

# Resumen de teoremas para el final de Lenguajes Formales y Computabilidad

Agustín Curto, agucurto95@gmail.com  
Francisco Nievas, frannievas@gmail.com

2017

## Contents

1	Notación y conceptos básicos	2
2	Procedimientos efectivos	4
3	Funciones $\Sigma$ -recursivas	6
4	El lenguaje $S^\Sigma$	16
5	Máquinas de Turing	25

# 1 Notación y conceptos básicos

**Lemma 1.** Sea  $S \subseteq \omega \times \Sigma^*$ , entonces  $S$  es rectangular si y solo si se cumple la siguiente propiedad:

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

**Lemma 2.** La relación  $<$  es un orden total estricto sobre  $\Sigma^*$ .

**Lemma 3.** La función  $s^< : \Sigma^* \rightarrow \Sigma^*$ , definida recursivamente de la siguiente manera:

$$\begin{aligned} s^<(\varepsilon) &= a_1 \\ s^<(\alpha a_i) &= \alpha a_{i+1} \quad \text{para } i < n \\ s^<(\alpha a_n) &= s^<(\alpha) a_1 \end{aligned}$$

tiene la siguiente propiedad:

$$s^<(\alpha) = \min\{\beta \in \Sigma^* : \alpha < \beta\}$$

**Corollary 4.**  $s^<$  es inyectiva.

**Lemma 5.** Se tiene que:

1.  $s^<(\alpha) \neq \varepsilon$ , para cada  $\alpha \in \Sigma^*$ .
2. Si  $\alpha \neq \varepsilon$ , entonces  $\alpha = s^<(\beta)$  para algún  $\beta$ .
3. Si  $S \subseteq \Sigma^* \neq \emptyset$ , entonces  $\exists \alpha \in S$  tal que  $\alpha < \beta$ , para cada  $\beta \in S - \{\alpha\}$ .

**Lemma 6.** Tenemos que:

$$\Sigma^* = \{s^<(0), s^<(1), \dots\}$$

Mas aún la función  $s^<$  es biyectiva.

**Lemma 7.** Sea  $n \geq 1$  fijo, entonces cada  $x \geq 1$  se escribe en forma única de la siguiente manera:

$$x = i_0 n^0 + \dots + i_{k-1} n^{k-1} + i_k n^k$$

con  $k \geq 0$  y  $1 \leq i_0, \dots, i_{k-1}, i_k \leq n$ .

**Lemma 8.** La función  $\#^<$  es biyectiva.

**Lemma 9.** Las funciones  $\#^<$  y  $s^<$  son una inversa de la otra.

**Lemma 10.** Si  $p, p_1, \dots, p_n$  son números primos y  $p$  divide a  $p_1 \dots p_n$ , entonces  $p = p_i$ , para algún  $i$ .

**Theorem 11.** Para cada  $x \in \mathbb{N}$ , hay una única sucesión  $(s_1, s_2, \dots) \in \omega^{[\mathbb{N}]}$  tal que:

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

Notar que  $\prod_{i=1}^{\infty} pr(i)^{s_i}$  tiene sentido ya que es un producto que solo tiene una cantidad finita de factores no iguales a 1.

**Lemma 12.** *Las funciones:*

$$\begin{aligned}\mathbb{N} &\rightarrow \omega^{[\mathbb{N}]} \\ x &\rightarrow ((x)_1, (x)_2, \dots)\end{aligned}$$

$$\begin{aligned}\omega^{[\mathbb{N}]} &\rightarrow \mathbb{N} \\ (s_1, s_2, \dots) &\rightarrow \langle s_1, s_2, \dots \rangle\end{aligned}$$

*son biyecciones una inversa de la otra.*

**Lemma 13.** *Para cada  $x \in \mathbb{N}$ :*

$$1. \quad Lt(x) = 0 \Leftrightarrow x = 1$$

$$2. \quad x = \prod_{i=1}^{Lt(x)} pr(i)^{(x)_i}$$

*Cabe destacar entonces que la función  $\lambda ix[(x)_i]$  tiene dominio igual a  $\mathbb{N}^2$  y la función  $\lambda ix[Lt(x)]$  tiene dominio igual a  $\mathbb{N}$ .*

## 2 Procedimientos efectivos

**Lemma 14.** Sean  $S_1, S_2 \subseteq \omega^n \times \Sigma^*$  conjuntos  $\Sigma$ -efectivamente enumerables, entonces:

a)  $S_1 \cup S_2$  es  $\Sigma$ -efectivamente enumerable.

b)  $S_1 \cap S_2$  es  $\Sigma$ -efectivamente enumerable.

*Proof.* El caso en el que alguno de los conjuntos es vacío es trivial. Supongamos que  $S_1, S_2 \neq \emptyset$  y sean  $\mathbb{P}_1$  y  $\mathbb{P}_2$  procedimientos que enumeran a  $S_1$  y  $S_2$ .

a) El siguiente procedimiento enumera al conjunto  $S_1 \cup S_2$ :

**Si  $x$  es par:** realizar  $\mathbb{P}_1$  partiendo de  $x/2$  y dar el elemento de  $S_1$  obtenido como salida.

**Si  $x$  es impar:** realizar  $\mathbb{P}_2$  partiendo de  $(x-1)/2$  y dar el elemento de  $S_2$  obtenido como salida.

b) Veamos ahora que  $S_1 \cap S_2$  es  $\Sigma$ -efectivamente enumerable:

**Si  $S_1 \cap S_2 = \emptyset$ :** entonces no hay nada que probar.

**Si  $S_1 \cap S_2 \neq \emptyset$ :** sea  $z_0$  un elemento fijo de  $S_1 \cap S_2$ . Sea  $\mathbb{P}$  un procedimiento efectivo el cual enumere a  $\omega \times \omega$ .

El siguiente procedimiento enumera al conjunto  $S_1 \cap S_2$ :

**Etap 1:** Realizar  $\mathbb{P}$  con dato de entrada  $x$ , para obtener un par  $(x_1, x_2) \in \omega \times \omega$ .

**Etap 2:** Realizar  $\mathbb{P}_1$  con dato de entrada  $x_1$  para obtener un elemento  $z_1 \in S_1$ .

**Etap 3:** Realizar  $\mathbb{P}_2$  con dato de entrada  $x_2$  para obtener un elemento  $z_2 \in S_2$ .

**Etap 4:** Si  $z_1 = z_2$ , entonces dar como dato de salida  $z_1$ . En caso contrario dar como dato de salida  $z_0$ .

□

**Lemma 15.** Si  $S \subseteq \omega^n \times \Sigma^*$  es  $\Sigma$ -efectivamente computable entonces  $S$  es  $\Sigma$ -efectivamente enumerable.

*Proof.* El caso en el que  $S$  es vacío es trivial. Supongamos  $S \neq \emptyset$ . Sea  $(\vec{z}, \gamma) \in S$ , fijo. Recordemos que  $\omega^n \times \Sigma^{*m}$  es  $\Sigma$ -efectivamente enumerable. Sean:

- $\mathbb{P}_1$  un procedimiento efectivo que enumere a  $\omega^n \times \Sigma^{*m}$
- $\mathbb{P}_2$  un procedimiento efectivo que compute a  $\chi_S$ .

El siguiente procedimiento enumera a  $S$ :

**Etap 1:** Realizar  $\mathbb{P}_1$  con  $x$  de entrada para obtener  $(\vec{x}, \vec{\alpha}) \in \omega^n \times \Sigma^{*m}$ .

**Etap 2:** Realizar  $\mathbb{P}_2$  con  $(\vec{x}, \vec{\alpha})$  de entrada para obtener el valor *Booleano*  $e$  de salida.

**Etap 3:** Si  $e = 1$ : dar como dato de salida  $(\vec{x}, \vec{\alpha})$ .

Si  $e = 0$ : dar como dato de salida  $(\vec{z}, \gamma)$ .

□

**Theorem 16.** Sea  $S \subseteq \omega^n \times \Sigma^*$ . Son equivalentes:

- a)  $S$  es  $\Sigma$ -efectivamente computable.
- b)  $S$  y  $(\omega^n \times \Sigma^*) - S$  son  $\Sigma$ -efectivamente enumerables.

*Proof.*  $\boxed{(a) \Rightarrow (b)}$  Si  $S$  es  $\Sigma$ -efectivamente computable, por el **Lemma 15** tenemos que  $S$  es  $\Sigma$ -efectivamente enumerable. Notese además que, dado que  $S$  es  $\Sigma$ -efectivamente computable,  $(\omega^n \times \Sigma^{*m}) - S$  también lo es, es decir, que aplicando nuevamente el **Lemma 15** tenemos que  $(\omega^n \times \Sigma^{*m}) - S$  es  $\Sigma$ -efectivamente enumerable.

$\boxed{(b) \Rightarrow (a)}$  Sean:

- $\mathbb{P}_1$  un procedimiento efectivo que enumere a  $S$ .
- $\mathbb{P}_2$  un procedimiento efectivo que enumere a  $(\omega^n \times \Sigma^{*m}) - S$ .

El siguiente procedimiento computa el predicado  $\chi_S$ :

**Etapas:** Darle a la variable  $T$  el valor 0.

**Etapas:** Realizar  $\mathbb{P}_1$  con el valor de  $T$  como entrada para obtener de salida la upla  $(\vec{y}, \vec{\beta})$ .

**Etapas:** Realizar  $\mathbb{P}_2$  con el valor de  $T$  como entrada para obtener de salida la upla  $(\vec{z}, \vec{\gamma})$ .

**Etapas:** Si  $(\vec{y}, \vec{\beta}) = (\vec{x}, \vec{\alpha})$ : entonces detenerse y dar como dato de salida el valor 1.

Si  $(\vec{z}, \vec{\gamma}) = (\vec{x}, \vec{\alpha})$ : entonces detenerse y dar como dato de salida el valor 0.

Si no sucede ninguna de las dos posibilidades: aumentar en 1 el valor de la variable  $T$  y dirigirse a la Etapa 2. □

**Theorem 17.** Dado  $S \subseteq \omega^n \times \Sigma^*$ , son equivalentes:

1.  $S$  es  $\Sigma$ -efectivamente enumerable.
2.  $S = \emptyset$  ó  $S = I_F$ , para alguna  $F : \omega \rightarrow \omega^n \times \Sigma^*$  tal que cada  $F_i$  es  $\Sigma$ -efectivamente computable.
3.  $S = I_F$ , para alguna  $F : D_F \subseteq \omega^k \times \Sigma^{*l} \rightarrow \omega^n \times \Sigma^*$  tal que cada  $F_i$  es  $\Sigma$ -efectivamente computable.
4.  $S = D_f$ , para alguna función  $f$  la cual es  $\Sigma$ -efectivamente computable.

### 3 Funciones $\Sigma$ -recursivas

**Lemma 18.** Si  $f, f_1, \dots, f_{n+m}$  son  $\Sigma$ -efectivamente computables, entonces  $f \circ (f_1, \dots, f_{n+m})$  lo es.

*Proof.* Sean  $\mathbb{P}, \mathbb{P}_1, \dots, \mathbb{P}_{n+m}$  procedimientos efectivos los cuales computen las funciones  $f, f_1, \dots, f_{n+m}$ , respectivamente. Usando estos procedimientos es fácil definir un procedimiento efectivo el cual compute a  $f \circ (f_1, \dots, f_{n+m})$ .  $\square$

**Lemma 19.** Si  $f$  y  $g$  son  $\Sigma$ -efectivamente computables, entonces  $R(f, g)$  lo es.

*Proof.* HACER!!!!!!  $\square$

**Lemma 20.** Si  $f$  y cada  $\mathcal{G}_a$  son  $\Sigma$ -efectivamente computables, entonces  $R(f, \mathcal{G})$  lo es.

**Theorem 21.** Si  $f \in \text{PR}^\Sigma$ , entonces  $f$  es  $\Sigma$ -efectivamente computable.

*Proof.* Dejamos al lector la prueba por inducción en  $k$  de que si  $f \in \text{PR}_k^\Sigma$ , entonces  $f$  es  $\Sigma$ -efectivamente computable, la cual sale en forma directa usando los lemas anteriores que garantizan que los constructores de composición y recursión primitiva preservan la computabilidad efectiva.  $\square$

**Lemma 22.** 1.  $\emptyset \in \text{PR}^\emptyset$ .

2.  $\lambda xy [x + y] \in \text{PR}^\emptyset$ .

3.  $\lambda xy [x.y] \in \text{PR}^\emptyset$ .

4.  $\lambda x [x!] \in \text{PR}^\emptyset$ .

*Proof.* 1. Notese que  $\emptyset = \text{Pred} \circ C_0^{0,0} \in \text{PR}_1^\emptyset$

2. Notar que:

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

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

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  $C_0^{1,0} = R(C_0^{0,0}, p_1^{2,0}) \in \text{PR}_1^\emptyset$ . También note que  $\lambda tx [t.x] = R(C_0^{1,0}, \lambda xy [x + y] \circ (p_1^{3,0}, p_3^{3,0}))$ , lo cual por (1) implica que  $\lambda tx [t.x] \in \text{PR}_3^\emptyset$ .

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

Ya que  $C_1^{0,0} = \text{Suc} \circ C_0^{0,0}$ , tenemos que  $C_1^{0,0} \in \text{PR}_1^\emptyset$ . Por (2), tenemos que  $\lambda xy [x.y] \circ (p_1^{2,0}, \text{Suc} \circ p_2^{2,0}) \in \text{PR}_4^\emptyset$ , obteniendo que  $\lambda x [x!] \in \text{PR}_5^\emptyset$ .  $\square$

**Lemma 23.** Supongamos  $\Sigma \neq \emptyset$ .

a)  $\lambda\alpha\beta[\alpha\beta] \in \text{PR}^\Sigma$ .

b)  $\lambda\alpha[|\alpha|] \in \text{PR}^\Sigma$ .

*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  $\lambda\alpha\beta[\alpha\beta] = R(p_1^{0,1}, \mathcal{G})$ , 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  $\lambda\alpha[|\alpha|] = R(C_0^{0,0}, \mathcal{G})$ , donde  $\mathcal{G}_a = \text{Suc} \circ p_1^{1,1}$ , para cada  $a \in \Sigma$ . □

**Lemma 24.** a)  $C_k^{n,m}, C_\alpha^{n,m} \in \text{PR}^\Sigma$ , para  $n, m, k \geq 0, \alpha \in \Sigma^*$ .

b)  $C_k^{n,0} \in \text{PR}^\emptyset$ , para  $n, k \geq 0$ .

**Lemma 25.** 1.  $\lambda xy[x^y] \in \text{PR}^\emptyset$ .

2.  $\lambda t\alpha[\alpha^t] \in \text{PR}^\Sigma$ .

*Proof.* a) Note que:  $\lambda tx[x^t] = R(C_1^{1,0}, \lambda xy[x.y] \circ (p_1^{3,0}, p_3^{3,0})) \in \text{PR}^\emptyset$ .

Osea que  $\lambda xy[x^y] = \lambda tx[x^t] \circ (p_2^{2,0}, p_1^{2,0}) \in \text{PR}^\emptyset$ .

b) Note que:  $\lambda t\alpha[\alpha^t] = R(C_\varepsilon^{0,1}, \lambda\alpha\beta[\alpha\beta] \circ (p_3^{1,2}, p_2^{1,2})) \in \text{PR}^\Sigma$ . □

**Lemma 26.** Si  $<$  es un orden total estricto sobre un alfabeto no vacío  $\Sigma$ , entonces  $s^<, \#^<$  y  $*^<$  pertenecen a  $\text{PR}^\Sigma$ .

