Cálculo de PRIMERO

El conjunto PRIMERO(α), siendo α una forma sentecial, es el conjunto de Terminales que pueden aparecer los primeros en cadenas derivadas de α.

El conjunto PRIMERO puede referirse a una forma sentencial (PRIMERO(α), siendo α por ejemplo aBD), pero también a un símbolo No Terminal de la gramática considerada (PRIMERO(A)). Si la gramática está formada por reglas $A \rightarrow \alpha_i$ el conjunto PRIMERO(A) será la unión de todos los PRIMERO(α_i), que deben ser calculados por separado.

Para calcular PRIMERO(α) basta con seguir las siguientes reglas para cada regla de producción de la gramática:

1. Si $\alpha \equiv \in$, donde $\in$ es la cadena vacía :

   Añadir { $\in$ } a PRIMERO(α)

2. Si $\alpha \equiv$ a, donde a es un Terminal:

   Añadir { a } a PRIMERO(α)

3. Si $\alpha \equiv$ A, donde A es un No Terminal:

   Añadir PRIMERO(A) a PRIMERO(α).

4. Si $\alpha \equiv$ AB, donde A y B son No Terminales y PRIMERO(A) incluye { $\in$ }:

   Añadir PRIMERO(A) a PRIMERO(α), pero sin incluir de momento { $\in$ }. Además, y puesto que A puede ser vacío, añadir PRIMERO(B) a PRIMERO(α).

   Esta regla puede extenderse a todos los No Terminales que aparezcan detrás de A. Por ejemplo, si $\alpha \equiv$ ABCD y PRIMERO (A), PRIMERO(B) y PRIMERO(C) incluyen { $\in$ }, entonces habrá que añadir a PRIMERO(α) la unión de PRIMERO(A), PRIMERO(B), PRIMERO(C) y PRIMERO(D), sin incluir { $\in$ }.

   Por último se decide si incluir { $\in$ } en PRIMERO(α): sólo debe incluirse si aparece en los conjuntos PRIMERO de todos los No Terminales. Por ejemplo en el caso anterior, si PRIMERO(D) también incluyese { $\in$ }, entonces PRIMERO(α) incluiría { $\in$ }.

Primer ejemplo detallado de cálculo de PRIMERO

Veamos un ejemplo de cálculo detallado de PRIMERO para la siguiente gramática, que describe expresiones aritméticas con sumas y multiplicaciones:

E → T E’

E’ → + T E’ | ∈

T → F T’

T’ → * F T’ | ∈

F → (E) | ident

- PRIMERO(E) = PRIMERO(TE’) (en adelante escribiremos simplemente PRIMERO(E)).

  PRIMERO(E) = PRIMERO(T) por la Regla 3, así que pasamos a calcular PRIMERO(T).
- PRIMERO(T) = PRIMERO(F) por la Regla 3, así que pasamos a calcular PRIMERO(F).
- PRIMERO(F) = { (, ident } por la Regla 2.
- En este punto ya sabemos que PRIMERO(E) = PRIMERO(T) = PRIMERO(F)= { (, ident }
- PRIMERO(E’) = { + , ∈ } por las Reglas 2 y 1.
- PRIMERO(T’) = { * , ∈ } por las Reglas 2 y 1.

Resultado:

PRIMERO(E) = { (, ident }

PRIMERO(E’) = { + , ∈ }

PRIMERO(T) = { (, ident }

PRIMERO(T’) = { * , ∈ }

PRIMERO(F) = { (, ident }

Segundo ejemplo detallado de cálculo de PRIMERO

Veamos otro ejemplo de cálculo detallado de PRIMERO para la siguiente gramática, con el objetivo de aclarar el uso de la Regla 4:

$$A \rightarrow Aa \mid BCD$$
$$B \rightarrow b \mid \in$$
$$C \rightarrow c \mid \in$$
$$D \rightarrow d \mid Ce$$

- PRIMERO(A) = PRIMERO(Aa) ∪ PRIMERO(BCD) = PRIMERO(BCD)
- Calculamos el resto de los conjuntos PRIMERO, ya que vamos a necesitarlos
- PRIMERO(B) = { b , ∈ } por las Reglas 2 y 1.
- PRIMERO(C) = { c , ∈ } por las Reglas 2 y 1.
- PRIMERO(D) = { d } ∪ PRIMERO(Ce).

