CI3725 - Traductores e Interpretadores
Abril-Julio 2006
Prof. Ernesto Hernández-Novich

Laboratorio Semana VI

Lenguaje reconocido por un Autómata

Sean $L_1 = L(M_1)$ y $L_2 = L(M_2)$ dos lenguajes reconocidos por los autómatas $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ y $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ respectivamente. Muestre que el lenguaje $L = L_1L_2$ es reconocido por el autómata $M = (Q_1 \cup Q_2, \Sigma, \delta, q_1, F)$ construido tal que

$$\begin{aligned}
\delta(q,a) &= \delta_1(q,a) \forall q \in Q_1 - F_1 & (1)\\
\delta(q,a) &= \delta_1(q,a) \cup \delta_2(q_2,a) \forall q \in F_1 & (2)\\
\delta(q,a) &= \delta_2(q,a) \forall q \in Q_2 & (3)\\
F &= \begin{cases} F_2 & \text{si } Q_2 \notin F_2 \\ F_1 \cup F_2 & \text{si } Q_2 \in F_2 \end{cases}
\end{aligned}$$

Si construimos un diagrama del autómata M podemos apreciar informalmente que comienza por simular M_1 y al llegar a cualquiera de los estados de F_1 es posible continuar el procesamiento dentro de M_1 si fuere necesario o bien continuar hacia M_2 como si se hubiera partido de q_2, todo esto de manera no determinística y en virtud de la Regla de Construcción 2 para δ.

Para demostrarlo formalmente, primero debemos demostrar que dada cualquier palabra en $L = L_1L_2$ ésta puede ser reconocida por M, y luego que si una palabra está en $L(M)$ entonces también está en L_1L_2.

Sean $x \in L_1$ y $y \in L_2$, en consecuencia son reconocidos por M_1y M_2 respectivamente; formalmente

$$\begin{aligned}
(q_1, x) &\vdash^*_{M_1} (p_1, \lambda), p_1 \in F_1\\
(q_2, y) &\vdash^*_{M_2} (p_2, \lambda), p_2 \in F_2
\end{aligned}$$

Consideremos entonces lo que ocurre al simular M con la entrada xy. Puesto que $x \in L_1$, por la construcción de M tendremos que

$$\exists p_1 \in F_1 : (q_1, xy) \vdash^*_M (p_1, y)$$

Tenemos que considerar dos posibilidades:

1. Si $y \neq \lambda$, entonces podemos escribir $y = az$ para algún $a \in \Sigma$, más aún, como $y \in L_2 \wedge y \neq \lambda \Rightarrow q_2 \notin F_2$. De acuerdo con la Regla de Construcción 2 para σ, M puede pasar de $p_1$hasta algún $r \in Q_2$ consumiendo a y como $y \in L_2$ consumir z con las transiciones de M resultantes de la Regla de Construcción 3. Formalmente

$$\exists r \in Q_2 \exists s \in F_2 : (q_1, az) \vdash_M (r, z) \vdash^*_M (s, \lambda)$$

y concluimos que $xy \in L(M)$.

2. Si $y = \lambda$, entonces $q_2 \in F_2$ y de acuerdo con la regla de construcción de F, tanto q_2 como p_1 estarían en F. Por lo tanto $xy \in L(M)$.

Ahora consideremos cualquier $w \in L(M)$, esto quiere decir que

$$\exists q \in F : (q_1, w) \vdash^*_M (q, \lambda)$$

y nuevamente tenemos que considerar dos posibilidades:

1. Si $q \in F_2$ de acuerdo con la Regla 2 de Construcción de δ quiere decir que $\exists r \in F_1 \wedge \exists s \in Q_2$ tal que podemos escribir $w = xay$ y

$$(q_1, xay) \vdash^*_M (r, ay) \vdash_M (s, y) \vdash^*_M (q, \lambda)$$

Como $r \in F_1 \Rightarrow r \in L(M_1) = L_1$ y $ay \in L(M_2) = L_2$ concluimos que $w \in L_1L_2$.

2. Si $q \in F_1$ de acuerdo con la Regla de Construcción de F quiere decir que $q_2 \in F_2 \Rightarrow \lambda \in L_2$. Esto es, $w = w\lambda$ con lo que $w \in L_1$y concluimos que $w \in L_1L_2$.