*Proof.* Supongamos  $\Sigma = \{a_1, \dots, a_k\}$  y  $<$  dado por  $a_1 < \dots < a_k$ . Ya que:

$$\begin{aligned}s^<(\varepsilon) &= a_1 \\ s^<(\alpha a_i) &= \alpha a_{i+1}, \text{ para } i < k \\ s^<(\alpha a_k) &= s^<(\alpha) a_1\end{aligned}$$

tenemos que  $s^< = R(C_{a_1}^{0,0}, \mathcal{G})$ , donde  $\mathcal{G}_{a_i} = d_{a_{i+1}} \circ p_1^{0,2}$ , para  $i = 1, \dots, k-1$  y  $\mathcal{G}_{a_k} = d_{a_1} \circ p_2^{0,2}$ . Osea que  $s^< \in \text{PR}^\Sigma$ . Ya que:

$$\begin{aligned}*^<(0) &= \varepsilon \\ *^<(t+1) &= s^<(*^<(t))\end{aligned}$$

podemos ver que  $*^< \in \text{PR}^\Sigma$ . Ya que:

$$\begin{aligned}\#^<(\varepsilon) &= 0 \\ \#^<(\alpha a_i) &= \#^<(\alpha).k + i, \text{ para } i = 1, \dots, k\end{aligned}$$

tenemos que  $\#^< = R(C_0^{0,0}, \mathcal{G})$ , donde  $\mathcal{G}_{a_i} = \lambda xy [x + y] \circ (\lambda xy [x.y] \circ (p_1^{1,1}, C_k^{1,1}), C_i^{1,1})$ , para  $i = 1, \dots, k$ .

Osea que  $\#^< \in \text{PR}^\Sigma$ . □

**Lemma 27.** a)  $\lambda xy [x \dot{-} y] \in \text{PR}^\emptyset$ .

b)  $\lambda xy [\max(x, y)] \in \text{PR}^\emptyset$ .

c)  $\lambda xy [x = y] \in \text{PR}^\emptyset$ .

d)  $\lambda xy [x \leq y] \in \text{PR}^\emptyset$ .

e) Si  $\Sigma \neq \emptyset \Rightarrow \lambda \alpha \beta [\alpha = \beta] \in \text{PR}^\Sigma$ .

*Proof.* a) Primero notar que  $\lambda x [x \dot{-} 1] = R(C_0^{0,0}, p_2^{2,0}) \in \text{PR}^\emptyset$ . También note que  $\lambda tx [x \dot{-} t] = R(p_1^{1,0}, \lambda x [x \dot{-} 1] \circ p_1^{3,0}) \in \text{PR}^\emptyset$ .

Osea que  $\lambda xy [x \dot{-} y] = \lambda tx [x \dot{-} t] \circ (p_2^{2,0}, p_1^{2,0}) \in \text{PR}^\emptyset$ .

b) Note que  $\lambda xy [\max(x, y)] = \lambda xy [(x + (y \dot{-} x))]$ .

c) Note que  $\lambda xy [x = y] = \lambda xy [1 \dot{-} ((x \dot{-} y) + (y \dot{-} x))]$ .

d) Note que  $\lambda xy [x \leq y] = \lambda xy [1 \dot{-} (x \dot{-} y)]$ .

e) Sea  $<$  un orden total estricto sobre  $\Sigma$ . Ya que  $\alpha = \beta$  sii  $\#^<(\alpha) = \#^<(\beta)$ , tenemos que  $\lambda \alpha \beta [\alpha = \beta] = \lambda xy [x = y] \circ (\#^< \circ p_1^{0,2}, \#^< \circ p_2^{0,2})$ .

Osea que podemos aplicar (c) y **Lema 28** implica que  $\chi_S$  es  $\Sigma$ -PR. □

**Lemma 28.** Si  $P : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \omega$  y  $Q : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \omega$  son predicados  $\Sigma$ -PR, entonces  $(P \vee Q)$ ,  $(P \wedge Q)$  y  $\text{not}P$  lo son también.

*Proof.* Note que:

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

□

**Lemma 29.** Si  $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$  son  $\Sigma$ -PR, entonces  $S_1 \cup S_2$ ,  $S_1 \cap S_2$  y  $S_1 - S_2$  lo son.

*Proof.* Note que:

$$\begin{aligned}\chi_{S_1 \cup S_2} &= (\chi_{S_1} \vee \chi_{S_2}) \\ \chi_{S_1 \cap S_2} &= (\chi_{S_1} \wedge \chi_{S_2}) \\ \chi_{S_1 - S_2} &= \lambda [x \dot{-} y] \circ (\chi_{S_1}, \chi_{S_2})\end{aligned}$$

□



**Corollary 30.** Si  $S \subseteq \omega^n \times \Sigma^{*m}$  es finito, entonces  $S$  es  $\Sigma$ -PR.

*Proof.* Por el lema anterior, podemos suponer que:

$$S = \{(z_1, \dots, z_n, \gamma_1, \dots, \gamma_m)\}$$

Noque  $\chi_S$  es el siguiente predicado:

$$(\chi_{z_1} \circ p_1^{n,m} \wedge \dots \wedge \chi_{z_n} \circ p_n^{n,m} \wedge \chi_{\gamma_1} \circ p_{n+1}^{n,m} \wedge \dots \wedge \chi_{\gamma_m} \circ p_{n+m}^{n,m})$$

Ya que los predicados:

$$\chi_{z_i} = \lambda xy [x = y] \circ (p_1^{1,0}, C_{z_i}^{1,0}) \quad \chi_{\gamma_i} = \lambda \alpha \beta [\alpha = \beta] \circ (p_1^{0,1}, C_{\gamma_i}^{0,1})$$

son  $\Sigma$ -PR, el **Lema 28** implica que  $\chi_S$  es  $\Sigma$ -PR.  $\square$

**Lemma 31.** Supongamos  $S_1, \dots, S_n \subseteq \omega, L_1, \dots, L_m \subseteq \Sigma^*$  son conjuntos no vacíos, entonces  $S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$  es  $\Sigma$ -PR sii  $S_1, \dots, S_n, L_1, \dots, L_m$  son  $\Sigma$ -PR.

*Proof.*  $\Rightarrow$  Veremos por ejemplo que  $L_1$  es  $\Sigma$ -PR. Sea  $(z_1, \dots, z_n, z_1, \dots, z_m)$  un elemento fijo de  $S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$ . Note que:  $\alpha \in L_1$  sii  $(z_1, \dots, z_n, \alpha, z_2, \dots, z_m) \in S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$ , lo cual implica que  $\chi_{L_1} = \chi_{S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m} \circ (C_{z_1}^{0,1}, \dots, C_{z_n}^{0,1}, p_1^{0,1}, C_{z_2}^{0,1}, \dots, C_{z_m}^{0,1})$ .

$\Leftarrow$  Note que  $\chi_{S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m}$  es el predicado  $(\chi_{S_1} \circ p_1^{n,m} \wedge \dots \wedge \chi_{S_n} \circ p_n^{n,m} \wedge \chi_{L_1} \circ p_{n+1}^{n,m} \wedge \dots \wedge \chi_{L_m} \circ p_{n+m}^{n,m})$ .  $\square$

**Lemma 32.** Supongamos  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -PR, donde  $O \in \{\omega, \Sigma^*\}$ . Si  $S \subseteq D_f$  es  $\Sigma$ -PR, entonces  $f|_S$  es  $\Sigma$ -PR.

*Proof.* Supongamos  $O = \Sigma^*$ , entonces  $f|_S = \lambda x \alpha [\alpha^x] \circ (Suc \circ Pred \circ \chi_S, f)$  es  $\Sigma$ -PR. El caso  $O = \omega$  es similar usando  $\lambda xy [x^y]$  en lugar de  $\lambda x \alpha [\alpha^x]$ .  $\square$

**Lemma 33.** Si  $f : D_f \subseteq \omega^n \times \Sigma^* \rightarrow O$  es  $\Sigma$ -PR, entonces existe una función  $\Sigma$ PR.  $\bar{f} : \omega^n \times \Sigma^{*m} \rightarrow O$ , tal que  $f = \bar{f}|_{D_f}$ .

*Proof.* Es fácil ver por inducción en  $k$  que el enunciado se cumple para cada  $f \in \text{PR}_k^\Sigma$ .  $\square$

**Proposition 34.** Un conjunto  $S$  es  $\Sigma$ -PR sii  $S$  es el dominio de una función  $\Sigma$ -PR.

*Proof.*  $\Rightarrow$  Note que  $S = D_{Pred \circ \chi_S}$ .

$\Leftarrow$  Probaremos por inducción en  $k$  que  $D_F$  es  $\Sigma$ -PR, para cada  $F \in \text{PR}_k^\Sigma$ . El caso  $k = 0$  es fácil. Supongamos el resultado vale para un  $k$  fijo y supongamos  $F \in \text{PR}_{k+1}^\Sigma$ . Veremos entonces que  $D_F$  es  $\Sigma$ -PR. Hay varios casos.

$$1. \quad F = R(f, g)$$

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

con  $S_1, \dots, S_n \subseteq \omega$  y  $L_1, \dots, L_m \subseteq \Sigma^*$  conjuntos no vacíos y  $f, g \in \text{PR}_k^\Sigma$ . Notese que por definición de  $R(f, g)$ , tenemos que  $D_F = \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$ .

Por HI tenemos que  $D_f = S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$  es  $\Sigma$ -PR, lo cual por el **Lema 31** nos dice que los conjuntos  $S_1, \dots, S_n, L_1, \dots, L_m$  son  $\Sigma$ -PR. Ya que  $\omega$  es  $\Sigma$ -PR, el **Lema 31** nos dice que  $D_F$  es  $\Sigma$ -PR.

Los otros casos de recursión primitiva son dejados al lector.

2.  $\boxed{F = g \circ (g_1, \dots, g_{n+m})}$

$$\begin{aligned} g & : D_g \subseteq \omega^n \times \Sigma^{*m} \rightarrow O \\ g_i & : D_{g_i} \subseteq \omega^k \times \Sigma^{*l} \rightarrow \omega, i = 1, \dots, n \\ g_i & : D_{g_i} \subseteq \omega^k \times \Sigma^{*l} \rightarrow \Sigma^*, i = n+1, \dots, n+m \end{aligned}$$

están en  $\text{PR}_k^\Sigma$ . Por **Lema 33**, hay funciones  $\Sigma$ -PR  $\bar{g}_1, \dots, \bar{g}_{n+m}$  las cuales son  $\Sigma$ -totales y cumplen  $g_i = \bar{g}_i \upharpoonright_{D_{g_i}}$ , para  $i = 1, \dots, n+m$ .

Por HI los conjuntos  $D_g, D_{g_i}, i = 1, \dots, n+m$ , son  $\Sigma$ -PR y por lo tanto  $S = \bigcap_{i=1}^{n+m} D_{g_i}$  lo es. Notese que  $\chi_{D_F} = (\chi_{D_g} \circ (\bar{g}_1, \dots, \bar{g}_{n+m}) \wedge \chi_S)$  lo cual nos dice que  $D_F$  es  $\Sigma$ -PR.

□

**Lemma 35.** Supongamos  $f_i : D_{f_i} \subseteq \omega^n \times \Sigma^{*m} \rightarrow O, i = 1, \dots, k$ , son funciones  $\Sigma$ -PR tales que  $D_{f_i} \cap D_{f_j} = \emptyset$  para  $i \neq j$ , entonces  $f_1 \cup \dots \cup f_k$  es  $\Sigma$ -PR.

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

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

funciones  $\Sigma$ -PR tales que  $\bar{f}_i|_{D_{f_i}} = f_i, i = 1, 2$  (por **Lemma 33**).

Por **Lemma 34** los conjuntos  $D_{f_1}$  y  $D_{f_2}$  son  $\Sigma$ -PR y por lo tanto lo es  $D_{f_1} \cup D_{f_2}$ . Ya que:

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

tenemos que  $f_1 \cup f_2$  es  $\Sigma$ -PR.

El caso  $k > 2$  puede probarse por inducción ya que:

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

□

**Corollary 36.** Supongamos  $f$  es una función  $\Sigma$ -mixta cuyo dominio es finito, entonces  $f$  es  $\Sigma$ -PR.

**Lemma 37.**  $\lambda i \alpha [[\alpha]_i]$  es  $\Sigma$ -PR.

**Lemma 38.** Sean  $n, m \geq 0$ .

- a) Si  $f : \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m \rightarrow \omega$  es  $\Sigma$ -PR, con  $S_1, \dots, S_n \subseteq \omega$  y  $L_1, \dots, L_m \subseteq \Sigma^*$  no vacíos, entonces lo son las funciones  $\lambda xy \vec{x} \vec{\alpha} \left[ \sum_{t=x}^{t=y} f(t, \vec{x}, \vec{\alpha}) \right]$  y  $\lambda xy \vec{x} \vec{\alpha} \left[ \prod_{t=x}^{t=y} f(t, \vec{x}, \vec{\alpha}) \right]$ .
- b) Si  $f : \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m \rightarrow \Sigma^*$  es  $\Sigma$ -PR, con  $S_1, \dots, S_n \subseteq \omega$  y  $L_1, \dots, L_m \subseteq \Sigma^*$  no vacíos, entonces lo es la función  $\lambda xy \vec{x} \vec{\alpha} \left[ \subset_{t=x}^{t=y} f(t, \vec{x}, \vec{\alpha}) \right]$ .

*Proof.* (a): Sea  $G = \lambda t x \vec{x} \vec{\alpha} \left[ \sum_{i=x}^{i=t} f(i, \vec{x}, \vec{\alpha}) \right]$ . Ya que:

$$\lambda xy \vec{x} \vec{\alpha} \left[ \sum_{i=x}^{i=y} f(i, \vec{x}, \vec{\alpha}) \right] = G \circ \left( p_2^{n+2,m}, p_1^{n+2,m}, p_3^{n+2,m}, \dots, p_{n+m+2}^{n+2,m} \right)$$

solo tenemos que probar que  $G$  es  $\Sigma$ -PR. Primero note que:

$$\begin{aligned} G(0, x, \vec{x}, \vec{\alpha}) &= \begin{cases} 0 & \text{si } x > 0 \\ f(0, \vec{x}, \vec{\alpha}) & \text{si } x = 0 \end{cases} \\ G(t+1, x, \vec{x}, \vec{\alpha}) &= \begin{cases} 0 & \text{si } x > t+1 \\ G(t, x, \vec{x}, \vec{\alpha}) + f(t+1, \vec{x}, \vec{\alpha}) & \text{si } x \leq t+1 \end{cases} \end{aligned}$$

Sean:

$$\begin{aligned} D_1 &= \{(x, \vec{x}, \vec{\alpha}) \in \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m : x > 0\} \\ D_2 &= \{(x, \vec{x}, \vec{\alpha}) \in \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m : x = 0\} \\ H_1 &= \{(z, t, x, \vec{x}, \vec{\alpha}) \in \omega^3 \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m : x > t+1\} \\ H_2 &= \{(z, t, x, \vec{x}, \vec{\alpha}) \in \omega^3 \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m : x \leq t+1\} \end{aligned}$$

Es fácil de chequear que estos conjuntos son  $\Sigma$ -PR. Veamos que por ejemplo  $H_1$  lo es. Es decir debemos ver que  $\chi_{H_1}$  es  $\Sigma$ -PR. Ya que  $f$  es  $\Sigma$ -PR tenemos que  $D_f = \omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$  es  $\Sigma$ -PR, lo cual por el **Lemma 31** nos dice que los conjuntos  $S_1, \dots, S_n, L_1, \dots, L_m$  son  $\Sigma$ -PR. Ya que  $\omega$  es  $\Sigma$ -PR, el **Lemma 31** nos dice que  $R = \omega^3 \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$  es  $\Sigma$ -PR. Notese que  $\chi_{H_1} = (\chi_R \wedge \lambda z t x \vec{x} \vec{\alpha} [x > t + 1])$  por cual  $\chi_{H_1}$  es  $\Sigma$ -PR ya que es la conjunción de dos predicados  $\Sigma$ -PR.