  Como PRIMERO(Ce) incluye el elemento vacío, aplicamos la Regla 4 y calculamos PRIMERO(e) = { e }, quedando PRIMERO(D) = { d, c, e }. Obsérvese que por la Regla 4, { ∈ } no se incluye: intuitivamente se comprueba que D no puede derivar la cadena vacía.
- PRIMERO(A) = PRIMERO(Aa) ∪ PRIMERO(BCD)

  PRIMERO(BCD) = PRIMERO(B) ∪ PRIMERO(C) ∪ PRIMERO(D) = { b , c , d , e }. Por la Regla 4 sabemos que { ∈ } no debe incluirse: no está incluida en PRIMERO(D), y por tanto resulta imposible deducir la cadena vacía a partir de A.

  PRIMERO(Aa) es un caso al que conviene prestar atención, aunque no deja de ser un caso más de aplicación de la Regla 4: si PRIMERO(BCD) incluyese {∈}, habría que incluir {a} en PRIMERO(A). Como no es así, no lo incluimos, quedando:

  PRIMERO(A) = PRIMERO(BCD) = { b , c , d , e }

Resultado:

PRIMERO(A) = { b , c , d , e }

PRIMERO(B) = { b , ∈ }

PRIMERO(C) = { c , ∈ }

PRIMERO(D) = { d , c , e }

Cálculo de SIGUIENTE

Si A es un símbolo No Terminal de una gramática, SIGUIENTE(A) es el conjunto de Terminales (incluyendo el símbolo de fin de cadena, $) que pueden aparecer justo después de A en alguna forma sentencial derivada del símbolo inicial.

El cálculo de conjuntos SIGUIENTE se realiza a partir de estas reglas:

1. Los conjuntos SIGUIENTE de todos los No Terminales son inicialmente vacíos, excepto el del No Terminal inicial de la gramática, en el que se incluye el símbolo {$}.

A partir de este punto, para calcular cada uno de los conjuntos deben aplicarse las dos siguientes reglas (no son excluyentes) por cada ocurrencia del No Terminal en la parte derecha de alguna regla de producción:

2. Si $A \rightarrow \alpha B \beta$:

   Añadir a SIGUIENTE(B) los elementos de PRIMERO(β), con la excepción de $\{\in\}$: este símbolo nunca se incluirá en los conjuntos SIGUIENTE.

3. Si $A \rightarrow \alpha B$, o bien $A \rightarrow \alpha B \beta$ donde PRIMERO(β) contiene $\{\in\}$:

   Añadir a SIGUIENTE(B) los elementos de SIGUIENTE(A).

Conviene comprender las reglas en lugar de intentar memorizarlas: en el caso de SIGUIENTE, se comprueba intuitivamente que los posibles Terminales que pueden aparecer siguiendo a un No Terminal B son:

- Los que pertenecen al conjunto PRIMERO del No Terminal que aparece inmediatamente después que el No Terminal B. Esto lo expresa la Regla 2.
- Si el No Terminal B aparece el último en la parte derecha de una regla de producción (o bien el No Terminal que hay a su derecha puede derivar la cadena vacía), estaremos en un caso como el siguiente:

  $S \rightarrow aAb$

  $A \rightarrow cB$

  $B \rightarrow ...$

  Intuitivamente, ¿cuál es el conjunto de Terminales que pueden seguir a B en una cadena? Como B aparece el último en la parte derecha de una regla de producción de A, habrá que buscar los Terminales que puedan seguir a A. Es decir, los siguientes de B serán, como mínimo, los siguientes de A, en este caso {b}. Esto es lo que expresa la Regla 3.

