

# Guía 7: El paradigma imperativo de Neumann: El lenguaje $\mathcal{S}^\Sigma$

## Sintaxis de $\mathcal{S}^\Sigma$

- Definiciones:

- Definimos  $Sig : Num^* \rightarrow Num^*$  como:

$$\begin{aligned} Sig(\varepsilon) &= 1 \\ Sig(\alpha 0) &= \alpha 1 \\ Sig(\alpha 1) &= \alpha 2 \\ Sig(\alpha 2) &= \alpha 3 \\ Sig(\alpha 3) &= \alpha 4 \\ Sig(\alpha 4) &= \alpha 5 \\ Sig(\alpha 5) &= \alpha 6 \\ Sig(\alpha 6) &= \alpha 7 \\ Sig(\alpha 7) &= \alpha 8 \\ Sig(\alpha 8) &= \alpha 9 \\ Sig(\alpha 9) &= Sig(\alpha) 0 \end{aligned}$$

- Definimos  $Dec : \omega \rightarrow Num^*$  como:

$$\begin{aligned} Dec(0) &= \varepsilon \\ Dec(n+1) &= Sig(Dec(n)) \end{aligned}$$

- Notar que para  $n \in N$ ,  $Dec(n)$  es la notación usual decimal de  $n$
- Para hacer más ágil la notación, escribiremos  $\bar{n}$  en lugar de  $Dec(n)$ , por lo que  $Dec = \lambda n[\bar{n}]$
- Sintaxis:** La sintaxis de  $\mathcal{S}^\Sigma$  será dada utilizando el alfabeto  $\Sigma \cup \Sigma_P$  donde

$$\Sigma_P = Num \cap \{\leftarrow, +, \dot{-}, \cdot, \neq, \curvearrowright, \varepsilon, N, K, P, L, I, F, G, O, T, B, E, S\}$$

- Las palabras de la forma:
  - $N\bar{k}$  con  $k \in N$  son llamadas *variables numéricas de  $\mathcal{S}^\Sigma$*
  - $P\bar{k}$  con  $k \in N$  son llamadas *variables alfabéticas de  $\mathcal{S}^\Sigma$*
  - $L\bar{k}$  con  $k \in N$  son llamadas *labels de  $\mathcal{S}^\Sigma$*
- Una *instrucción básica de  $\mathcal{S}^\Sigma$*  es una palabra  $(\Sigma \cup \Sigma_P)^*$  la cual es de alguna de las siguientes formas (donde  $a \in \Sigma$ ;  $k, n \in N$ ):
  - $N\bar{k} \leftarrow N\bar{k} + 1$
  - $N\bar{k} \leftarrow N\bar{k} \dot{-} 1$ 
    - Si el contenido de  $N\bar{k}$  es 0, dejarlo sin modificar. Caso contrario, disminuir en 1 el contenido de  $N\bar{k}$
  - $N\bar{k} \leftarrow N\bar{n}$ 
    - Copiar el contenido de  $N\bar{n}$  en  $N\bar{k}$  sin modificar el contenido de  $N\bar{n}$