Además note que  $G = R(h, g)$ , donde:

$$\begin{aligned} h &= C_0^{n+1, m} \upharpoonright_{D_1} \cup \lambda x \vec{x} \vec{\alpha} [f(0, \vec{x}, \vec{\alpha})] \upharpoonright_{D_2} \\ g &= C_0^{n+3, m} \upharpoonright_{H_1} \cup \lambda z t x \vec{x} \vec{\alpha} [z + f(t + 1, \vec{x}, \vec{\alpha})] \upharpoonright_{H_2} \end{aligned}$$

O sea que el **Lemma 35** y el **Lemma 32** garantizan que  $G$  es  $\Sigma$ -PR.  $\square$

**Lemma 39.** Sean  $n, m \geq 0$ .

- a) Sea  $P : S \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m \rightarrow \omega$  un predicado  $\Sigma$ -PR y supongamos  $\bar{S} \subseteq S$  es  $\Sigma$ -PR, entonces  $\lambda x \vec{x} \vec{\alpha} [(\forall t \in \bar{S})_{t \leq x} P(t, \vec{x}, \vec{\alpha})]$  y  $\lambda x \vec{x} \vec{\alpha} [(\exists t \in \bar{S})_{t \leq x} P(t, \vec{x}, \vec{\alpha})]$  son predicados  $\Sigma$ -PR. (Notar que el dominio de estos predicados es  $\omega \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m$ ).
- b) Sea  $P : S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m \times L \rightarrow \omega$  un predicado  $\Sigma$ -PR y supongamos  $\bar{L} \subseteq L$  es  $\Sigma$ -PR, entonces  $\lambda x \vec{x} \vec{\alpha} [(\forall \alpha \in \bar{L})_{|\alpha| \leq x} P(\vec{x}, \vec{\alpha}, \alpha)]$  y  $\lambda x \vec{x} \vec{\alpha} [(\exists \alpha \in \bar{L})_{|\alpha| \leq x} P(\vec{x}, \vec{\alpha}, \alpha)]$  son predicados  $\Sigma$ -PR.

*Proof.* Se probará solamente el inciso (a). Sea:

$$\bar{P} = P \upharpoonright_{\bar{S} \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m} \cup C_1^{1+n, m} \upharpoonright_{(\omega - \bar{S}) \times S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m}$$

Notese que  $\bar{P}$  es  $\Sigma$ -PR. Ya que:

$$\begin{aligned} \lambda x \vec{x} \vec{\alpha} [(\forall t \in \bar{S})_{t \leq x} P(t, \vec{x}, \vec{\alpha})] &= \lambda x \vec{x} \vec{\alpha} \left[ \prod_{t=0}^{t=x} \bar{P}(t, \vec{x}, \vec{\alpha}) \right] \\ &= \lambda x y \vec{x} \vec{\alpha} \left[ \prod_{t=x}^{t=y} \bar{P}(t, \vec{x}, \vec{\alpha}) \right] \circ (C_0^{1+n, m}, p_1^{1+n, m}, \dots, p_{1+n+m}^{1+n, m}) \end{aligned}$$

el **Lemma 38** implica que  $\lambda x \vec{x} \vec{\alpha} [(\forall t \in \bar{S})_{t \leq x} P(t, \vec{x}, \vec{\alpha})]$  es  $\Sigma$ -PR.

Finalmente note que:

$$\lambda x \vec{x} \vec{\alpha} [(\exists t \in \bar{S})_{t \leq x} P(t, \vec{x}, \vec{\alpha})] = \neg \lambda x \vec{x} \vec{\alpha} [(\forall t \in \bar{S})_{t \leq x} \neg P(t, \vec{x}, \vec{\alpha})]$$

es  $\Sigma$ -PR.  $\square$

**Lemma 40.** a) El predicado  $\lambda x y [x \text{ divide } y]$  es  $\emptyset$ -PR.

b) El predicado  $\lambda x [x \text{ es primo}]$  es  $\emptyset$ -PR.

c) El predicado  $\lambda \alpha \beta [\alpha \text{ inicial } \beta]$  es  $\Sigma$ -PR.

*Proof.* a) Si tomamos  $P = \lambda t x_1 x_2 [x_2 = t.x_1] \in \text{PR}^\emptyset$ , tenemos que:

$$\begin{aligned} \lambda x_1 x_2 [x_1 \text{ divide } x_2] &= \lambda x_1 x_2 [(\exists t \in \omega)_{t \leq x_2} P(t, x_1, x_2)] \\ &= \lambda x x_1 x_2 [(\exists t \in \omega)_{t \leq x} P(t, x_1, x_2)] \circ (p_2^{2,0}, p_1^{2,0}, p_2^{2,0}) \end{aligned}$$

por lo que podemos aplicar el **Lemma 39** anterior.

b) Ya que:

$$x \text{ es primo sii } x > 1 \wedge ((\forall t \in \omega)_{t \leq x} t = 1 \vee t = x \vee \neg(t \text{ divide } x))$$

podemos usar un argumento similar al de la prueba de (a).

c) es dejado al lector. □

**Lemma 41.** Si  $P : D_P \subseteq \omega \times \omega^n \times \Sigma^{*m} \rightarrow \omega$  es un predicado  $\Sigma$ -EC y  $D_P$  es  $\Sigma$ -EC, entonces la función  $M(P)$  es  $\Sigma$ -EC.

*Proof.* Ejercicio. □

**Theorem 42.** Si  $f \in R^\Sigma$ , entonces  $f$  es  $\Sigma$ -EC.

*Proof.* Dejamos al lector la prueba por inducción en  $k$  de que si  $f \in R_k^\Sigma$ , entonces  $f$  es  $\Sigma$ -EC. □

**Lemma 43.** Sean  $n, m \geq 0$ . Sea  $P : D_P \subseteq \omega \times \omega^n \times \Sigma^{*m} \rightarrow \omega$  un predicado  $\Sigma$ -PR, entonces:

a)  $M(P)$  es  $\Sigma$ -R.

b) Si hay una función  $\Sigma$ -PR  $f : \omega^n \times \Sigma^{*m} \rightarrow \omega$  tal que:

$$M(P)(\vec{x}, \vec{\alpha}) = \min_t P(t, \vec{x}, \vec{\alpha}) \leq f(\vec{x}, \vec{\alpha}), \text{ para cada } (\vec{x}, \vec{\alpha}) \in D_{M(P)}$$

entonces  $M(P)$  es  $\Sigma$ -PR.

*Proof.* a) Sea  $\bar{P} = P \upharpoonright_{D_P} \cup C_0^{m+1, m} \upharpoonright_{(\omega^{n+1} \times \Sigma^{*m}) - D_P}$ . Dejamos al lector verificar cuidadosamente que  $M(P) = M(\bar{P})$ . Veremos entonces que  $M(\bar{P})$  es  $\Sigma$ -R. Note que  $\bar{P}$  es  $\Sigma$ -PR. Sea  $k$  tal que  $\bar{P} \in PR_k^\Sigma$ . Ya que  $\bar{P}$  es  $\Sigma$ -total y  $\bar{P} \in PR_k^\Sigma \subseteq R_k^\Sigma$ , tenemos que  $M(\bar{P}) \in R_{k+1}^\Sigma$  y por lo tanto  $M(\bar{P}) \in R^\Sigma$ .

b) Primero veremos que  $D_{M(\bar{P})}$  es un conjunto  $\Sigma$ -PR. Notese que:

$$\chi_{D_{M(\bar{P})}} = \lambda \vec{x} \vec{\alpha} \left[ (\exists t \in \omega)_{t \leq f(\vec{x}, \vec{\alpha})} \bar{P}(t, \vec{x}, \vec{\alpha}) \right]$$

lo cual nos dice que:

$$\chi_{D_{M(\bar{P})}} = \lambda x \vec{x} \vec{\alpha} \left[ (\exists t \in \omega)_{t \leq x} \bar{P}(t, \vec{x}, \vec{\alpha}) \right] \circ (f, p_1^{n, m}, \dots, p_{n+m}^{n, m})$$

Pero el **Lemma 39** nos dice que  $\lambda x \vec{x} \vec{\alpha} \left[ (\exists t \in \omega)_{t \leq x} \bar{P}(t, \vec{x}, \vec{\alpha}) \right]$  es  $\Sigma$ -PR por lo cual tenemos que  $\chi_{D_{M(\bar{P})}}$  lo es.

Sea:

$$P_1 = \lambda t \vec{x} \vec{\alpha} \left[ \bar{P}(t, \vec{x}, \vec{\alpha}) \wedge (\forall j \in \omega)_{j \leq t} j = t \vee \neg \bar{P}(j, \vec{x}, \vec{\alpha}) \right]$$

Note que  $P_1$  es  $\Sigma$ -total. Dejamos al lector usando Lemmas anteriores probar que  $P_1$  es  $\Sigma$ -PR. Además notese que para cada  $(\vec{x}, \vec{\alpha}) \in \omega^n \times \Sigma^{*m}$  tenemos que:

$$P_1(t, \vec{x}, \vec{\alpha}) = 1 \text{ si y solo si } t = M(\bar{P})(\vec{x}, \vec{\alpha})$$

Esto nos dice que:

$$M(\bar{P}) = \left( \lambda \vec{x} \vec{\alpha} \left[ \prod_{t=0}^{f(\vec{x}, \vec{\alpha})} t^{P_1(t, \vec{x}, \vec{\alpha})} \right] \right) \upharpoonright_{D_{M(\bar{P})}}$$

por lo cual para probar que  $M(\bar{P})$  es  $\Sigma$ -PR solo nos resta probar que:

$$F = \lambda \vec{x} \vec{\alpha} \left[ \prod_{t=0}^{f(\vec{x}, \vec{\alpha})} t^{P_1(t, \vec{x}, \vec{\alpha})} \right]$$

lo es. Pero:

$$F = \lambda xy \vec{x} \vec{\alpha} \left[ \prod_{t=x}^y t^{P_1(t, \vec{x}, \vec{\alpha})} \right] \circ (C_0^{n,m}, f, p_1^{n,m}, \dots, p_{n+m}^{n,m})$$

y por lo tanto el **Lemma 38** nos dice que  $F$  es  $\Sigma$ -PR. De esta manera hemos probado que  $M(\bar{P})$  es  $\Sigma$ -PR y por lo tanto  $M(P)$  lo es. □

**Lemma 44.** *Las siguientes funciones son  $\emptyset$ -PR:*

- a)  $Q : \omega \times \mathbf{N} \rightarrow \omega$   
 $(x, y) \rightarrow$  cociente de la division de  $x$  por  $y$
- b)  $R : \omega \times \mathbf{N} \rightarrow \omega$   
 $(x, y) \rightarrow$  resto de la division de  $x$  por  $y$
- c)  $pr : \mathbf{N} \rightarrow \omega$   
 $n \rightarrow$   $n$ -esimo numero primo

*Proof.* a) Veamos primero veamos que  $Q = M(P)$ , donde  $P = \lambda txy [(t+1).y > x]$ . Notar que:

$$\begin{aligned} D_{M(P)} &= \{(x, y) : (\exists t \in \omega) P(t, x, y) = 1\} \\ &= \{(x, y) : (\exists t \in \omega) (t+1).y > x\} \\ &= \omega \times \mathbf{N} \\ &= D_Q \end{aligned}$$

Dejamos al lector la fácil verificación de que para cada  $(x, y) \in \omega \times \mathbf{N}$ , se tiene que:

$$Q(x, y) = M(P)(x, y) = \min_t (t+1).y > x$$

Esto prueba que  $Q = M(P)$ . Ya que  $P$  es  $\emptyset$ -PR y además:

$$Q(x, y) \leq p_1^{2,0}(x, y), \text{ para cada } (x, y) \in \omega \times \mathbf{N}$$

el inciso (b) del **Lemma 43** implica que  $Q \in \text{PR}^\emptyset$ .

b) Notese que:

$$R = \lambda xy [x \dot{-} Q(x, y).y]$$

y por lo tanto  $R \in \text{PR}^\emptyset$ .

c) Para ver que  $pr$  es  $\emptyset$ -PR, veremos que la extensión  $h : \omega \rightarrow \omega$ , dada por  $h(0) = 0$  y  $h(n) = pr(n)$ ,  $n \geq 1$ , es  $\emptyset$ -PR. Primero note que:

$$\begin{aligned} h(0) &= 0 \\ h(x+1) &= \min_t (t \text{ es primo} \wedge t > h(x)) \end{aligned}$$

Osea que  $h = R(C_0^{0,0}, M(P))$ , donde:

$$P = \lambda t z x [t \text{ es primo} \wedge t > z]$$

Es decir que solo nos resta ver que  $M(P)$  es  $\emptyset$ -PR.

Claramente  $P$  es  $\emptyset$ -PR. Veamos que para cada  $(z, x) \in \omega^2$ , tenemos que:

$$M(P)(z, x) = \min_t (t \text{ es primo} \wedge t > z) \leq z! + 1$$

Sea  $p$  primo tal que  $p$  divide a  $z! + 1$ . Es fácil ver que entonces  $p > z$ . Pero esto claramente nos dice que:

$$\min_t (t \text{ es primo} \wedge t > z) \leq p \leq z! + 1$$

Osea que el inciso (b) del **Lemma 43** implica que  $M(P)$  es  $\emptyset$ -PR ya que podemos tomar  $f = \lambda z x [z! + 1]$ . □

**Lemma 45.** Las funciones  $\lambda xi [(x)_i]$  y  $\lambda x [Lt(x)]$  son  $\emptyset$ -PR.

**Lemma 46.** Este lemma no se evalua.

**Lemma 47.** Supongamos que  $\Sigma \neq \emptyset$ . Sea  $<$  un orden total estricto sobre  $\Sigma$ , sean  $n, m \geq 0$  y sea  $P : D_P \subseteq \omega^n \times \Sigma^{*m} \times \Sigma^* \rightarrow \omega$  un predicado  $\Sigma$ -PR, entonces:

a)  $M^<(P)$  es  $\Sigma$ -R.

b) Si existe una función  $\Sigma$ -PR  $f : \omega^n \times \Sigma^{*m} \rightarrow \omega$  tal que:

$$|M^<(P)(\vec{x}, \vec{\alpha})| = |\min_\alpha^< P(\vec{x}, \vec{\alpha}, \alpha)| \leq f(\vec{x}, \vec{\alpha}), \text{ para cada } (\vec{x}, \vec{\alpha}) \in D_{M^<(P)}$$

entonces  $M^<(P)$  es  $\Sigma$ -PR.

