# Guía 5: Paradigma de Godel

## Idea

- La idea es partir de un conjunto inicial de funciones muy simples y obviamente $\Sigma$-efectivamente computables, y luego obtener nuevas funciones $\Sigma$-efectivamente computables usando constructores que preservan la computabilidad efectiva
- Las funciones $\Sigma$-recursivas son las que se obtienen iterando el uso de estos constructores, partiendo del conjunto inicial
  $\{Suc, Pred, C_0^{0,0}, C_\varepsilon^{0,0}\} \cup \{d_a : a \in \Sigma\} \cup \{p_j^{n,m} : 1 \leq j \leq n+m\}$
- Los constructores que se usarán (y conservan computabilidad efectiva) serán:
  - *Composición*
  - *Recursión primitiva*
  - *Minimización de predicados totales*
- Una función es $\Sigma$-recursiva primitiva si se obtiene a partir del conjunto inicial usando solo composición y recursión primitiva

## Composición

- *Definición*:
  - Dadas funciones $\Sigma$-mixtas $f, f_1, .., f_r$ con $r \geq 1$, diremos que $f \circ [f_1, .., f_r]$ es obtenida por composición a partir de las funciones $f, f_1, .., f_r$
- *Lemas*:
  - *$\Sigma$-mixta*: Sean $f, f_1, .., f_r$ funciones $\Sigma$-mixtas con $r \geq 1$ y $f \circ [f_1, .., f_r] \neq \emptyset$, entonces $\exists n, m, k, l \in \omega, s \in \{\#, *\}$ tales que:
    - $r = n + m$
    - $f$ es de tipo $(n, m, s)$
    - $f_i$ es de tipo $(k, l, \#)$ para $1 \leq i \leq n$
    - $f_i$ es de tipo $(k, l, *)$ para $n+1 \leq i \leq r$
      Y además, $f \circ [f_1, .., f_r]$ es $\Sigma$-mixta de tipo $(k, l, s)$ con:
    - $D_{f \circ [f_1, .., f_r]} = \{(\vec{x}, \vec{a}) \in \cap_{i=1}^r D_{f_i} : (f_1(\vec{x}, \vec{a}), .., f_r(\vec{x}, \vec{a})) \in D_f\}$
    - $f \circ [f_1, .., f_r](\vec{x}, \vec{a}) = f(f_1(\vec{x}, \vec{a}), .., f_r(\vec{x}, \vec{a}))$
  - *Conservación de computabilidad efectiva*: Si $f, f_1, .., f_r$ son $\Sigma$-efectivamente computables, entonces $f \circ [f_1, .., f_r]$ también lo es

## Recursión primitiva

### Conjuntos rectangulares

- *Definición*:

    - Sea $\Sigma$ un alfabeto finito, un conjunto $\Sigma$-mixto $S$ es llamado *rectangular* si es de la forma $S_1 \times \ldots \times S_n \times L_1 \times \ldots \times L_m$ con $S_i \subseteq \omega$ y $L_j \subseteq \Sigma^*$.
- *Lemas*:
    - Sea $S \subseteq \omega \times \Sigma^*$, entonces $S$ es *rectangular* $\Leftrightarrow$ si $(x, \alpha), (y, \beta) \in S$, entonces $(x, \beta) \in S$

# Variable numérica - Valores numéricos

- *Definición*: sean $f, g$ funciones dadas por:

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

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

  con $S_i \subseteq \omega$ y $L_i \subseteq \Sigma^*$ conjuntos no vacíos, entonces existe una única función $R : \omega \times S_1 \times .. \times S_n \times L_1 \times \ldots \times L_m \to \omega$ tal que:

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

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

    - Llamaremos $R(f, g)$ a esta función y diremos que $R(f, g)$ es obtenida por recursión primitiva a partir de $f$ y $g$
- *Lemas*:
    - *Conservación de computabilidad efectiva*: Si $f, g$ son $\Sigma$-efectivamente computables, entonces $R(f, g)$ también lo es

# Variable numérica - Valores alfabéticos

- *Definición*: sean $f, g$ funciones dadas por:

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

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

  con $S_i \subseteq \omega$ y $L_i \subseteq \Sigma^*$ conjuntos no vacíos, entonces existe una única función $R(f, g) : \omega \times S_1 \times .. \times S_n \times L_1 \times \ldots \times L_m \to \Sigma^*$ tal que:

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

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

    - Llamaremos $R(f, g)$ a esta función y diremos que $R(f, g)$ es obtenida por recursión primitiva a partir de $f$ y $g$
- *Lemas*:
    - *Conservación de computabilidad efectiva*: Si $f, g$ son $\Sigma$-efectivamente computables, entonces $R(f, g)$ también lo es