Primer ejemplo detallado de cálculo de SIGUIENTE

Veamos ejemplos de cálculo detallado de SIGUIENTE para las mismas gramáticas que utilizamos antes:

$E \rightarrow T\ E'$

$E' \rightarrow +\ T\ E' \mid \in$

$T \rightarrow F\ T'$

$T' \rightarrow *\ F\ T' \mid \in$

$F \rightarrow (\ E\ ) \mid ident$

Partimos de los conjuntos PRIMERO calculados previamente:

PRIMERO(E) = { (, ident }

PRIMERO(E') = { + , ∈ }

PRIMERO(T) = { (, ident }

PRIMERO(T') = { * , ∈ }

PRIMERO(F) = { (, ident }

- SIGUIENTE(E) = {$} por la Regla 1.
- SIGUIENTE(E) = SIGUIENTE(E) ∪ {) } = { $,) } por la Regla 2: hemos buscado E en las partes derechas de las reglas y hemos encontrado F → (E) , comprobando que PRIMERO( )) = {) }. Como E no aparece en la parte derecha de ninguna otra regla, damos por cerrado su conjunto SIGUIENTE.
- SIGUIENTE(E') = SIGUIENTE(E) = { $,) } por la Regla 3: E' aparece el último en la parte derecha de dos reglas de producción, aunque el No Terminal que las origina es el mismo: E. Por tanto, se añade SIGUIENTE(E) a SIGUIENTE(E') y, como E' no aparece más, se da por cerrado su conjunto.
- SIGUIENTE(T) = PRIMERO(E') ∪ SIGUIENTE(E') = { + , $,) } por las Reglas 2 y 3. Hemos aplicado la Regla 3 porque en E → +TE' el No Terminal T podría ser el último a la derecha, ya que E' deriva la palabra vacía.
- SIGUIENTE(T') = SIGUIENTE(T) = { + , $,) } por la Regla 3.
- SIGUIENTE(F) = PRIMERO(T') ∪ SIGUIENTE(T) ∪ SIGUIENTE(T') = { * , + , $,) } por las Reglas 2 y 3.

Resultado:

SIGUIENTE(E) = { $,) }

SIGUIENTE(E') = { $,) }

SIGUIENTE(T) = { + , $,) }

SIGUIENTE(T') = { + , $,) }

SIGUIENTE(F) = { * , + , $,) }

Segundo ejemplo detallado de cálculo de SIGUIENTE

Veamos el segundo ejemplo de cálculo detallado de SIGUIENTE:

$A \rightarrow Aa \mid BCD$

$B \rightarrow b \mid \in$

$C \rightarrow c \mid \in$

$D \rightarrow d \mid Ce$

Partimos del resultado anterior:

PRIMERO(A) = { b , c , d , e }

PRIMERO(B) = { b , ∈ }

PRIMERO(C) = { c , ∈ }

PRIMERO(D) = { d , c , e }

- SIGUIENTE(A) = { $ } por la Regla 1.
- SIGUIENTE(A) = SIGUIENTE(A) ∪ PRIMERO(a) = { $, a } por Regla 2.
- SIGUIENTE(B) = PRIMERO(CD) por la Regla 2
- PRIMERO (CD) = PRIMERO(C) ∪ PRIMERO(D) = { c , d , e } por la Regla 4 de cálculo de PRIMERO.
- SIGUIENTE(C) = PRIMERO(D) ∪ PRIMERO(e) = { d , c , e } por la Regla 2
- SIGUIENTE(D) = SIGUIENTE(A) = { $, a } por la Regla 3.

Resultado:

SIGUIENTE(A) = { $, a }

SIGUIENTE(B) = { c , d , e }

SIGUIENTE(C) = { c , d , e }

SIGUIENTE(D) = { $, a }