**Lemma 48.** Este lemma no se evalua.

**Lemma 49.** Este lemma no se evalua.

**Lemma 50.** Este lemma no se evalua.

**Theorem 51.** Sean  $\Sigma$  y  $\Gamma$  alfabetos cualesquiera.

a) Supongamos una función  $f$  es  $\Sigma$ -mixta y  $\Gamma$ -mixta, entonces  $f$  es  $\Sigma$ -R (respectivamente  $\Sigma$ -PR) sii  $f$  es  $\Gamma$ -R (respectivamente  $\Gamma$ -PR).

b) Supongamos un conjunto  $S$  es  $\Sigma$ -mixto y  $\Gamma$ -mixto, entonces  $S$  es  $\Sigma$ -PR sii  $S$  es  $\Gamma$ -PR.

## 4 El lenguaje $S^\Sigma$

**Lemma 52:** Se tiene que: (a) Si  $I_1 \dots I_n = J_1 \dots J_m$ , con  $I_1, \dots, I_n, J_1, \dots, J_m \in \text{Ins}^\Sigma$ , entonces  $n = m$  y  $I_j = J_j$  para cada  $j \geq 1$ . (b) Si  $\mathcal{P} \in \text{Pro}^\Sigma$ , entonces existe una unica sucesion de instrucciones  $I_1, \dots, I_n$  tal que  $\mathcal{P} = I_1 \dots I_n$

**Proof:**(a) Supongamos  $I_n$  es un tramo final propio de  $J_m$ . Notar que entonces  $n > 1$ . Es facil ver que entonces ya sea  $J_m = L\bar{u}I_n$  para algun  $u \in \mathbf{N}$ , o  $I_n$  es de la forma GOTO  $L\bar{n}$  y  $J_m$  es de la forma  $w\text{IF } P\bar{k} \text{ BEGINS } a \text{ GOTO } L\bar{n}$  donde  $w \in \{L\bar{n} : n \in \mathbf{N}\} \cup \{\varepsilon\}$ . El segundo caso no puede darse porque entonces el anteultimo simbolo de  $I_{n-1}$  deberia ser S lo cual no sucede para ninguna instruccion. O sea que

$$I_1 \dots I_n = J_1 \dots J_{m-1} L\bar{u} I_n$$

lo cual dice que (\*)  $I_1 \dots I_{n-1} = J_1 \dots J_{m-1} L\bar{u}$ . Es decir que  $L\bar{u}$  es tramo final de  $I_{n-1}$  y por lo tanto GOTO  $L\bar{u}$  es tramo final de  $I_{n-1}$ . Por (\*), GOTO es tramo final de  $J_1 \dots J_{m-1}$ , lo cual es imposible. Hemos llegado a una contradiccion lo cual nos dice que  $I_n$  no es un tramo final propio de  $J_m$ . Por simetria tenemos que  $I_n = J_m$ , lo cual usando un razonamiento inductivo nos dice que  $n = m$  y  $I_j = J_j$  para cada  $j \geq 1$ .

(b) Es consecuencia directa de (a).  $\square$

**Theorem 53:** Si  $f$  es  $\Sigma$ -computable, entonces  $f$  es  $\Sigma$ -efectivamente computable.

**Proof:**Supongamos por ejemplo que  $f : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \omega$  es computada por  $\mathcal{P} \in \text{Pro}^\Sigma$ . Es claro que el procedimiento que consiste en realizar las sucesivas instrucciones de  $\mathcal{P}$  (partiendo de  $((x_1, \dots, x_n, 0, 0, \dots), (\alpha_1, \dots, \alpha_m, \varepsilon, \varepsilon, \dots))$ ) y eventualmente terminar en caso de que nos toque realizar la instruccion  $n(\mathcal{P}) + 1$ , y dar como salida el contenido de la variable N1, es un procedimiento efectivo que computa a  $f$ .  $\square$

**Proposition 54:**

(a) Sea  $f : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \omega$  una funcion  $\Sigma$ -computable. Entonces hay un macro  $[V\bar{n} + 1 \leftarrow f(V1, \dots, V\bar{n}, W1, \dots, W\bar{m})]$  (b) Sea  $f : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \Sigma^*$  una funcion  $\Sigma$ -computable. Entonces hay un macro  $[W\bar{m} + 1 \leftarrow f(V1, \dots, V\bar{n}, W1, \dots, W\bar{m})]$

**Proof:**(b) Sea  $\mathcal{P}$  un programa que compute a  $f$ . Tomemos un  $k$  tal que  $k \geq n, m$  y tal que todas las variables y labels de  $\mathcal{P}$  estan en el conjunto

$$\{N1, \dots, N\bar{k}, P1, \dots, P\bar{k}, L1, \dots, L\bar{k}\}.$$

Sea  $\mathcal{P}'$  la palabra que resulta de reemplazar en  $\mathcal{P}$ : - la variable  $N\bar{j}$  por  $V\bar{n} + \bar{j}$ , para cada  $j = 1, \dots, k$  - la variable  $P\bar{j}$  por  $W\bar{m} + \bar{j}$ , para cada  $j = 1, \dots, k$  - el label  $L\bar{j}$  por  $A\bar{j}$ , para cada  $j = 1, \dots, k$  Notese que

$$V\bar{n} + 1 \leftarrow V1$$

$$\vdots$$

$$V\bar{n} + \bar{n} \leftarrow V\bar{n}$$

$$V\bar{n} + \bar{n} + 1 \leftarrow 0$$

$$\vdots$$

$$V\bar{n} + \bar{k} \leftarrow 0$$

$$W\bar{m} + 1 \leftarrow W1$$

$$\vdots$$

$$W\bar{m} + \bar{m} \leftarrow W\bar{m}$$

$$W\bar{m} + \bar{m} + 1 \leftarrow \varepsilon$$

$$\vdots$$

$$W\bar{m} + \bar{k} \leftarrow \varepsilon$$

$$\mathcal{P}'$$

es el macro buscado, el cual tendra sus variables auxiliares y labels en la lista  $V\bar{n} + 1, \dots, V\bar{n} + \bar{k}, W\bar{m} + 1, \dots, W\bar{m} + \bar{k}$ .



□

**Proposition 55:** Sea  $P : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow \omega$  un predicado  $\Sigma$ -computable. Entonces hay un macro [IF  $P(V1, \dots, V\bar{n}, W1, \dots, W\bar{m})$  GOTO A1]

**Theorem 56:** Si  $h$  es  $\Sigma$ -recursiva, entonces  $h$  es  $\Sigma$ -computable.

**Proof:** Probaremos por induccion en  $k$  que

(\*) Si  $h \in R_k^\Sigma$ , entonces  $h$  es  $\Sigma$ -computable. El caso  $k = 0$  es dejado al lector. Supongamos (\*) vale para  $k$ , veremos que vale para  $k + 1$ . Sea  $h \in R_{k+1}^\Sigma - R_k^\Sigma$ . Hay varios casos

Caso 1. Supongamos  $h = M(P)$ , con  $P : \omega \times \omega^n \times \Sigma^{*m} \rightarrow \omega$ , un predicado perteneciente a  $R_k^\Sigma$ . Por hipotesis inductiva,  $P$  es  $\Sigma$ -computable y por lo tanto tenemos un macro

[IF  $P(V1, \dots, V\bar{n} + \bar{1}, W1, \dots, W\bar{m})$  GOTO A1]

L2 IF  $P(\overline{N\bar{n} + \bar{1}}, N1, \dots, N\bar{n}, P1, \dots, P\bar{m})$  GOTO L1

$\overline{N\bar{n} + \bar{1}} \leftarrow \overline{N\bar{n} + \bar{1}} + 1$

GOTO L2

L1  $N1 \leftarrow \overline{N\bar{n} + \bar{1}}$

lo cual nos permite realizar el siguiente programa Es facil chequear que este programa computa  $h$ . Caso 2. Supongamos  $h = R(f, \mathcal{G})$ , con

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

$\mathcal{G}_a : S_1 \times \dots \times S_n \times L_1 \times \dots \times L_m \times \Sigma^* \times \Sigma^* \rightarrow \Sigma^*, a \in \Sigma$

elementos de  $R_k^\Sigma$ . Sea  $\Sigma = \{a_1, \dots, a_r\}$ . Por hipotesis inductiva, las funciones  $f, \mathcal{G}_a, a \in \Sigma$ , son  $\Sigma$ -computables y por lo tanto podemos hacer el siguiente programa via el uso de macros

$[P\bar{m} + \bar{3} \leftarrow f(N1, \dots, N\bar{n}, P1, \dots, P\bar{m})]$

$Lr + \bar{1}$  IF  $P\bar{m} + \bar{1}$  BEGINS  $a_1$  GOTO L1

$\vdots$

IF  $P\bar{m} + \bar{1}$  BEGINS  $a_r$  GOTO  $L\bar{r}$

GOTO  $Lr + \bar{2}$

L1  $P\bar{m} + \bar{1} \leftarrow \frown P\bar{m} + \bar{1}$

$[P\bar{m} + \bar{3} \leftarrow \mathcal{G}_{a_1}(N1, \dots, N\bar{n}, P1, \dots, P\bar{m}, P\bar{m} + \bar{2}, P\bar{m} + \bar{3})]$

$P\bar{m} + \bar{2} \leftarrow P\bar{m} + \bar{2}a_1$

GOTO  $Lr + \bar{1}$

$\vdots$

$L\bar{r}$   $P\bar{m} + \bar{1} \leftarrow \frown P\bar{m} + \bar{1}$

$P\bar{m} + \bar{3} \leftarrow \mathcal{G}_{a_r}(N1, \dots, N\bar{n}, P1, \dots, P\bar{m}, P\bar{m} + \bar{2}, P\bar{m} + \bar{3})$

$P\bar{m} + \bar{2} \leftarrow P\bar{m} + \bar{2}a_r$

GOTO  $Lr + \bar{1}$

$Lr + \bar{2}$   $P1 \leftarrow P\bar{m} + \bar{3}$

Es facil chequear que este programa computa  $h$ . El resto de los casos son dejados al lector.

□

**Lemma 57:** Sea  $\Sigma$  un alfabeto cualquiera. Las funciones  $S$  y  $-$  son  $(\Sigma \cup \Sigma_p)$ -p.r..

**Proof:** Use el Teorema 51. □

**Lemma 58:** Para cada  $n, x \in \omega$ , tenemos que  $|\bar{n}| \leq x$  si y solo si  $n \leq 10^x - 1$

**Lemma 59:**  $\text{Ins}^\Sigma$  es un conjunto  $(\Sigma \cup \Sigma_p)$ -p.r..

**Proof:** Para simplificar la Proof asumiremos que  $\Sigma = \{ @, \& \}$ . Ya que  $\text{Ins}^\Sigma$  es union de los siguientes conjuntos

$$\begin{aligned}
L_1 &= \{N\bar{k} \leftarrow N\bar{k} + 1 : k \in \mathbf{N}\} \\
L_2 &= \{N\bar{k} \leftarrow N\bar{k} - 1 : k \in \mathbf{N}\} \\
L_3 &= \{N\bar{k} \leftarrow N\bar{n} : k, n \in \mathbf{N}\} \\
L_4 &= \{N\bar{k} \leftarrow 0 : k \in \mathbf{N}\} \\
L_5 &= \{\text{IF } N\bar{k} \neq 0 \text{ GOTO } L\bar{m} : k, m \in \mathbf{N}\} \\
L_6 &= \{P\bar{k} \leftarrow P\bar{k}.@ : k \in \mathbf{N}\} \\
L_7 &= \{P\bar{k} \leftarrow P\bar{k}.\& : k \in \mathbf{N}\} \\
L_8 &= \{P\bar{k} \leftarrow \neg P\bar{k} : k \in \mathbf{N}\} \\
L_9 &= \{P\bar{k} \leftarrow P\bar{n} : k, n \in \mathbf{N}\} \\
L_{10} &= \{P\bar{k} \leftarrow \varepsilon : k \in \mathbf{N}\} \\
L_{10} &= \{\text{IF } P\bar{k} \text{ BEGINS } @ \text{ GOTO } L\bar{m} : k, m \in \mathbf{N}\} \\
L_{11} &= \{\text{IF } P\bar{k} \text{ BEGINS } \& \text{ GOTO } L\bar{m} : k, m \in \mathbf{N}\} \\
L_{12} &= \{\text{GOTO } L\bar{m} : m \in \mathbf{N}\} \\
L_{13} &= \{\text{SKIP}\} \\
L_{14} &= \{L\bar{k}\alpha : k \in \mathbf{N} \text{ y } \alpha \in L_1 \cup \dots \cup L_{13}\}
\end{aligned}$$

solo debemos probar que  $L_1, \dots, L_{14}$  son  $(\Sigma \cup \Sigma_p)$ -p.r.. Veremos primero por ejemplo que  $L_{10} = \{\text{IFP}\bar{k}\text{BEGINS@GOTOL}\bar{m} : k, m \in \mathbf{N}\}$

es  $(\Sigma \cup \Sigma_p)$ -p.r.. Primero notese que  $\alpha \in L_{10}$  si y solo si existen  $k, m \in \mathbf{N}$  tales que  $\alpha = \text{IFP}\bar{k}\text{BEGINS@GOTOL}\bar{m}$

Mas formalmente tenemos que  $\alpha \in L_{10}$  si y solo si  $(\exists k \in \mathbf{N})(\exists m \in \mathbf{N}) \alpha = \text{IFP}\bar{k}\text{BEGINS@GOTOL}\bar{m}$

Ya que cuando existen tales  $k, m$  tenemos que  $\bar{k}$  y  $\bar{m}$  son subpalabras de  $\alpha$ , el lema anterior nos dice que  $\alpha \in L_{10}$  si y solo si  $(\exists k \in \mathbf{N})_{k \leq 10|\alpha|} (\exists m \in \mathbf{N})_{m \leq 10|\alpha|} \alpha = \text{IFP}\bar{k}\text{BEGINS@GOTOL}\bar{m}$

Sea  $P = \lambda m k \alpha [\alpha = \text{IFP}\bar{k}\text{BEGINS@GOTOL}\bar{m}]$

Ya que  $D_{\lambda k [\bar{k}]} = \omega$ , tenemos que  $D_P = \omega \times (\Sigma \cup \Sigma_p)^* \times (\Sigma \cup \Sigma_p)^*$ . Notese que  $P = \lambda \alpha \beta [\alpha = \beta] \circ (p_3^{2,1}, f)$

donde  $f = \lambda \alpha_1 \alpha_2 \alpha_3 \alpha_4 [\alpha_1 \alpha_2 \alpha_3 \alpha_4] \circ (C_{\text{IFP}}^{2,1}, \lambda k [\bar{k}] \circ p_2^{2,1}, C_{\text{BEGINS@GOTOL}}^{2,1}, \lambda k [\bar{k}] \circ p_1^{2,1})$

lo cual nos dice que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Notese que

$$\chi_{L_{10}} = \lambda \alpha [(\exists k \in \mathbf{N})_{k \leq 10|\alpha|} (\exists m \in \mathbf{N})_{m \leq 10|\alpha|} P(m, k, \alpha)]$$

Esto nos dice que podemos usar dos veces el Lema 39 para ver que  $\chi_{L_{10}}$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Veamos como. Sea  $Q = \lambda k \alpha [(\exists m \in \mathbf{N})_{m \leq 10|\alpha|} P(m, k, \alpha)]$