Lenguaje Generado por una Gramática

Para probar que un lenguaje L es generado por una gramática G, es necesario verificar dos cosas:

- Que toda cadena derivada a partir del símbolo inicial de la gramática está en el lenguaje; formalmente, es necesario demostrar que
$$\forall w : (G \Rightarrow^* w) \Longrightarrow w \in L$$
- Que toda cadena perteneciente al lenguaje puede ser derivada a partir del símbolo inicial de la gramática; formalmente, es necesario demostrar que
$$\forall w \in L \;\text{ existe}\, G \Rightarrow^* w$$

Ejemplo

Sea la gramática

$$\begin{array}{rcl} S & \rightarrow & (S)S \\ S & \rightarrow & \lambda \end{array}$$

demostrar que genera el lenguaje L de palabras con paréntesis balanceados.

Primero demostraremos que toda palabra derivada partiendo de S tiene paréntesis balanceados. Usaremos inducción sobre el número de pasos en la derivación.

Caso base.

En un sólo paso solamente puede derivarse

$$S \rightarrow \lambda$$

que tiene paréntesis balanceados (ninguno, pero están balanceados).

Hipótesis Inductiva.

Asumiremos que toda derivación de n o menos pasos produce una palabra que tiene paréntesis balanceados.

Inducción.

Consideremos una derivación de $n+1$ pasos. En virtud de las reglas de la gramática, esa derivación tiene que ser de la forma
$$S \Rightarrow (S)S \Rightarrow^* (x)S \Rightarrow^* (x)y$$
Observamos que las derivaciones $S \Rightarrow^* x$ y $S \Rightarrow^* y$ tienen n o menos pasos, por lo tanto x e y son palabras con paréntesis balanceados, y en consecuencia $(x)y$ también tiene paréntesis balanceados. Concluimos que toda palabra derivada por la gramática G está en el lenguaje L.

Finalmente mostraremos que toda palabra con paréntesis balanceados puede ser derivada a partir del símbolo S. Usaremos inducción sobre la longitud de la palabra.

Caso Base.

Sea x una palabra que tiene paréntesis balanceados. Si $|x| = 0$ quiere decir que $x = \lambda$ y usando G podemos derivar $S \Rightarrow \lambda$.

Hipótesis Inductiva.

Si w es una palabra que tiene paréntesis balanceados, entonces su longitud es necesariamente par. Asumiremos que para toda palabra w con paréntesis balanceados tal que $|w| \leq 2n$ existe $S \Rightarrow^* w$.

Inducción.

Consideremos una palabra w de longitud $2(n+1)$ con paréntesis balanceados. Necesariamente debe comenzar con un paréntesis "abriendo" y ser de la forma $(x)y$. Más aún, escojamos x tal que (x) sea el prefijo izquierdo más corto con paréntesis balanceados, así x e y tienen paréntesis balanceados por ser parte de una palabra con paréntesis balanceados. Pero notemos que

$$|w| \;=\; 2(n+1) = 2n+2$$

y como $w = (x)y$ entonces $|x| \leq 2n \wedge |y| \leq 2n$. Por la hipótesis inductiva

$$\begin{aligned} |x| \leq 2n &\implies S \Rightarrow^* x \\ |y| \leq 2n &\implies S \Rightarrow^* y \end{aligned}$$

y usando G podemos derivar

$$S \Rightarrow (S)S \Rightarrow^* (x)S \Rightarrow^* (x)y = w$$

Concluimos que toda palabra con paréntesis balanceados puede ser derivada por la gramática G.

Gramáticas Ambiguas

Una gramática es ambigua cuando existe al menos una cadena que puede ser derivada de más de una forma, bien sea con derivaciones más izquierdas o más derechas. Por ejemplo, la gramática

$$\begin{aligned} E &\rightarrow E + E \\ E &\rightarrow E * E \\ E &\rightarrow (E) \\ E &\rightarrow \mathbf{id} \end{aligned}$$

es ambigua, pues si consideramos la cadena $\mathbf{id} + \mathbf{id} * \mathbf{id}$ podemos obtener dos derivaciones más izquierdas diferentes:

Caso 1

$$\begin{aligned} E &\Rightarrow E + E \\ &\Rightarrow \mathbf{id} + E \\ &\Rightarrow \mathbf{id} + E * E \\ &\Rightarrow \mathbf{id} + \mathbf{id} * E \\ &\Rightarrow \mathbf{id} + \mathbf{id} * \mathbf{id} \end{aligned}$$

que tiene el árbol de derivación. Nótese que ésta derivación coincide con la precedencia de operadores natural para la aritmética, i.e. la multiplicación tiene mayor precedencia que la adición.