# Variable alfabética - Valores numéricos

- *Definiciones*:
  - *Familia $\Sigma$-indexada de funciones*: Dado un alfabeto $\Sigma$, una familia $\Sigma$-indexada de funciones es una función $\mathcal{G}$ tal que $D_{\mathcal{G}} = \Sigma$ y $\forall a \in D_{\mathcal{G}}, \mathcal{G}(a)$ es una función
    - Si $\mathcal{G}$ es una familia $\Sigma$-indexada de funciones, entonces para $a \in \Sigma$ escribiremos $\mathcal{G}_a$ en lugar de $\mathcal{G}(a)$
  - *Recursión primitiva*: Sea $\Sigma$ un alfabeto finito, y sean $f$ una función y $\mathcal{G}$ una familia $\Sigma$-indexada de funciones tales que:

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

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

    para cada $a \in \Sigma$, y con $S_i \subseteq \omega$ y $L_i \subseteq \Sigma^*$ conjuntos no vacíos, entonces definimos

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

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

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

    - Diremos que $R(f, \mathcal{G})$ es obtenida por recursión primitiva a partir de $f$ y $\mathcal{G}$
- *Lemas*:
  - *Conservación de computabilidad efectiva*: Si $f$ y cada $\mathcal{G}_a$ son $\Sigma$-efectivamente computables, entonces $R(f, \mathcal{G})$ también lo es

# Variable alfabética - Valores alfabéticos

- *Definición*:
  - Sea $\Sigma$ un alfabeto finito, y sean $f$ una función y $\mathcal{G}$ una familia $\Sigma$-indexada de funciones tales que:

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

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

    para cada $a \in \Sigma$, y con $S_i \subseteq \omega$ y $L_i \subseteq \Sigma^*$ conjuntos no vacíos, entonces definimos

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

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

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

    - Diremos que $R(f, \mathcal{G})$ es obtenida por recursión primitiva a partir de $f$ y $\mathcal{G}$
- *Lemas*:

- *Conservación de computabilidad efectiva*: Si $f$ y cada $\mathcal{G}_a$ son $\Sigma$-efectivamente computables, entonces $R(f, \mathcal{G})$ también lo es

# Funciones $\Sigma$-recursivas primitivas

- *Definición*: Dados los conjuntos $PR_0^\Sigma \subseteq PR_1^\Sigma \subseteq \ldots \subseteq PR^\Sigma$ definidos como

$$PR_0^\Sigma = \{Suc, Pred, C_0^{0,0}, C_\varepsilon^{0,0}\} \cup \{d_a : a \in \Sigma\} \cup \{p_j^{n,m} : 1 \leq j \leq n+m\}$$

$$\begin{aligned} PR_{k+1}^\Sigma =& PR_k^\Sigma \cup \{f \circ [f_1, .., f_r] : f, f_1, .., f_r \in PR_k^\Sigma, r \geq 1\} \\ & \cup \{R(f, \mathcal{G}) : f, \mathcal{G}_a \in PR_k^\Sigma \forall a \in \Sigma\} \cup \{R(f, g) : f, g \in PR_k^\Sigma\} \end{aligned}$$

$$PR^\Sigma = \bigcup_{k \in \omega} PR_k^\Sigma$$

  Diremos que una función es llamada $\Sigma$-recursiva primitiva ($\Sigma$-*p.r.*) si pertenece a $PR^\Sigma$
- *Proposiciones*:
  - *Computabilidad efectiva*: Si $f \in PR^\Sigma$, entonces $f$ es $\Sigma$-efectivamente computable
- *Lemas*:
  - *Ejemplos de funciones $\Sigma$-p.r.*:
    - *Operaciones numéricas*:
      - $\lambda xy[x + y] \in PR^\Sigma$
      - $\lambda xy[x \cdot y] \in PR^\Sigma$
      - $\lambda x[x!] \in PR^\Sigma$
      - $\lambda xy[x^y] \in PR^\Sigma$
      - $\lambda xy[x \dot{-} y] \in PR^\Sigma$
      - $\lambda xy[max(x, y)] \in PR^\Sigma$
      - $\lambda xy[x = y] \in PR^\Sigma$
      - $\lambda xy[x \leq y] \in PR^\Sigma$
    - *Operaciones de palabras*
      - $\lambda \alpha\beta[\alpha\beta] \in PR^\Sigma$
      - $\lambda \alpha[|\alpha|] \in PR^\Sigma$
      - $\lambda t\alpha[\alpha^t] \in PR^\Sigma$
      - $\lambda \alpha\beta[\alpha = \beta] \in PR^\Sigma$
    - *Otras*:
      - $\emptyset \in PR^\Sigma$
      - $\forall n, m, k \in \omega, \alpha \in \Sigma^*; C_k^{n,m}, C_\alpha^{n,m} \in PR^\Sigma$
      - Si $\leq$ es un orden total sobre $\Sigma$, entonces $s^\leq, \#^\leq, *^\leq \in PR^\Sigma$
  - *Operaciones lógicas entre predicados*: Si $P, Q$ son predicados $\Sigma$-p.r. con igual *dominio*, entonces $P \wedge Q, P \vee Q, \neg P$ son $\Sigma$-p.r. también.