Por el Lema 39 tenemos que  $\lambda x k \alpha [(\exists m \in \mathbf{N})_{m \leq x} P(m, k, \alpha)]$

es  $(\Sigma \cup \Sigma_p)$ -p.r. lo cual nos dice que  $Q = \lambda x k \alpha [(\exists m \in \mathbf{N})_{m \leq x} P(m, k, \alpha)] \circ (\lambda \alpha [10|\alpha|] \circ p_2^{1,1}, p_1^{1,1}, p_2^{1,1})$

lo es. Ya que  $\chi_{L_{10}} = \lambda \alpha [(\exists k \in \mathbf{N})_{k \leq 10|\alpha|} Q(k, \alpha)]$

podemos en forma similar aplicar el Lema 39 y obtener finalmente que  $\chi_{L_{10}}$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. En forma similar podemos probar que  $L_1, \dots, L_{13}$  son  $(\Sigma \cup \Sigma_p)$ -p.r.. Esto nos dice que  $L_1 \cup \dots \cup L_{13}$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Notese que  $L_1 \cup \dots \cup L_{13}$  es el conjunto de las instrucciones basicas de  $\mathcal{S}^\Sigma$ . Llamemos  $\text{InsBas}^\Sigma$  a dicho conjunto. Para ver que  $L_{14}$  es  $(\Sigma \cup \Sigma_p)$ -p.r. notemos que

$$\chi_{L_{14}} = \lambda \alpha [(\exists k \in \mathbf{N})_{k \leq 10|\alpha|} (\exists \beta \in \text{InsBas}^\Sigma)_{|\beta| \leq |\alpha|} \alpha = L\bar{k}\beta]$$

lo cual nos dice que aplicando dos veces el Lema 39 obtenemos que  $\chi_{L_{14}}$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Ya que  $\text{Ins}^\Sigma = \text{InsBas}^\Sigma \cup L_{14}$  tenemos que  $\text{Ins}^\Sigma$  es  $(\Sigma \cup \Sigma_p)$ -p.r..  $\square$

**Lemma 60:** *Bas* y *Lab* son funciones  $(\Sigma \cup \Sigma_p)$ -p.r.

**Proof:** Sea  $<$  un orden total estricto sobre  $\Sigma \cup \Sigma_p$ . Sea  $L = \{L\bar{k} : k \in \mathbf{N}\} \cup \{\varepsilon\}$ . Dejamos al lector probar que  $L$  es un conjunto  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea

$$P = \lambda I \alpha [\alpha \in \text{Ins}^\Sigma \wedge I \in \text{Ins}^\Sigma \wedge [\alpha]_1 \neq L \wedge (\exists \beta \in L) I = \beta \alpha]$$

Note que  $D_P = (\Sigma \cup \Sigma_p)^{*2}$ . Dejamos al lector probar que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Notese ademas que cuando  $I \in \text{Ins}^\Sigma$  tenemos que  $P(I, \alpha) = 1$  sii  $\alpha = \text{Bas}(I)$ . Dejamos al lector probar que  $\text{Bas} = M^<(P)$  por lo que para ver que  $\text{Bas}$  es  $(\Sigma \cup \Sigma_p)$ -p.r., solo nos falta ver que la funcion  $\text{Bas}$  es acotada por alguna funcion  $(\Sigma \cup \Sigma_p)$ -p.r. y  $(\Sigma \cup \Sigma_p)$ -total. Pero esto es trivial ya que  $|\text{Bas}(I)| \leq |I| = p_1^{0,1}(I)$  para cada  $I \in \text{Ins}^\Sigma$ . Finalmente note que

$$\text{Lab} = M^<(\lambda I \alpha [\alpha \text{Bas}(I) = I])$$

lo cual nos dice que  $\text{Lab}$  es  $(\Sigma \cup \Sigma_p)$ -p.r..  $\square$  Recordemos que dado un programa  $\mathcal{P}$  habiamos definido  $I_i^{\mathcal{P}} = \varepsilon$ , para  $i = 0$  o  $i > n(\mathcal{P})$ . O sea que la funcion  $(\Sigma \cup \Sigma_p)$ -mixta  $\lambda i \mathcal{P} [I_i^{\mathcal{P}}]$  tiene dominio igual a  $\omega \times \text{Pro}^\Sigma$ .

**Lemma 61:** (a)  $\text{Pro}^\Sigma$  es un conjunto  $(\Sigma \cup \Sigma_p)$ -p.r. (b)  $\lambda \mathcal{P} [n(\mathcal{P})]$  y  $\lambda i \mathcal{P} [I_i^{\mathcal{P}}]$  son funciones  $(\Sigma \cup \Sigma_p)$ -p.r..

**Proof:** Ya que  $\text{Pro}^\Sigma = D_{\lambda \mathcal{P} [n(\mathcal{P})]}$  tenemos que (b) implica (a). Para probar (b) sea  $<$  un orden total estricto sobre  $\Sigma \cup \Sigma_p$ . Sea  $P$  el siguiente predicado

$$\lambda x [Lt(x) > 0 \wedge (\forall t \in \mathbf{N})_{t \leq Lt(x)} *^<((x)_t) \in \text{Ins}^\Sigma \wedge (\forall t \in \mathbf{N})_{t \leq Lt(x)} (\forall m \in \mathbf{N}) \neg (L\bar{m} \text{ t-final } *^<((x)_t)) \vee (\exists j \in \mathbf{N})_{j \leq Lt(x)} (\exists \alpha \in (\Sigma \cup \Sigma_p) - Num) L\bar{m} \alpha \text{ t-}$$

Notese que  $D_P = \mathbf{N}$  y que  $P(x) = 1$  sii  $Lt(x) > 0$ ,  $*^<((x)_t) \in \text{Ins}^\Sigma$ , para cada  $t = 1, \dots, Lt(x)$  y ademas  $\bigcap_{t=1}^{t=Lt(x)} *^<((x)_t) \in \text{Pro}^\Sigma$ . Para ver que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r. solo nos falta acotar el cuantificador  $(\forall m \in \mathbf{N})$  de la expresion lambda que define a  $P$ . Ya que nos interesan los valores de  $m$  para los cuales  $\bar{m}$  es posiblemente una subpalabra de alguna de las palabras  $*^<((x)_j)$ , el Lema 58 nos dice que una cota posible es  $10^{\max\{|*^<((x)_j)|: 1 \leq j \leq Lt(x)\}} - 1$ . Dejamos al lector los detalles de la Proof de que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea

$$Q = \lambda x \alpha [P(x) \wedge \alpha = \bigcap_{t=1}^{t=Lt(x)} *^<((x)_t)]$$

Note que  $D_Q = \mathbf{N} \times (\Sigma \cup \Sigma_p)^*$ . Claramente  $Q$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Ademas note que  $D_{M(Q)} = \text{Pro}^\Sigma$ . Notese que para  $\mathcal{P} \in \text{Pro}^\Sigma$ , tenemos que  $M(Q)(\mathcal{P})$  es aquel numero tal que pensado como infinitupla (via mirar su secuencia de exponentes) codifica la secuencia de instrucciones que forman a  $\mathcal{P}$ . Es decir  $M(Q)(\mathcal{P}) = \langle \#^<(I_1^{\mathcal{P}}), \#^<(I_2^{\mathcal{P}}), \dots, \#^<(I_{n(\mathcal{P})}^{\mathcal{P}}), 0, 0, \dots \rangle$

Por (b) del Lema 43,  $M(Q)$  es  $(\Sigma \cup \Sigma_p)$ -p.r. ya que para cada  $\mathcal{P} \in \text{Pro}^\Sigma$  tenemos que

$$\begin{aligned} M(Q)(\mathcal{P}) &= \langle \#^<(I_1^{\mathcal{P}}), \#^<(I_2^{\mathcal{P}}), \dots, \#^<(I_{n(\mathcal{P})}^{\mathcal{P}}), 0, 0, \dots \rangle \\ &= \prod_{i=1}^{n(\mathcal{P})} pr(i) \#^<(I_1^{\mathcal{P}}) \\ &\leq \prod_{i=1}^{|\mathcal{P}|} pr(i) \#^<(\mathcal{P}) \end{aligned}$$

$$\begin{aligned} \lambda \mathcal{P} [n(\mathcal{P})] &= \lambda x [Lt(x)] \circ M(Q) \\ \lambda i \mathcal{P} [I_i^{\mathcal{P}}] &= *^< \circ g \circ (p_1^{1,1}, M(Q) \circ p_2^{1,1}) \end{aligned}$$

donde  $g = C_0^{1,1} \mid_{\{0\} \times \omega} \cup \lambda i x [(x)_i]$ , lo cual dice que  $\lambda \mathcal{P} [n(\mathcal{P})]$  y  $\lambda i \mathcal{P} [I_i^{\mathcal{P}}]$  son funciones  $(\Sigma \cup \Sigma_p)$ -p.r..  $\square$

**Lemma 62:** Dado un orden total estricto  $<$  sobre  $\Sigma \cup \Sigma_p$ , las funciones  $s$ ,  $S_\#$  y  $S_*$  son  $(\Sigma \cup \Sigma_p)$ -p.r..

**Proof:** Necesitaremos algunas funciones  $(\Sigma \cup \Sigma_p)$ -p.r.. Dada una instruccion  $I$  en la cual al menos ocurre una variable, usaremos  $\#Var1(I)$  para denotar el numero de la primer variable que ocurre en  $I$ . Por ejemplo

$$\#Var1(L\bar{n} \text{ IF } N\bar{k} \neq 0 \text{ GOTO } L\bar{m}) = k$$

Notese que  $\lambda I [\#Var1(I)]$  tiene dominio igual a  $\text{Ins}^\Sigma - L$ , donde  $L$  es la union de los siguientes conjuntos

$$\{\text{GOTOL}\bar{m} : m \in \mathbf{N}\} \cup \{\text{L}\bar{k}\text{GOTOL}\bar{m} : k, m \in \mathbf{N}\} \cup \{\text{SKIP}\} \cup \{\text{L}\bar{k}\text{SKIP} : k \in \mathbf{N}\}$$

Dada una instruccion  $I$  en la cual ocurren dos variables, usaremos  $\#Var2(I)$  para denotar el numero de la segunda variable que ocurre en  $I$ . Por ejemplo  $\#Var2(N\bar{k} \leftarrow N\bar{m}) = m$

Notese que el dominio de  $\lambda I[\#Var2(I)]$  es igual a la union de los siguientes conjuntos

$$\{N\bar{k} \leftarrow N\bar{m} : k, m \in \mathbf{N}\} \cup \{L\bar{j} \ N\bar{k} \leftarrow N\bar{m} : j, k, m \in \mathbf{N}\}$$

$$\{P\bar{k} \leftarrow P\bar{m} : k, m \in \mathbf{N}\} \cup \{L\bar{j} \ P\bar{k} \leftarrow P\bar{m} : j, k, m \in \mathbf{N}\}$$

Ademas notese que para una instruccion  $I$  tenemos que

$$\begin{aligned} \#Var1(I) &= \min_k (N\bar{k} \leftarrow \text{ocu } I \vee N\bar{k} \neq \text{ocu } I) \\ \#Var2(I) &= \min_k (N\bar{k} \text{ t-final } I \vee N\bar{k} + \text{ocu } I) \end{aligned}$$

Esto nos dice que si llamamos  $P$  al predicado  $\lambda k \alpha [\alpha \in \text{Ins}^\Sigma \wedge (N\bar{k} \leftarrow \text{ocu } \alpha \vee N\bar{k} \neq \text{ocu } \alpha \vee P\bar{k} \leftarrow \text{ocu } \alpha]$  entonces  $\lambda I[\#Var1(I)] = M(P)$  por lo cual  $\lambda I[\#Var1(I)]$  es  $(\Sigma \cup \Sigma_p)$ -p.r. Similarmente se

puede ver que  $\lambda I[\#Var2(I)]$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea  $F_- : \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j) \rightarrow \langle (x)_1, \dots, (x)_{j-1}, (x)_j - 1, (x)_{j+1}, \dots \rangle$$

Ya que  $F_-(x, j) = \begin{cases} Q(x, pr(j)) & \text{si } pr(j) \text{ divide } x \\ x & \text{caso contrario} \end{cases}$

tenemos que  $F_-$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea  $F_+ : \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j) \rightarrow \langle (x)_1, \dots, (x)_{j-1}, (x)_j + 1, (x)_{j+1}, \dots \rangle$$

Ya que  $F_+(x, j) = x.pr(j)$  tenemos que  $F_+$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea  $F_{\leftarrow} : \mathbf{N} \times \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j, k) \rightarrow \langle (x)_1, \dots, (x)_{j-1}, (x)_j, (x)_{j+1}, \dots \rangle$$

Ya que  $F_{\leftarrow}(x, j, k) = Q(x, pr(j)^{(x)_j}).pr(j)^{(x)_k}$  tenemos que  $F_{\leftarrow}$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Sea  $F_0 : \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j) \rightarrow \langle (x)_1, \dots, (x)_{j-1}, \#^<(*^<((x)_j)) \rangle$$

Es facil ver que  $F_0$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. Para cada  $a \in \Sigma$ , sea  $F_a : \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j) \rightarrow \langle (x)_1, \dots, (x)_{j-1}, \#^<(*^<((x)_j)) \rangle$$

Es facil ver que  $F_a$  es  $(\Sigma \cup \Sigma_p)$ -p.r.. En forma similar puede ser probado que  $F_{\sim} : \mathbf{N} \times \mathbf{N} \rightarrow \omega$

$$(x, j) \rightarrow \langle (x)_1, \dots, (x)_j \rangle$$

es  $(\Sigma \cup \Sigma_p)$ -p.r. Dado  $(i, x, y, \mathcal{P}) \in \omega \times \mathbf{N} \times \mathbf{N} \times \text{Pro}^\Sigma$ , tenemos varios casos en los cuales los valores  $s(i, x, y, \mathcal{P})$ ,  $S_{\#}(i, x, y, \mathcal{P})$  y  $S_*(i, x, y, \mathcal{P})$  pueden ser obtenidos usando las funciones antes definidas:

- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i \\ (1) \text{ CASO } i = 0 \vee i > n(\mathcal{P}). \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_*(i, x, y, \mathcal{P}) &= y \\ (2) \text{ CASO } (\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = N\bar{j} \leftarrow N\bar{j} + 1. \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= F_+(x, \#Var1(I_i^{\mathcal{P}})) \\ S_*(i, x, y, \mathcal{P}) &= y \\ s(i, x, y, \mathcal{P}) &= i + 1 \\ (3) \text{ CASO } (\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = N\bar{j} \leftarrow N\bar{j} - 1. \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= F_-(x, \#Var1(I_i^{\mathcal{P}})) \\ S_*(i, x, y, \mathcal{P}) &= y \\ s(i, x, y, \mathcal{P}) &= i + 1 \\ (4) \text{ CASO } (\exists j, k \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = N\bar{j} \leftarrow N\bar{k}. \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= F_{\leftarrow}(x, \#Var1(I_i^{\mathcal{P}}), \#Var2(I_i^{\mathcal{P}})) \\ S_*(i, x, y, \mathcal{P}) &= y \\ s(i, x, y, \mathcal{P}) &= i + 1 \\ (5) \text{ CASO } (\exists j, k \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = N\bar{j} \leftarrow 0. \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= F_0(x, \#Var1(I_i^{\mathcal{P}})) \\ S_*(i, x, y, \mathcal{P}) &= y \\ s(i, x, y, \mathcal{P}) &= i \\ (6) \text{ CASO } (\exists j, m \in \omega) \left( \text{Bas}(I_i^{\mathcal{P}}) = \text{IF } N\bar{j} \neq 0 \text{ GOTO } L\bar{m} \wedge (x)_j = 0 \right). \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_*(i, x, y, \mathcal{P}) &= y \\ s(i, x, y, \mathcal{P}) &= m \\ (7) \text{ CASO } (\exists j, m \in \omega) \left( \text{Bas}(I_i^{\mathcal{P}}) = \text{IF } N\bar{j} \neq 0 \text{ GOTO } L\bar{m} \wedge (x)_j \neq 0 \right). \text{ Entonces } S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_*(i, x, y, \mathcal{P}) &= y \end{aligned}$$