Caso 2

$$\begin{aligned} E &\Rightarrow E * E \\ &\Rightarrow E + E * E \\ &\Rightarrow \mathbf{id} + E * E \\ &\Rightarrow \mathbf{id} + \mathbf{id} * E \\ &\Rightarrow \mathbf{id} + \mathbf{id} * \mathbf{id} \end{aligned}$$

que tiene el árbol de derivación. Nótese que ésta derivación no coincide con la precedencia de operadores natural para la aritmética pues la adición tiene mayor precedencia que la multiplicación.

En muchos casos, es posible reescribir una gramática de manera que desaparezca la ambigüedad. Sin embargo no hay un algoritmo general que garantice que eso sea posible. Para esta gramática particular, es posible eliminar la ambigüedad si se escribe como

$$\begin{aligned} E &\rightarrow E + T \\ E &\rightarrow T \\ T &\rightarrow T * F \\ T &\rightarrow F \\ F &\rightarrow (E) \\ F &\rightarrow \mathbf{id} \end{aligned}$$

y si volvemos a considerar la frase $\mathbf{id} + \mathbf{id} * \mathbf{id}$ encontraremos una sola derivación más izquierda, a saber

$$\begin{aligned} E &\Rightarrow E + T \\ &\Rightarrow \mathbf{id} + T \\ &\Rightarrow \mathbf{id} + T * F \\ &\Rightarrow \mathbf{id} + F * F \\ &\Rightarrow \mathbf{id} + \mathbf{id} * F \\ &\Rightarrow \mathbf{id} + \mathbf{id} * \mathbf{id} \end{aligned}$$

con árbol de derivación. Nótese que ésta derivación coincide con la precedencia de operadores deseada.

Ejemplo Completo 1

Sea la gramática G sobre $\Sigma = \{\mathbf{a}, \mathbf{b}\}$ con producciones

$$\begin{aligned} S &\rightarrow S + S \\ &\rightarrow SS \\ &\rightarrow S* \\ &\rightarrow (S) \\ &\rightarrow \mathbf{a} \\ &\rightarrow \mathbf{b} \end{aligned}$$

- Demuestre que G genera el lenguaje de las Expresiones Regulares sobre Σ.

  Primero debemos demostrar que toda palabra derivada a partir de S es una Expresión Regular sobre Σ. Usaremos inducción sobre el número de pasos de derivación.

  Caso Base

  Sólo hay dos derivaciones de un paso

  $$\begin{aligned} S &\Rightarrow \mathbf{a} \\ S &\Rightarrow \mathbf{b} \end{aligned}$$

  por lo que todo lo derivable a partir de S en un paso es una Expresión Regular en Σ.

  Hipótesis Inductiva

  Toda derivación de n o menos pasos produce una frase que es una Expresión Regular en Σ.

  Inducción

  Consideremos una derivación de $n+1$ pasos, entonces debemos considerar cuatro posibilidades:

  - Es de la forma
    $$S \Rightarrow S + S \Rightarrow^* x + S \Rightarrow x + y$$
    pero la derivación $S \Rightarrow^* x$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva x es una Expresión Regular en Σ. Así mismo la derivación $S \Rightarrow^* y$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva y es una Expresión Regular en Σ. En consecuencia $x + y$ es una Expresión Regular en Σ.
  - Es de la forma
    $$S \Rightarrow SS \Rightarrow^* xS \Rightarrow^* xy$$
    pero la derivación $S \Rightarrow^* x$ tiene n o menos pasos y de acuerdo con la hipótesis inductiva x es una Expresión Regular en Σ. Así mismo la derivación $S \Rightarrow^* y$ tiene n o menos pasos y de acuerdo con la hipótesis inductiva y es una Expresión Regular en Σ. En consecuencia xy es una Expresión Regular en Σ.

- Es de la forma
$$S \Rightarrow S* \Rightarrow^* x*$$
pero la derivación $S \Rightarrow^* x$ tiene exactamente n pasos y de acuerdo con la hipótesis inductiva x es una Expresión Regular en Σ. En consecuencia $x*$ es una Expresión Regular en Σ.
- Es de la forma
$$S \Rightarrow (S) \Rightarrow^* (x)$$
pero la derivación $S \Rightarrow^* x$ tiene exactamente n pasos y de acuerdo con la hipótesis inductiva x es una Expresión Regular en Σ. En consecuencia (x) es una Expresión Regular en Σ.

Ahora debemos demostrar que si la palabra w es una Expresión Regular en Σ, es posible derivarla a partir de S. Usaremos inducción sobre la longitud de la palabra w.