- $N\bar{k} \leftarrow 0$
- $P\bar{k} \leftarrow P\bar{k}.a$ 
  - Modificar el contenido de  $P\bar{k}$  agregando el símbolo  $a$  a la derecha
- $P\bar{k} \leftarrow \neg P\bar{k}$ 
  - Si el contenido de  $P\bar{k}$  es  $\varepsilon$ , dejarlo sin modificar. Caso contrario, eliminar el primer símbolo del contenido de  $P\bar{k}$
- $P\bar{k} \leftarrow P\bar{n}$
- $P\bar{k} \leftarrow \varepsilon$
- IF  $N\bar{k} \neq 0$  GOTO  $L\bar{n}$
- IF  $P\bar{k}$  BEGINS  $a$  GOTO  $L\bar{n}$ 
  - Si el contenido de  $P\bar{k}$  comienza con el símbolo  $a$ , entonces ir a la instrucción con label  $L\bar{n}$ . Caso contrario, continuar con la siguiente instrucción
- GOTO  $L\bar{n}$
- SKIP
- Una *instrucción* de  $\mathcal{S}^\Sigma$  es ya sea una instrucción básica de  $\mathcal{S}^\Sigma$ , o una palabra de la forma  $\alpha I$ , donde  $\alpha \in \{L\bar{n} : n \in N\}$  e  $I$  es una instrucción básica de  $\mathcal{S}^\Sigma$ .
  - Usaremos  $Ins^\Sigma$  para denotar el conjunto de todas las instrucciones de  $\mathcal{S}^\Sigma$
  - Cuando  $I$  es de la forma  $L\bar{n}J$  con  $J$  una instrucción básica, diremos que  $L\bar{n}$  es la *label* de  $I$
- Un *programa* de  $\mathcal{S}^\Sigma$  es una palabra de la forma  $I_1 I_2 \dots I_n$  donde  $n \geq 1, I_1, \dots, I_n \in Ins^\Sigma$  y además se cumple la *ley de los GOTO*:  $\forall i \in \{1, \dots, n\}$ , si  $GOTO L\bar{m}$  es un tramo final de  $I_i$ , entonces  $\exists j \in \{1, \dots, n\}$  tal que  $I_j$  tiene label  $L\bar{m}$ 
  - Usaremos  $Pro^\Sigma$  para denotar el conjunto de todos los programas de  $\mathcal{S}^\Sigma$
  - Definimos  $n(\mathcal{P})$  como la cantidad de instrucciones de  $\mathcal{P}$ , e  $I_i^\mathcal{P}$  como la  $i$ -ésima instrucción de  $\mathcal{P}$  para  $i \in \{1, \dots, n(\mathcal{P})\}$ . Además,  $I_i^\mathcal{P} = \varepsilon$  cuando  $i = 0$  o  $i > n(\mathcal{P})$ 
    - Notamos con  $\lambda \mathcal{P}[n(\mathcal{P})]$  y  $\lambda i \mathcal{P}[I_i^\mathcal{P}]$
- *Lemas*:
  - *Parseo de programas*: Sea  $\Sigma$  un alfabeto finito, se tiene que:
    - Si  $I_1 \dots I_n = J_1 \dots J_m$  con  $I_1, \dots, I_n, J_1, \dots, J_m \in Ins^\Sigma$ , entonces  $n = m$  y  $I_i = J_i \forall i \geq 1$
    - Si  $\mathcal{P} \in Pro^\Sigma$ , entonces existe una *única* sucesión de instrucciones  $I_1, \dots, I_n$  tal que  $\mathcal{P} = I_1 \dots I_n$   
Luego, esto significa que, dado un programa  $\mathcal{P}$ , tenemos unívocamente determinados  $n(\mathcal{P})$  e  $I_1, \dots, I_{n(\mathcal{P})}$  tales que  $\mathcal{P} = I_1 \dots I_{n(\mathcal{P})}$

## Semántica de $\mathcal{S}^\Sigma$

- *Definiciones*:

- **Bas:** Definimos  $Bas : Ins^\Sigma \rightarrow (\Sigma \cup \Sigma_P)^*$  dada por

$$Bas(I) = \begin{cases} J & \text{si } I \text{ es de la forma } L\bar{k}J \text{ con } J \in Ins^\Sigma \\ I & \text{en otro caso} \end{cases}$$

- Asumiremos siempre que en una computación vía un programa de  $S^\Sigma$ , todas excepto una cantidad finita de las variables numéricas tienen el valor 0 y todas excepto una cantidad finita de las variables alfabéticas tienen el valor  $\varepsilon$
- **Estado:** Un estado es un par  $(\vec{s}, \vec{\sigma}) = ((s_1, s_2, \dots), (\sigma_1, \sigma_2, \dots)) \in \omega^{[N]} \times \Sigma^{*[N]}$  y, si  $i \geq 1$ , entonces diremos que  $s_i$  es el contenido o valor de la variable  $N\bar{i}$  en el estado  $(\vec{s}, \vec{\sigma})$  y  $\sigma_i$  es el contenido o valor de la variable  $P\bar{i}$  en el estado  $(\vec{s}, \vec{\sigma})$
- **Descripción instantánea:** Una descripción instantánea es una terna  $(i, \vec{s}, \vec{\sigma})$  tal que  $(\vec{s}, \vec{\sigma})$  es un estado e  $i \in \omega$ .
  - Es decir,  $\omega \times \omega^{[N]} \times \Sigma^{*[N]}$  es el conjunto de todas las descripciones instantáneas
  - Intuitivamente,  $(i, \vec{s}, \vec{\sigma})$  nos dice que las variables están en el estado  $(\vec{s}, \vec{\sigma})$  y que la instrucción que debemos realizar es  $I_i^P$
- **Sucesora:** Dado un programa  $\mathcal{P}$ , definimos  $S_{\mathcal{P}} : \omega \times \omega^{[N]} \times \Sigma^{*[N]} \rightarrow \omega \times \omega^{[N]} \times \Sigma^{*[N]}$  como la función que asignará a una descripción instantánea  $(i, \vec{s}, \vec{\sigma})$  la descripción instantánea sucesora de  $(i, \vec{s}, \vec{\sigma})$  con respecto a  $\mathcal{P}$ . Es decir, hay varios casos posibles:
  - Si  $i \notin \{1, \dots, n(\mathcal{P})\}$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i, \vec{s}, \vec{\sigma})$
  - Si  $Bas(I_i^P) = N\bar{k} \leftarrow N\bar{k} - 1$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, (s_1, \dots, s_{k-1}, s_k - 1, s_{k+1}, \dots), \vec{\sigma})$
  - Si  $Bas(I_i^P) = N\bar{k} \leftarrow N\bar{k} + 1$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, (s_1, \dots, s_{k-1}, s_k + 1, s_{k+1}, \dots), \vec{\sigma})$
  - Si  $Bas(I_i^P) = N\bar{k} \leftarrow N\bar{n}$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, (s_1, \dots, s_{k-1}, s_n, s_{k+1}, \dots), \vec{\sigma})$
  - Si  $Bas(I_i^P) = N\bar{k} \leftarrow 0$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, (s_1, \dots, s_{k-1}, 0, s_{k+1}, \dots), \vec{\sigma})$
  - Si  $Bas(I_i^P) = \text{IF } N\bar{k} \neq 0 \text{ GOTO } L\bar{m}$ , entonces hay dos posibilidades:
    - Si el valor de  $N\bar{k}$  es 0, entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, \vec{\sigma})$
    - Si el valor de  $N\bar{k}$  es no nulo, entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (\min\{l : I_l^P \text{ tiene label } L\bar{m}\}, \vec{s}, \vec{\sigma})$
  - Si  $Bas(I_i^P) = P\bar{k} \leftarrow \neg P\bar{k}$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, (\sigma_1, \dots, \sigma_{k-1}, \neg \sigma_k, \sigma_{k+1}, \dots))$
  - Si  $Bas(I_i^P) = P\bar{k} \leftarrow P\bar{k}.a$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, (\sigma_1, \dots, \sigma_{k-1}, \sigma_k a, \sigma_{k+1}, \dots))$
  - Si  $Bas(I_i^P) = P\bar{k} \leftarrow P\bar{n}$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, (\sigma_1, \dots, \sigma_{k-1}, \sigma_n, \sigma_{k+1}, \dots))$
  - Si  $Bas(I_i^P) = P\bar{k} \leftarrow \varepsilon$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, (\sigma_1, \dots, \sigma_{k-1}, \varepsilon, \sigma_{k+1}, \dots))$