- (8) CASO  $(\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = P\bar{j} \leftarrow P\bar{j}.a$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i + 1 \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= F_a(y, \#Var1(I_i^{\mathcal{P}})) \end{aligned}$$
- (9) CASO  $(\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = P\bar{j} \leftarrow \neg P\bar{j}$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i + 1 \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= F_{\neg}(y, \#Var1(I_i^{\mathcal{P}})) \end{aligned}$$
- (10) CASO  $(\exists j, k \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = P\bar{j} \leftarrow P\bar{k}$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i + 1 \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= F_{\leftarrow}(y, \#Var1(I_i^{\mathcal{P}}), \#Var2(I_i^{\mathcal{P}})) \end{aligned}$$
- (11) CASO  $(\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = P\bar{j} \leftarrow \varepsilon$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i + 1 \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= F_0(y, \#Var1(I_i^{\mathcal{P}})) \end{aligned}$$
- (12) CASO  $(\exists j, m \in \omega)(\exists a \in \Sigma) \left( \text{Bas}(I_i^{\mathcal{P}}) = \text{IF } P\bar{j} \text{ BEGINS } a \text{ GOTO } L\bar{m} \wedge [*^<((y)_j)]_1 \neq a \right)$ .
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= i + 1 \\ \text{Entonces } S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= y \end{aligned}$$
- (13) CASO  $(\exists j, m \in \omega)(\exists a \in \Sigma) \left( \text{Bas}(I_i^{\mathcal{P}}) = \text{IF } P\bar{j} \text{ BEGINS } a \text{ GOTO } L\bar{m} \wedge [*^<((y)_j)]_1 = a \right)$ .
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= \min_l \left( \text{Lab}(I_l^{\mathcal{P}}) \neq \varepsilon \wedge \text{Lab}(I_l^{\mathcal{P}}) \text{ t-final } I_i^{\mathcal{P}} \right) \\ \text{Entonces } S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= y \end{aligned}$$
- (14) CASO  $(\exists j \in \omega) \text{ Bas}(I_i^{\mathcal{P}}) = \text{GOTO } L\bar{j}$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= \min_l \left( \text{Lab}(I_l^{\mathcal{P}}) \neq \varepsilon \wedge \text{Lab}(I_l^{\mathcal{P}}) \right) \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= y \end{aligned}$$
- (15) CASO  $\text{Bas}(I_i^{\mathcal{P}}) = \text{SKIP}$ . Entonces
- $$\begin{aligned} s(i, x, y, \mathcal{P}) &= k + 1 \\ S_{\#}(i, x, y, \mathcal{P}) &= x \\ S_{*}(i, x, y, \mathcal{P}) &= y \end{aligned}$$

O sea que los casos anteriores nos permiten definir conjuntos  $S_1, \dots, S_{15}$ , los cuales son disjuntos de a pares y cuya union da el conjunto  $\omega \times \mathbf{N} \times \mathbf{N} \times \text{Pro}^{\Sigma}$ , de manera que cada una de las funciones  $s, S_{\#}$  y  $S_{*}$  pueden escribirse como union disjunta de funciones  $(\Sigma \cup \Sigma_p)$ -p.r. restringidas respectivamente a los conjuntos  $S_1, \dots, S_{15}$ . Ya que los conjuntos  $S_1, \dots, S_{15}$  son  $(\Sigma \cup \Sigma_p)$ -p.r. el Lema 35 nos dice que  $s, S_{\#}$  y  $S_{*}$  lo son.  $\square$

**Lemma 63:** Sean: ... hacer!!

**Proof:**Hacer

**Proposition 64:** Sean  $n, m \leq 0$ , las funciones  $i^{n,m}, E_{\#j}^{n,m}, j = 1, 2, \dots$  son  $\Sigma \cup \Sigma_p$ -PR.

**Proof:**Sea  $<$  un orden total estricto sobre  $\Sigma \cup \Sigma_p$  y sean  $s, S_{\#}, S$  las funciones previamente definidas en el Lemma 62, definamos:

$$\begin{aligned} C_{\#}^{n,m} &= \lambda t \vec{x} \vec{\alpha} \mathcal{P} \left[ \left\langle E_{\#1}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), E_{\#2}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), \dots \right\rangle \right] \\ C_{*}^{n,m} &= \lambda t \vec{x} \vec{\alpha} \mathcal{P} \left[ \left\langle \#^<(E_{*1}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P})), \#^<(E_{*2}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P})), \dots \right\rangle \right] \end{aligned}$$

$$\begin{aligned} i^{n,m}(0, \vec{x}, \vec{\alpha}, \mathcal{P}) &= 1 \\ C_{\#}^{n,m}(0, \vec{x}, \vec{\alpha}, \mathcal{P}) &= \langle x_1, \dots, x_n \rangle \\ C_{*}^{n,m}(0, \vec{x}, \vec{\alpha}, \mathcal{P}) &= \langle \#^<(\alpha_1), \dots, \#^<(\alpha_m) \rangle \\ \text{Notese que } i^{n,m}(t+1, \vec{x}, \vec{\alpha}, \mathcal{P}) &= s(i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{\#}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{*}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P})) \\ C_{\#}^{n,m}(t+1, \vec{x}, \vec{\alpha}, \mathcal{P}) &= S_{\#}(i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{\#}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{*}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P})) \\ C_{*}^{n,m}(t+1, \vec{x}, \vec{\alpha}, \mathcal{P}) &= S_{*}(i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{\#}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}), C_{*}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P})) \end{aligned}$$

Por el Lema 63 tenemos que  $i^{n,m}, C_{\#}^{n,m}$  y  $C_{*}^{n,m}$  son  $(\Sigma \cup \Sigma_p)$ -p.r.. Adem as notese que

$$\begin{aligned} E_{\#j}^{n,m} &= \lambda t \vec{x} \vec{\alpha} \mathcal{P} \left[ (C_{\#}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}))_j \right] \\ E_{*j}^{n,m} &= \lambda t \vec{x} \vec{\alpha} \mathcal{P} \left[ *^<((C_{*}^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}))_j) \right] \end{aligned}$$

por lo cual las funciones  $E_{\#j}^{n,m}$ ,  $E_{*j}^{n,m}$ ,  $j = 1, 2, \dots$ , son  $(\Sigma \cup \Sigma_p)$ -p.r.  $\square$

**Theorem 65:** Las funciones  $\Phi_{\#}^{n,m}$  y  $\Phi_{*}^{n,m}$  son  $(\Sigma \cup \Sigma_p)$ -recursivas.

**Proof:** Veremos que  $\Phi_{\#}^{n,m}$  es  $(\Sigma \cup \Sigma_p)$ -recursiva. Sea  $H$  el predicado  $(\Sigma \cup \Sigma_p)$ -mixto

$$\lambda t \vec{x} \vec{\alpha} \mathcal{P} [i^{n,m}(t, x_1, \dots, x_n, \alpha_1, \dots, \alpha_m, \mathcal{P}) = n(\mathcal{P}) + 1].$$

Note que  $D_H = \omega^{n+1} \times \Sigma^{*m} \times \text{Pro}^{\Sigma}$ . Ya que the funciones  $i^{n,m}$  y  $\lambda \mathcal{P} [n(\mathcal{P})]$  son  $(\Sigma \cup \Sigma_p)$ -p.r.,  $H$  lo es. Notar que  $D_{M(H)} = D_{\Phi_{\#}^{n,m}}$ . Ademas para  $(\vec{x}, \vec{\alpha}, \mathcal{P}) \in D_{M(H)}$ , tenemos que  $M(H)(\vec{x}, \vec{\alpha}, \mathcal{P})$  es la menor cantidad de pasos necesarios para que  $\mathcal{P}$  termine partiendo del estado  $((x_1, \dots, x_n, 0, 0, \dots), (\alpha_1, \dots, \alpha_m, \varepsilon, \varepsilon, \dots))$ . Ya que  $H$  es  $(\Sigma \cup \Sigma_p)$ -p.r., tenemos que  $M(H)$  es  $(\Sigma \cup \Sigma_p)$ -r.. Notese que para  $(\vec{x}, \vec{\alpha}, \mathcal{P}) \in D_{M(H)} = D_{\Phi_{\#}^{n,m}}$  tenemos que  $\Phi_{\#}^{n,m}(\vec{x}, \vec{\alpha}, \mathcal{P}) = E_{\#1}^{n,m}(M(H)(\vec{x}, \vec{\alpha}, \mathcal{P}), \vec{x}, \vec{\alpha}, \mathcal{P})$

lo cual con un poco mas de trabajo nos permite probar que  $\Phi_{\#}^{n,m} = E_{\#1}^{n,m} \circ (M(H), p_1^{n,m+1}, \dots, p_{n+m+1}^{n,m+1})$

Ya que la funcion  $E_{\#1}^{n,m}$  es  $(\Sigma \cup \Sigma_p)$ -r., lo es  $\Phi_{\#}^{n,m}$ .  $\square$

**Corollary 66:** Si  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -computable, entonces  $f$  es  $\Sigma$ -recursiva.

**Proof:** Haremos el caso  $O = \Sigma^*$ . Sea  $\mathcal{P}_0$  un programa que compute a  $f$ . Primero veremos que  $f$  es  $(\Sigma \cup \Sigma_p)$ -recursiva. Note que

$$f = \Phi_{*}^{n,m} \circ (p_1^{n,m}, \dots, p_{n+m}^{n,m}, C_{\mathcal{P}_0}^{n,m})$$

donde cabe destacar que  $p_1^{n,m}, \dots, p_{n+m}^{n,m}$  son las proyecciones respecto del alfabeto  $\Sigma \cup \Sigma_p$ , es decir que tienen dominio  $\omega^n \times (\Sigma \cup \Sigma_p)^{*m}$ . Ya que  $\Phi_{*}^{n,m}$  es  $(\Sigma \cup \Sigma_p)$ -recursiva tenemos que  $f$  lo es. O sea que el Teorema 51 nos dice que  $f$  es  $\Sigma$ -recursiva.  $\square$

**Tesis de Church:** Toda funcion  $\Sigma$ -efectivamente computable es  $\Sigma$ -recursiva.

**Corollary 67:** Si  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -recursiva, entonces existe un predicado  $\Sigma$ -p.r.  $P : \mathbf{N} \times \omega^n \times \Sigma^{*m} \rightarrow \omega$  y una funcion  $\Sigma$ -p.r.  $g : \mathbf{N} \rightarrow O$  tales que  $f = g \circ M(P)$ .

**Proof:** Supongamos que  $O = \Sigma^*$ . Sea  $\mathcal{P}_0$  un programa el cual compute a  $f$ . Sea  $<$  un orden total estricto sobre  $\Sigma$ . Note que podemos tomar

$$P = \lambda t \vec{x} \vec{\alpha} [i^{n,m}((t)_1, \vec{x}, \vec{\alpha}, \mathcal{P}_0) = n(\mathcal{P}_0) + 1 \wedge (t)_2 = \#^{<}(E_{*1}^{n,m}((t)_1, \vec{x}, \vec{\alpha}, \mathcal{P}_0))]$$

$$g = \lambda t [*^{<}((t)_2)].$$

(Justifique por que  $P$  es  $\Sigma$ -p.r..)  $\square$

**Lemma 68:** Supongamos  $f_i : D_{f_i} \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$ ,  $i = 1, \dots, k$ , son funciones  $\Sigma$ -recursivas tales que  $D_{f_i} \cap D_{f_j} = \emptyset$  para  $i \neq j$ . Entonces la funcion  $f_1 \cup \dots \cup f_k$  es  $\Sigma$ -recursiva.

**Proof:** Probaremos el caso  $k = 2$  y  $O = \Sigma^*$ . Sean  $\mathcal{P}_1$  y  $\mathcal{P}_2$  programas que computen las funciones  $f_1$  y  $f_2$ , respectivamente. Sean

$$P_1 = \lambda t \vec{x} \vec{\alpha} [i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}_1) = n(\mathcal{P}_1) + 1]$$

$$P_2 = \lambda t \vec{x} \vec{\alpha} [i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}_2) = n(\mathcal{P}_2) + 1]$$

Notese que  $D_{P_1} = D_{P_2} = \omega \times \omega^n \times \Sigma^{*m}$  y que  $P_1$  y  $P_2$  son  $(\Sigma \cup \Sigma_p)$ -p.r.. Ya que son  $\Sigma$ -mixtos, el Teorema 51 nos dice que son  $\Sigma$ -p.r.. Tambien notese que  $D_{M((P_1 \vee P_2))} = D_{f_1} \cup D_{f_2}$ .

$$\begin{aligned} \text{Definamos } g_1 &= \lambda \vec{x} \vec{\alpha} \left[ E_{*1}^{n,m}(M((P_1 \vee P_2))(\vec{x}, \vec{\alpha}), \vec{x}, \vec{\alpha}, \mathcal{P}_1)^{P_i(M((P_1 \vee P_2))(\vec{x}, \vec{\alpha}), \vec{x}, \vec{\alpha})} \right] \\ g_2 &= \lambda \vec{x} \vec{\alpha} \left[ E_{*1}^{n,m}(M((P_1 \vee P_2))(\vec{x}, \vec{\alpha}), \vec{x}, \vec{\alpha}, \mathcal{P}_2)^{P_i(M((P_1 \vee P_2))(\vec{x}, \vec{\alpha}), \vec{x}, \vec{\alpha})} \right] \end{aligned}$$

Notese que  $g_1$  y  $g_2$  son  $\Sigma$ -recursivas y que  $D_{g_1} = D_{g_2} = D_{f_1} \cup D_{f_2}$ , Ademas notese que

$$g_1(\vec{x}, \vec{\alpha}) = \begin{cases} f_1(\vec{x}, \vec{\alpha}) & \text{si } (\vec{x}, \vec{\alpha}) \in D_{f_1} \\ \varepsilon & \text{caso contrario} \end{cases}$$

$$g_2(\vec{x}, \vec{\alpha}) = \begin{cases} f_2(\vec{x}, \vec{\alpha}) & \text{si } (\vec{x}, \vec{\alpha}) \in D_{f_2} \\ \varepsilon & \text{caso contrario} \end{cases}$$

O sea que  $f_1 \cup f_2 = \lambda \alpha \beta [\alpha \beta] \circ (g_1, g_2)$  es  $\Sigma$ -recursiva.  $\square$

**Lemma 69:** Supongamos  $\Sigma \supseteq \Sigma_p$ . Entonces  $\text{Halt}^{\Sigma}$  es no  $\Sigma$ -recursivo.

**Proof:** Supongamos  $\text{Halt}^{\Sigma}$  es  $\Sigma$ -recursivo y por lo tanto  $\Sigma$ -computable. Por la proposicion de existencia de macros tenemos que hay un macro

[IF  $Halt^\Sigma(W1)$  GOTO A1]

Sea  $\mathcal{P}_0$  el siguiente programa de  $\mathcal{S}^\Sigma$  L1 [IF  $Halt^\Sigma(P1)$  GOTO L1]

Note que -  $\mathcal{P}_0$  termina partiendo desde  $((0, 0, \dots), (\mathcal{P}_0, \varepsilon, \varepsilon, \dots))$  sii  $Halt^\Sigma(\mathcal{P}_0) = 0$ , lo cual produce una contradiccion si tomamos en (\*)  $\mathcal{P} = \mathcal{P}_0$ .  $\square$

**Theorem 70:** Sea  $S \subseteq \omega^n \times \Sigma^{*m}$ . Entonces  $S$  es  $\Sigma$ -efectivamente enumerable sii  $S$  es  $\Sigma$ -recursivamente enumerable

**Proof:**( $\Rightarrow$ ) Use la Tesis de Church.

( $\Leftarrow$ ) Use el Theorem 42.  $\square$

**Theorem 71:** Dado  $S \subseteq \omega^n \times \Sigma^{*m}$ , son equivalentes (1)  $S$  es  $\Sigma$ -recursivamente enumerable (2)  $S = I_F$ , para alguna  $F : D_F \subseteq \omega^k \times \Sigma^{*l} \rightarrow \omega^n \times \Sigma^{*m}$  tal que cada  $F_i$  es  $\Sigma$ -recursiva. (3)  $S = D_f$ , para alguna funcion  $\Sigma$ -recursiva  $f$  (4)  $S = \emptyset$  o  $S = I_F$ , para alguna  $F : \omega \rightarrow \omega^n \times \Sigma^{*m}$  tal que cada  $F_i$  es  $\Sigma$ -p.r.