# Conjuntos $\Sigma$-recursivos primitivos

- *Definición*:
  - Un conjunto $\Sigma$-mixto $S \subseteq \omega^n \times \Sigma^{*m}$ es llamado $\Sigma$-recursivo primitivo si su función característica $\chi_S^{\omega^n \times \Sigma^{*m}}$ es $\Sigma$-p.r.
    - *Notar que* $\chi_S^{\omega^n \times \Sigma^{*m}} = \lambda \vec{x} \vec{\alpha} [(\vec{x}, \vec{\alpha}) \in S]$
- *Lemas*:
  - Si $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ son $\Sigma$-p.r., entonces $S_1 \cup S_2, S_1 \cap S_2, S_1 - S_2$ son $\Sigma$-p.r. también
  - Si $S \subseteq \omega^n \times \Sigma^{*m}$ es *finito*, entonces $S$ es $\Sigma$-p.r.
  - *Conjunto rectangular $\Sigma$-p.r.:* Sean $S_1, .., S_n \subseteq \omega, L_1, .., L_m \subseteq \Sigma^*$ conjuntos no vacíos, entonces $S_1 \times .. \times S_n \times L_1 \times .. \times L_m$ es $\Sigma$-p.r. $\Leftrightarrow S_1, .., S_n, L_1, .., L_m$ son $\Sigma$-p.r.
  - *$\Sigma$-p.r. para función restringida*: Sea $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ que es $\Sigma$-p.r. (con $O \in \{\omega, \Sigma^*\}$), si $S \subseteq D_f$ es $\Sigma$-p.r., entonces $f|_S$ es $\Sigma$-p.r. también
    - *Definición*: Dada una función $f$ y un conjunto $S \subseteq D_f$, usaremos $f|_S$ para denotar la restricción de $f$ al conjunto $S$, es decir, $f|_S = f \cap (S \times I_f)$. Notar que $D_{f|_S} = S$ y $f|_S(e) = f(e) \forall e \in S$.
  - Sean $O \in \{\omega, \Sigma^*\}$ y $n, m \in \omega$, si $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ es $\Sigma$-p.r., entonces existe una función $\Sigma$-p.r. $\bar{f} : \omega^n \times \Sigma^{*m} \to O$ tal que $f = \bar{f}|_{D_f}$
- *Proposiciones*:
  - *Relación Conjunto/Función $\Sigma$-p.r.*: Un conjunto $S$ es $\Sigma$-p.r. si y solo si $S$ es el dominio de alguna función $\Sigma$-p.r.

## Lema de división por casos para funciones $\Sigma$-p.r.

- *Observación*: Si $f_i : D_f \to O \forall i = 1, .., k$ son funciones tales que $\forall i \neq j, D_{f_i} \cap D_{f_j} = \emptyset$, entonces $f_1 \cup .. \cup f_k$ es la función dada por

$$D_{f_1} \cup .. \cup D_{f_k} \to O$$

$$e \to \begin{cases} f_1(e) & \text{si } e \in D_{f_1} \\ \vdots & \vdots \\ f_k(e) & \text{si } e \in D_{f_k} \end{cases}$$

- *Lema*: Sean $O \in \{\omega, \Sigma^{*m}\}$ y $n, m \in \omega$, y supongamos que $f_i : D_{f_i} \subseteq \omega^n \times \Sigma^{*m} \to O \forall i = 1, .., k$ son $\Sigma$-p.r. tales que $\forall i \neq j, D_{f_i} \cap D_{f_j} = \emptyset$. Entonces $f_1 \cup .. \cup f_k$ es $\Sigma$-p.r.
- *Consejo*: Si se quiere usar este lema para probar que una función $f$ es $\Sigma$-p.r., entonces primero hay que definir las funciones $f_1, .., f_k$ tales que $\forall i \neq j, D_{f_i} \cap D_{f_j} = \emptyset$ y $f = f_1 \cup .. \cup f_k$ y luego comenzar a probar y ver si algo es o no $\Sigma$-p.r.
  - Determinar el $k$, es decir, la cantidad de *casos* en la descripción de $f$.
  - Para cada *caso* de la descripción de $f$, asociar un subconjunto del dominio de $f$ el cual sea justamente definido por la propiedad correspondiente (tener en cuenta que debe ser *subconjunto* y una descripción puede no usar todas las variables)

    - Notar que los subconjuntos $S_1, .., S_k$ definidos deben ser disjuntos de a pares y, unidos, deben dar el dominio de $f$
    - Para cada $i$ defina $f_i$ de la siguiente manera:
        - Dominio de $f_i$ es $S_i$
        - Regla de $f_i$ dada por la regla de describe $f$ para el caso $i$-ésimo
    - En general, suele suceder que $f_i$ es la restricción a $S_i$ de una función con dominio más amplio. Por ello, a veces se prueba entonces que tanto dicha función como $S_i$ son $\Sigma$-p.r., resultando así, por lema, que $f_i$ es $\Sigma$-p.r.
- *Ejemplo*: Con esto, se puede probar que $\lambda i \alpha[[\alpha]_i]$ es $\Sigma$-p.r.