- Si  $Bas(I_i^{\mathcal{P}}) = \text{IF } P\bar{k} \text{ BEGINS } a \text{ GOTO } L\bar{m}$ , entonces hay dos posibilidades:
  - Si el valor de  $P\bar{k}$  comienza con  $a$ , entonces  
 $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (\min\{l : I_l^{\mathcal{P}} \text{ tiene label } L\bar{m}\}, \vec{s}, \vec{\sigma})$
  - Si el valor de  $P\bar{k}$  no comienza con  $a$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, \vec{\sigma})$
- Si  $Bas(I_i^{\mathcal{P}}) = \text{GOTO } L\bar{m}$ , entonces  
 $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (\min\{l : I_l^{\mathcal{P}} \text{ tiene label } L\bar{m}\}, \vec{s}, \vec{\sigma})$
- Si  $Bas(I_i^{\mathcal{P}}) = \text{SKIP}$ , entonces  $S_{\mathcal{P}}(i, \vec{s}, \vec{\sigma}) = (i + 1, \vec{s}, \vec{\sigma})$
- *Computación partiendo de un estado:* Dado un programa  $\mathcal{P}$  y un estado  $(\vec{s}, \vec{\sigma})$ , a la infinitupla

$$((1, \vec{s}, \vec{\sigma}), S_{\mathcal{P}}(1, \vec{s}, \vec{\sigma}), S_{\mathcal{P}}(S_{\mathcal{P}}(1, \vec{s}, \vec{\sigma})), \dots)$$

la llamaremos la *computación de  $\mathcal{P}$  partiendo del estado  $(\vec{s}, \vec{\sigma})$* .

- Diremos que  $S_{\mathcal{P}}(S_{\mathcal{P}}(\dots(S_{\mathcal{P}}(1, \vec{s}, \vec{\sigma})).\dots)) = (j, \vec{u}, \vec{\eta})$  con  $S_{\mathcal{P}}$  aplicado  $t$  veces, es la *descripción instantánea obtenida luego de  $t$  pasos partiendo del estado  $(\vec{s}, \vec{\sigma})$* , y  $(\vec{u}, \vec{\eta})$  es el estado obtenido luego de  $t$  pasos partiendo del estado  $(\vec{s}, \vec{\sigma})$
- *Detención:*
  - Cuando la primer coordenada de  $S_{\mathcal{P}}(S_{\mathcal{P}}(\dots(S_{\mathcal{P}}(1, \vec{s}, \vec{\sigma})).\dots))$  (con  $S_{\mathcal{P}}$  aplicado  $t$  veces) es  $n(\mathcal{P}) + 1$ , diremos que  $\mathcal{P}$  se detiene (luego de  $t$  pasos), partiendo desde el estado  $(\vec{s}, \vec{\sigma})$
  - Caso contrario, si ninguna de las primeras coordenadas en la infinitupla de la computación es igual a  $n(\mathcal{P}) + 1$ , diremos que  $\mathcal{P}$  no se detiene partiendo desde el estado  $(\vec{s}, \vec{\sigma})$