**Proof:**(2) $\Rightarrow$ (3). Para  $i = 1, \dots, n + m$ , sea  $\mathcal{P}_i$  un programa el cual computa a  $F_i$  y sea  $<$  un orden total estricto sobre  $\Sigma$ . Sea  $P : \mathbf{N} \times \omega^n \times \Sigma^{*m} \rightarrow \omega$  dado por  $P(t, \vec{x}, \vec{\alpha}) = 1$  sii se cumplen las siguientes condiciones

$$\begin{aligned} i^{k,l}(((t)_{k+l+1}, (t)_1, \dots, (t)_k, *^<((t)_{k+1}), \dots, *^<((t)_{k+l})), \mathcal{P}_1) &= n(\mathcal{P}_1) + 1 \\ &\vdots \\ i(((t)_{k+l+1}, (t)_1 \dots (t)_k, *^<((t)_{k+1}) \dots *^<((t)_{k+l})), \mathcal{P}_{n+m}) &= n(\mathcal{P}_{n+m}) + 1 \\ E_{\#1}^{k,l}(((t)_{k+l+1}, (t)_1, \dots, (t)_k, *^<((t)_{k+1}), \dots, *^<((t)_{k+l})), \mathcal{P}_1) &= x_1 \\ &\vdots \\ E_{\#1}^{k,l}(((t)_{k+l+1}, (t)_1, \dots, (t)_k, *^<((t)_{k+1}), \dots, *^<((t)_{k+l})), \mathcal{P}_n) &= x_n \\ E_{*1}^{k,l}(((t)_{k+l+1}, (t)_1, \dots, (t)_k, *^<((t)_{k+1}), \dots, *^<((t)_{k+l})), \mathcal{P}_{n+1}) &= \alpha_1 \\ &\vdots \\ E_{*1}^{k,l}(((t)_{k+l+1}, (t)_1, \dots, (t)_k, *^<((t)_{k+1}), \dots, *^<((t)_{k+l})), \mathcal{P}_{n+m}) &= \alpha_m \end{aligned}$$

Note que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r. y por lo tanto  $P$  es  $\Sigma$ -p.r.. Pero entonces  $M(P)$  es  $\Sigma$ -r. lo cual nos dice que se cumple (3) ya que  $D_{M(P)} = I_F = S$ . (3) $\Rightarrow$ (4). Supongamos  $S \neq \emptyset$ . Sea  $(z_1, \dots, z_n, \gamma_1, \dots, \gamma_m) \in S$  fijo. Sea  $\mathcal{P}$  un programa el cual compute a  $f$  y sea  $<$  un orden total estricto sobre  $\Sigma$ . Sea  $P : \mathbf{N} \rightarrow \omega$  dado por  $P(x) = 1$  sii

$$i^{n,m}((x)_{n+m+1}, (x)_1, \dots, (x)_n, *^<((x)_{n+1}), \dots, *^<((x)_{n+m})), \mathcal{P}) = n(\mathcal{P}) + 1$$

Es facil ver que  $P$  es  $(\Sigma \cup \Sigma_p)$ -p.r. por lo cual es  $\Sigma$ -p.r.. Sea  $\bar{P} = P \cup C_0^{1,0} \upharpoonright_{\{0\}}$ . Para  $i = 1, \dots, n$ , definamos  $F_i : \omega \rightarrow \omega$  de la siguiente manera  $F_i(x) = \begin{cases} (x)_i & \text{si } \bar{P}(x) = 1 \\ z_i & \text{si } \bar{P}(x) \neq 1 \end{cases}$

Para  $i = n + 1, \dots, n + m$ , definamos  $F_i : \omega \rightarrow \Sigma^*$  de la siguiente manera  $F_i(x) = \begin{cases} *^<((x)_i) & \text{si } \bar{P}(x) = 1 \\ \gamma_{i-n} & \text{si } \bar{P}(x) \neq 1 \end{cases}$

Por el lema de division por casos, cada  $F_i$  es  $\Sigma$ -p.r.. Es facil ver que  $F = (F_1, \dots, F_{n+m})$  cumple (4).  $\square$

**Corollary 72:** Supongamos  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -recursiva y  $S \subseteq D_f$  es  $\Sigma$ -r.e., entonces  $f \upharpoonright_S$  es  $\Sigma$ -recursiva.

**Proof:**Supongamos  $O = \Sigma^*$ . Por el Theorem anterior  $S = D_g$ , para alguna funcion  $\Sigma$ -recursiva  $g$ . Notese que componiendo adecuadamente podemos suponer que  $I_g = \{\varepsilon\}$ . O sea que tenemos  $f \upharpoonright_S = \lambda \alpha \beta [\alpha \beta] \circ (f, g)$ .  $\square$

**Corollary 73:** Supongamos  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -recursiva y  $S \subseteq I_f$  es  $\Sigma$ -r.e., entonces  $f^{-1}(S) = \{(\vec{x}, \vec{\alpha}) : f(\vec{x}, \vec{\alpha}) \in S\}$  es  $\Sigma$ -r.e..

**Proof:**Por el Theorem anterior  $S = D_g$ , para alguna funcion  $\Sigma$ -recursiva  $g$ . O sea que  $f^{-1}(S) = D_{g \circ f}$  es  $\Sigma$ -r.e..  $\square$

**Corollary 74:** Supongamos  $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$  son conjuntos  $\Sigma$ -r.e.. Entonces  $S_1 \cap S_2$  es  $\Sigma$ -r.e..

**Proof:** Por el Theorem anterior  $S_i = D_{g_i}$ , con  $g_1, g_2$  funciones  $\Sigma$ -recursivas. Notese que podemos suponer que  $I_{g_1}, I_{g_2} \subseteq \omega$  por lo que  $S_1 \cap S_2 = D_{\lambda xy[xy] \circ (g_1, g_2)}$  es  $\Sigma$ -r.e..  $\square$

**Corollary 75:** Supongamos  $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$  son conjuntos  $\Sigma$ -r.e.. Entonces  $S_1 \cup S_2$  es  $\Sigma$ -r.e..

**Proof:** Supongamos  $S_1 \neq \emptyset \neq S_2$ . Sean  $F, G : \omega \rightarrow \omega^n \times \Sigma^{*m}$  tales que  $I_F = S_1$ ,  $I_G = S_2$  y las funciones  $F_i$ 's y  $G_i$ 's son  $\Sigma$ -recursivas. Sean  $f = \lambda x [Q(x, 2)]$  y  $g = \lambda x [Q(x-1, 2)]$ . Sea  $H : \omega \rightarrow \omega^n \times \Sigma^{*m}$  dada por

$$H_i = (F_i \circ f)|_{\{x:x \text{ es par}\}} \cup (G_i \circ g)|_{\{x:x \text{ es impar}\}}$$

Por el Corollary 72 y el Lema 68, cada  $H_i$  es  $\Sigma$ -recursiva. Ya que  $I_H = S_1 \cup S_2$ , tenemos que  $S_1 \cup S_2$  es  $\Sigma$ -r.e.  $\square$

**Theorem 76:** Sea  $S \subseteq \omega^n \times \Sigma^{*m}$ . Entonces  $S$  es  $\Sigma$ -efectivamente computable sii  $S$  es  $\Sigma$ -recursivo **Proof:** ( $\Rightarrow$ ) Use la Tesis de Church.

( $\Leftarrow$ ) Use el Teorema 42.  $\square$

**Theorem 77:** Sea  $S \subseteq \omega^n \times \Sigma^{*m}$ . Son equivalentes (a)  $S$  es  $\Sigma$ -recursivo (b)  $S$  y  $(\omega^n \times \Sigma^{*m}) - S$  son  $\Sigma$ -recursivamente enumerables

**Proof:** (a) $\Rightarrow$ (b). Note que  $S = D_{Pred \circ \chi_S}$ .

(b) $\Rightarrow$ (a). Note que  $\chi_S = C_1^{n,m}|_S \cup C_0^{n,m}|_{\omega^n \times \Sigma^{*m} - S}$ .  $\square$

**Lemma 78:** Supongamos que  $\Sigma \supseteq \Sigma_p$ . Entonces  $A = \{\mathcal{P} \in \text{Pro}^\Sigma : \text{Halt}^\Sigma(\mathcal{P})\}$

es  $\Sigma$ -r.e. y no es  $\Sigma$ -recursivo. Mas aun el conjunto  $N = \{\mathcal{P} \in \text{Pro}^\Sigma : \neg \text{Halt}^\Sigma(\mathcal{P})\}$  no es  $\Sigma$ -r.e.

**Proof:** Sea  $P = \lambda t \mathcal{P} [i^{0,1}(t, \mathcal{P}, \mathcal{P}) = n(\mathcal{P}) + 1]$ . Note que  $P$  es  $\Sigma$ -p.r. por lo que  $M(P)$  es  $\Sigma$ -r.. Ademas note que  $D_{M(P)} = A$ , lo cual implica que  $A$  es  $\Sigma$ -r.e.. Ya que  $\text{Halt}^\Sigma$  es no  $\Sigma$ -recursivo (Lema 69) y

$$\text{Halt}^\Sigma = C_1^{0,1}|_A \cup C_0^{0,1}|_N$$

el Lema 68 nos dice que  $N$  no es  $\Sigma$ -r.e.. Finalmente supongamos  $A$  es  $\Sigma$ -recursivo. Entonces el conjunto  $N = (\Sigma^* - A) \cap \text{Pro}^\Sigma$

deberia serlo, lo cual es absurdo.  $\square$



## 5 Máquinas de Turing

**Lemma 52.** *El predicado  $\lambda n d d' [d \vdash d']$  es  $(\Gamma \cup Q)$ -PR.*

*Proof.* Note que  $D_{\lambda d d' [d \vdash d']} = Des \times Des$ . También nótese que los predicados

$$\begin{aligned} \lambda p \sigma q \gamma [(p, \sigma, L) \in \delta(q, \gamma)] \\ \lambda p \sigma q \gamma [(p, \sigma, R) \in \delta(q, \gamma)] \\ \lambda p \sigma q \gamma [(p, \sigma, K) \in \delta(q, \gamma)] \end{aligned}$$

son  $(\Gamma \cup Q)$ -p.r. ya que los tres tienen dominio igual a  $Q \times \Gamma \times Q \times \Gamma$  el cual es finito (Corolario 36). Sea  $P_R : Des \times Des \times \Gamma \times \Gamma^* \times \Gamma^* \times Q \times Q \rightarrow \omega$  definido por  $P_R(d, d', \sigma, \alpha, \beta, p, q) = 1$  sii

$$d = \alpha p \beta \wedge (q, \sigma, R) \in \delta(p, [\beta B]_1) \wedge d' = \alpha \sigma q^\frown \beta$$

Sea  $P_L : Des \times Des \times \Gamma \times \Gamma^* \times \Gamma^* \times Q \times Q \rightarrow \omega$  definido por  $P_L(d, d', \sigma, \alpha, \beta, p, q) = 1$  sii

$$d = \alpha p \beta \wedge (q, \sigma, L) \in \delta(p, [\beta B]_1) \wedge \alpha \neq \varepsilon \wedge d' = [\alpha^\frown q [\alpha]_{|\alpha|} \sigma^\frown \beta]$$

Sea  $P_K : Des \times Des \times \Gamma \times \Gamma^* \times \Gamma^* \times Q \times Q \rightarrow \omega$  definido por  $P_K(d, d', \sigma, \alpha, \beta, p, q) = 1$  sii

$$d = \alpha p \beta \wedge (q, \sigma, K) \in \delta(p, [\beta B]_1) \wedge d' = [\alpha q \sigma^\frown \beta]$$

Se deja al lector la verificación de que estos predicados son  $(\Gamma \cup Q)$ -p.r. Nótese que  $\lambda d d' [d \vdash d']$  es igual al predicado

$$\lambda d d' [(\exists \sigma \in \Gamma)(\exists \alpha, \beta \in \Gamma^*)(\exists p, q \in Q)(P_R \vee P_L \vee P_K)(d, d', \sigma, \alpha, \beta, p, q)]$$

lo cual por el Lema 39 nos dice que  $\lambda d d' [d \vdash d']$  es  $(\Gamma \cup Q)$ -p.r. □

**Proposition 53.**  $\lambda n d d' [d \vdash^n d']$  es  $(\Gamma \cup Q)$ -p.r..

**Theorem 54.** Sea  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$  una máquina de Turing. Entonces  $L(M)$  es  $\Sigma$ -recursivamente enumerable.

*Proof.* Sea  $P$  el siguiente predicado  $(\Gamma \cup Q)$ -mixto

$$\lambda n \alpha \left[ (\exists d \in Des) [q_0 B \alpha] \vdash^n d \wedge St(d) \in F \right]$$

Nótese que  $D_P = \omega \times \Gamma^*$ . Dejamos al lector probar que  $P$  es  $(\Gamma \cup Q)$ -p.r.. Sea  $P' = P \upharpoonright_{\omega \times \Sigma^*}$ . Nótese que  $P'(n, \alpha) = 1$  sii  $\alpha \in L(M)$  atestiguado por una computación de longitud  $n$ . Ya que  $P'$  es  $(\Gamma \cup Q)$ -p.r. (por que?) y además es  $\Sigma$ -mixto, el Teorema 51 nos dice que  $P'$  es  $\Sigma$ -p.r.. Ya que  $L(M) = D_{M(P')}$ , el Teorema 71 nos dice que  $L(M)$  es  $\Sigma$ -r.e. □

**Theorem 55.** Supongamos  $f : S \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -Turing computable. Entonces  $f$  es  $\Sigma$ -recursiva.

