(i, \alpha, \zeta) \in \omega \times \Sigma^* \times \Sigma^* : i \neq |\alpha| + 1\} \\
S_2 &= \{(i, \alpha, \zeta) \in \omega \times \Sigma^* \times \Sigma^* : i = |\alpha| + 1\}
\end{aligned}$$

son Σ-p.r. ya que

$$\begin{aligned}
\chi_{S_1}^{\omega \times \Sigma^* \times \Sigma^*} &= \lambda xy\,[x \neq y] \circ \left[p_1^{1,2}, Suc \circ \lambda\alpha\,[|\alpha|] \circ p_2^{1,2}\right] \\
\chi_{S_2}^{\omega \times \Sigma^* \times \Sigma^*} &= \lambda xy\,[x = y] \circ \left[p_1^{1,2}, Suc \circ \lambda\alpha\,[|\alpha|] \circ p_2^{1,2}\right]
\end{aligned}$$

Notese que por el Lema 17 tenemos que $p_3^{1,2}|_{S_1}$ y $C_@^{1,2}|_{S_2}$ son Σ-p.r.. Ademas

$$\mathcal{G}_@ = p_3^{1,2}|_{S_1} \cup C_@^{1,2}|_{S_2}$$

por lo cual el Lema 20 nos dice que $\mathcal{G}_@$ es Σ-p.r.. Analogamente se prueba que $\mathcal{G}_!$ es Σ-p.r.. $\blacksquare$