*Proof.* Supongamos  $O = \Sigma^*$  y sea  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, \vdash, F)$  una máquina de Turing determinística con unit la cual compute a  $f$ . Sea  $<$  un orden total estricto sobre  $\Gamma \cup Q$ . Sea  $P : \mathbf{N} \times \omega^n \times \Sigma^{*m} \rightarrow \omega$  dado por  $P(x, \vec{x}, \vec{\alpha}) = 1$  sii

$$(\exists q \in Q) [q_0 B \vdash^{x_1} \dots B \vdash^{x_n} B \alpha_1 \dots B \alpha_m] \vdash^{(x)_1} [q B *^{<} ((x)_2)] \wedge \\ (\forall d \in Des)_{|d| \leq |*^{<}((x)_2)|+2} [q B *^{<} ((x)_2)] \not\vdash d$$

Es fácil ver que  $P$  es  $(\Gamma \cup Q)$ -p.r. por lo que  $P$  es  $\Sigma$ -p.r. ya que es  $\Sigma$ -mixto. Nótese que

$$f = \lambda \vec{x} \vec{\alpha} \left[ \left( \min_x P(x, \vec{x}, \vec{\alpha}) \right)_2 \right],$$

lo cual nos dice que  $f$  es  $\Sigma$ -recursiva. □

**Lemma 56.** Sea  $\mathcal{P} \in \text{Pro}^\Sigma$  y sea  $k$  tal que las variables que ocurren en  $\mathcal{P}$  están todas en la lista  $N1, \dots, Nk, P1, \dots, Pk$ . Para cada  $a \in \Sigma \cup \{\vdash\}$ , sea  $\tilde{a}$  un nuevo símbolo. Sea  $\Gamma = \Sigma \cup \{B, \vdash\} \cup \{\tilde{a} : a \in \Sigma \cup \{\vdash\}\}$ . Entonces hay una máquina de Turing determinística con unit  $M = (Q, \Gamma, \Sigma, \delta, q_0, B, \vdash, \{q_f\})$  la cual satisface

(1)  $\delta(q_f, \sigma) = \emptyset$ , para cada  $\sigma \in \Gamma$ .

(2) Cualesquiera sean  $x_1, \dots, x_k \in \omega$  y  $\alpha_1, \dots, \alpha_k \in \Sigma^*$ , el programa  $\mathcal{P}$  se detiene partiendo del estado

$$((x_1, \dots, x_k, 0, \dots), (\alpha_1, \dots, \alpha_k, \varepsilon, \dots))$$

sii  $M$  se detiene partiendo de la descripción instantánea

$$[q_0 B \vdash^{x_1} B \dots B \vdash^{x_k} B \alpha_1 B \dots B \alpha_k B]$$

(3) Si  $x_1, \dots, x_k \in \omega$  y  $\alpha_1, \dots, \alpha_k \in \Sigma^*$  son tales que  $\mathcal{P}$  se detiene partiendo del estado

$$((x_1, \dots, x_k, 0, \dots), (\alpha_1, \dots, \alpha_k, \varepsilon, \dots))$$

y llega al estado

$$((y_1, \dots, y_k, 0, \dots), (\beta_1, \dots, \beta_k, \varepsilon, \dots))$$

entonces

$$[q_0 B \vdash^{x_1} B \dots B \vdash^{x_k} B \alpha_1 B \dots B \alpha_k B] \vdash^* [q_f B \vdash^{y_1} B \dots B \vdash^{y_k} B \beta_1 B \dots B \beta_k B]$$

*Proof.* Dado un estado  $((x_1, \dots, x_k, 0, \dots), (\alpha_1, \dots, \alpha_k, \varepsilon, \dots))$  lo representaremos en la cinta de la siguiente manera

$$B \vdash^{x_1} \dots B \vdash^{x_k} B \alpha_1 \dots B \alpha_k B B B B \dots$$

A continuación describiremos una serie de máquinas las cuales simularan, vía la representación anterior, las distintas clases de instrucciones que pueden ocurrir en  $\mathcal{P}$ . Todas las máquinas definidas tendrán a  $\vdash$  como unit y a  $B$  como blanco, tendrán a  $\Sigma$  como su alfabeto terminal y su alfabeto mayor sera  $\Gamma = \Sigma \cup \{B, \vdash\} \cup \{\tilde{a} : a \in \Sigma \cup \{\vdash\}\}$ . Además tendrán uno o dos estados finales con la propiedad de que si  $q$  es un estado final, entonces  $\delta(q, \sigma) = \emptyset$ , para cada  $\sigma \in \Gamma$ . Esta propiedad es importante ya que nos permitirá concatenar pares de dichas máquinas identificando algún estado final de la primera con el inicial de la segunda.

Para  $1 \leq i \leq k$ , sea  $M_i^+$  una máquina tal que

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_{i-1}} B \mid^{x_i+1} B \mid^{x_{i+1}} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_f
\end{array}$$

Es claro que la máquina  $M_i^+$  simula la instrucción  $N\bar{i} \leftarrow N\bar{i} + 1$ .

Para  $1 \leq i \leq k$ , sea  $M_i^-$  una máquina tal que

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_{i-1}} B \mid^{x_i-1} B \mid^{x_{i+1}} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_f
\end{array}$$

Para  $1 \leq i \leq k$  y  $a \in \Sigma$ , sea  $M_i^a$  una máquina tal que

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_{i-1} B\alpha_i a B\alpha_{i+1} \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_f
\end{array}$$

Para  $1 \leq i \leq k$ , sea  $M_i^\curvearrowright$  una máquina tal que

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_{i-1} B^\curvearrowright \alpha_i B\alpha_{i+1} \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_f
\end{array}$$

Para  $j = 1, \dots, k$ , y  $a \in \Sigma$ , sea  $IF_j^a$  una máquina con dos estados finales  $q_{si}$  y  $q_{no}$  tal que si  $\alpha_j$  comienza con  $a$ , entonces

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_{si}
\end{array}$$

y en caso contrario

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_{no}
\end{array}$$

Análogamente para  $j = 1, \dots, k$ , sea  $IF_j$  una máquina tal que si  $x_j \neq 0$ , entonces

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_{si}
\end{array}$$

y si  $x_j = 0$ , entonces

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_{no}
\end{array}$$

Para  $1 \leq i, j \leq k$ , sea  $M_{i \leftarrow j}^*$  una máquina tal que

$$\begin{array}{ccc}
B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_{i-1} B\alpha_j B\alpha_{i+1} \dots B\alpha_k \\
\uparrow & & \uparrow \\
q_0 & & q_f
\end{array}$$

Para  $1 \leq i, j \leq k$ , sea  $M_{i \leftarrow j}^\#$  una máquina tal que

$$\begin{array}{ccc} B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_{i-1}} B \mid^{x_j} B \mid^{x_{i+1}} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

Para  $1 \leq i \leq k$ , sea  $M_{i \leftarrow 0}$  una máquina tal que

$$\begin{array}{ccc} B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_{i-1}} BB \mid^{x_{i+1}} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

Para  $1 \leq i \leq k$ , sea  $M_{i \leftarrow \varepsilon}$  una máquina tal que

$$\begin{array}{ccc} B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_k & \vdash^* & B \mid^{x_1} \dots B \mid^{x_k} B\alpha_1 \dots B\alpha_{i-1} BB\alpha_{i+1} \dots B\alpha_k \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

Sea

$M_{\text{SKIP}} = (\{q_0, q_f\}, \Gamma, \Sigma, \delta, q_0, B, \mid, \{q_f\})$ ,  
con  $\delta(q_0, B) = \{(q_f, B, K)\}$  y  $\delta = \emptyset$  en cualquier otro caso.

Finalmente sea

$$M_{\text{GOTO}} = (\{q_0, q_{si}, q_{no}\}, \Gamma, \Sigma, \delta, q_0, B, \mid, \{q_{si}, q_{no}\}),$$

con  $\delta(q_0, B) = \{(q_{si}, B, K)\}$  y  $\delta = \emptyset$  en cualquier otro caso.

Para poder hacer concretamente las maquinas recién descriptas deberemos diseñar antes algunas máquinas auxiliares. Para cada  $j \geq 1$ , sea  $D_j$  la máquina descripta en la Figura 1. Notese que

$$\begin{array}{ccc} \alpha B\beta_1 B\beta_2 B \dots B\beta_j B\gamma & \vdash^* & \alpha B\beta_1 B\beta_2 B \dots B\beta_j B\gamma \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

siempre que  $\alpha, \gamma \in \Gamma^*$ ,  $\beta_1, \dots, \beta_j \in (\Gamma - \{B\})^*$ . Análogamente tenemos definidas las maquinas  $I_j$ .

Para  $j \geq 1$ , sea  $TD_j$  una máquina con un solo estado final  $q_f$  y tal que

$$\begin{array}{ccc} \alpha B\gamma & \vdash^* & \alpha BB\gamma \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

cada vez que  $\alpha, \gamma \in \Gamma^*$  y  $\gamma$  tiene exactamente  $j$  ocurrencias de  $B$ . Es decir la máquina  $TD_j$  corre un espacio a la derecha todo el bloque  $\gamma$  y agrega un blanco en el espacio que se genera a la izquierda de dicho bloque. Por ejemplo, para el caso de  $\Sigma = \{\&\}$  podemos tomar  $TD_3$  igual a la máquina de la Figura 3.

Análogamente, para  $j \geq 1$ , sea  $TI_j$  una máquina tal que

$$\begin{array}{ccc} \alpha B\sigma\gamma & \vdash^* & \alpha B\gamma \\ \uparrow & & \uparrow \\ q_0 & & q_f \end{array}$$

cada vez que  $\alpha \in \Gamma^*$ ,  $\sigma \in \Gamma$  y  $\gamma$  tiene exactamente  $j$  ocurrencias de  $B$ . Es decir la máquina  $TI_j$  corre un espacio a la izquierda todo el bloque  $\gamma$  (por lo cual en el lugar de  $\sigma$  queda el primer símbolo de  $\gamma$ ). Teniendo las maquinas auxiliares antes definidas podemos combinarlas para obtener las maquinas simuladoras de instrucciones. Por ejemplo  $M_i^a$  puede ser la máquina descrita en la Figura 4. En la Figura 2 tenemos una posible forma de diseñar la máquina  $IF_i^a$ . En la Figura 7 tenemos una posible forma de diseñar la máquina  $M_{i \leftarrow j}^*$  para el caso  $\Sigma = \{a, b\}$  y  $i < j$ .

Spongamos ahora que  $\mathcal{P} = I_1 \dots I_n$ . Para cada  $i = 1, \dots, n$ , definiremos una máquina  $M_i$  que simulara la instrucción  $I_i$ . Luego uniremos adecuadamente estas maquinas para formar la máquina que simulara a  $\mathcal{P}$

- Si  $Bas(I_i) = N\bar{j} \leftarrow N\bar{j} + 1$  tomaremos  $M_i = M_j^+$
- Si  $Bas(I_i) = N\bar{j} \leftarrow N\bar{j} - 1$  tomaremos  $M_i = M_j^-$
- Si  $Bas(I_i) = N\bar{j} \leftarrow 0$  tomaremos  $M_i = M_{j \leftarrow 0}$ .
- Si  $Bas(I_i) = N\bar{j} \leftarrow N\bar{m}$  tomaremos  $M_i = M_{j \leftarrow m}^\#$ .
- Si  $Bas(I_i) = IF\ N\bar{j} \neq 0\ GOTO\ L\bar{m}$  tomaremos  $M_i = IF_j$ .
- Si  $Bas(I_i) = P\bar{j} \leftarrow P\bar{j}.a$  tomaremos  $M_i = M_j^a$ .
- Si  $Bas(I_i) = P\bar{j} \leftarrow \neg P\bar{j}$  tomaremos  $M_i = M_j^\neg$ .
- $Bas(I_i) = P\bar{j} \leftarrow \varepsilon$  tomaremos  $M_i = M_{j \leftarrow \varepsilon}$ .
- $Bas(I_i) = P\bar{j} \leftarrow P\bar{m}$  tomaremos  $M_i = M_{j \leftarrow m}^*$ .
- $Bas(I_i) = IF\ P\bar{j}\ BEGINS\ a\ GOTO\ L\bar{m}$  tomaremos  $M_i = IF_j^a$ .
- $Bas(I_i) = SKIP$  tomaremos  $M_i = M_{SKIP}$ .
- $Bas(I_i) = GOTO\ L\bar{m}$  tomaremos  $M_i = M_{GOTO}$ .

Ya que la máquina  $M_i$  puede tener uno o dos estados finales, la representaremos como se muestra en la Figura 5, entendiendo que en el caso en que  $M_i$  tiene un solo estado final, este esta representado por el circulo de abajo a la izquierda y en el caso en que  $M_i$  tiene dos estados finales, el estado final representado con líneas punteadas corresponde al estado  $q_{si}$  y el otro al estado  $q_{no}$ .

Para armar la máquina que simulara a  $\mathcal{P}$  hacemos lo siguiente. Primero unimos las maquinas  $M_1, \dots, M_n$  como lo muestra la Figura 6. Luego para cada  $i$  tal que  $Bas(I_i)$  es de la forma  $\alpha GOTO\ L\bar{m}$ , ligamos con una flecha de la forma

$$\xrightarrow{B, B, K}$$

el estado final  $q_{si}$  de la  $M_i$  con el estado inicial de la  $M_h$ , donde  $h$  es tal que  $I_h$  es la primer instrucción que tiene label  $L\bar{m}$ . Es intuitivamente claro que la máquina así obtenida cumple con lo requerido aunque una Proof formal de esto puede resultar extremadamente tediosa.  $\square$

**Theorem 57.** Si  $f : D_f \subseteq \omega^n \times \Sigma^{*m} \rightarrow O$  es  $\Sigma$ -recursiva, entonces  $f$  es  $\Sigma$ -Turing computable.

*Proof.* Supongamos  $O = \Sigma^*$ . Ya que  $f$  es  $\Sigma$ -computable, existe  $\mathcal{P} \in \text{Pro}^\Sigma$  el cual computa  $f$ . Note que podemos suponer que  $\mathcal{P}$  tiene la propiedad de que cuando  $\mathcal{P}$  termina, en el estado alcanzado las variables numéricas tienen todas el valor 0 y las alfabéticas distintas de P1 todas el valor  $\varepsilon$ . Sea  $M$  la máquina de Turing con unit dada por el lema anterior, donde elegimos el numero  $k$  con la propiedad adicional de ser mayor que  $n$  y  $m$ . Sea  $M_1$  una máquina tal que para cada  $(\vec{x}, \vec{\alpha}) \in \omega^n \times \Sigma^{*m}$ ,

$$\left[ q_0 B \mid^{x_1} B \dots B \mid^{x_n} B \alpha_1 B \dots B \alpha_n B \right] \vdash^* \left[ q B \mid^{x_1} B \dots B \mid^{x_n} B^{k-n} B \alpha_1 B \dots B \alpha_m B \right]$$


donde  $q_0$  es el estado inicial de  $M_1$  y  $q$  es un estado tal que  $\delta(q, \sigma) = \emptyset$ , para cada  $\sigma$ . Sea  $M_2$  una máquina tal que para cada  $\alpha \in \Sigma^*$ ,

$$\left[ q_0 B^{k+1} \alpha \right] \vdash^* \left[ q B \alpha \right]$$

donde  $q_0$  es el estado inicial de  $M_2$  y  $q$  es un estado tal que  $\delta(q, \sigma) = \emptyset$ , para cada  $\sigma$ . Note que la concatenación de  $M_1$ ,  $M$  y  $M_2$  (en ese orden) produce una máquina de Turing la cual computa  $f$ .  $\square$

## References

- [1] DIEGO VAGGIONE, «Apunte de Clase, 2017», *FaMAF, UNC*.
- [2] AGUSTÍN CURTO, «Carpeta de Clase, 2017», *FaMAF, UNC*.

Por favor, mejorá este documento en github   
<https://github.com/acurto714/resumenLengForm>