Caso Base

Si w es una Expresión Regular en Σ y $|w| = 1$ entonces tiene que ser

$$\begin{array}{lll} w = \mathbf{a} & \text{y en ese caso} & S \Rightarrow \mathbf{a} \\ w = \mathbf{b} & \text{y en ese caso} & S \Rightarrow \mathbf{b} \end{array}$$

Hipótesis Inductiva

Para toda w Expresión Regular en Σ tal que $|w| \leq n$, existe una derivación $S \Rightarrow^* w$.

Inducción

Consideremos w Expresión Regular en Σ tal que $|w| = n + 1$, entonces debemos considerar cuatro posibilidades:

- Es de la forma $w = x + y$. En ese caso notamos que $|x| < n$ por tanto existe $S \Rightarrow^* x$, así como también $|y| < n$ por tanto existe $S \Rightarrow^* y$. En consecuencia podemos derivar
$$S \Rightarrow S + S \Rightarrow^* x + S \Rightarrow^* x + y$$
- Es de la forma $w = xy$. En ese caso notamos que $|x| \leq n$ por tanto existe $S \Rightarrow^* x$, así como también $|y| \leq n$ por tanto existe $S \Rightarrow^* y$. En consecuencia podemos derivar
$$S \Rightarrow SS \Rightarrow^* xS \Rightarrow^* xy$$
- Es de la forma $w = x*$. En ese caso notamos que $|x| = n$ por tanto existe $S \Rightarrow^* x$. En consecuencia podemos derivar
$$S \Rightarrow S* \Rightarrow^n x*$$
- Es de la forma $w = (x)$. En ese caso notamos que $|x| = n - 1$ por tanto existe $S \Rightarrow^* x$. En consecuencia podemos derivar
$$S \Rightarrow (S) \Rightarrow^{n-1} (x)$$

- Demuestre que la gramática G es ambigua.

  Consideremos la frase $ab * a$ y apliquemos derivaciones más izquierdas:

  Caso 1

$$\begin{array}{lll} S & \Rightarrow & SS \\ & \Rightarrow & \mathbf{a}S \\ & \Rightarrow & \mathbf{a}SS \\ & \Rightarrow & \mathbf{a}S * S \\ & \Rightarrow & \mathbf{ab} * S \\ & \Rightarrow & \mathbf{ab} * \mathbf{a} \end{array}$$

Caso 2

$$\begin{aligned} S &\Rightarrow SS \\ &\Rightarrow SSS \\ &\Rightarrow \mathbf{a}SS \\ &\Rightarrow \mathbf{a}S*S \\ &\Rightarrow \mathbf{ab}*S \\ &\Rightarrow \mathbf{ab}*\mathbf{a} \end{aligned}$$

- Reescriba la gramática para eliminar la ambiguedad de manera que se preserve la precedencia acostumbrada para Expresiones Regulares, i.e. las expresiones entre paréntesis tiene máxima precedencia, seguidas de las clausuras reflexiva-transitivas, seguidas de la concatenación y finalmente la unión.

  En este caso es posible reescribir la gramática, con una estructura muy similar a la de expresiones aritméticas que estudiamos previamente. La gramática quedaría

$$\begin{aligned} S &\rightarrow S+C \\ S &\rightarrow C \\ C &\rightarrow CE \\ C &\rightarrow E \\ E &\rightarrow T* \\ E &\rightarrow T \\ T &\rightarrow (S) \\ T &\rightarrow \mathbf{a} \\ T &\rightarrow \mathbf{b} \end{aligned}$$

  y si volvemos a considerar la frase $\mathbf{ab}*\mathbf{a}$ encontraremos una sola derivación izquierda, a saber

$$\begin{aligned} S &\Rightarrow C \\ &\Rightarrow CE \\ &\Rightarrow CEE \\ &\Rightarrow EEE \\ &\Rightarrow TEE \\ &\Rightarrow \mathbf{a}EE \\ &\Rightarrow \mathbf{a}T*E \\ &\Rightarrow \mathbf{ab}*E \\ &\Rightarrow \mathbf{ab}*T \\ &\Rightarrow \mathbf{ab}*\mathbf{a} \end{aligned}$$

Ejemplo Completo 2

Sea la gramática G sobre $\Sigma = \{\mathbf{a}, \mathbf{b}\}$ con producciones

$$\begin{aligned} S &\rightarrow aSbS \\ S &\rightarrow bSaS \\ S &\rightarrow \lambda \end{aligned}$$

- Demuestre que G genera el lenguaje de palabras sobre Σ con igual cantidad de **a** y **b** sin importar su orden.

  Primero debemos demostrar que toda palabra derivada a partir de S es tiene la misma cantidad de **a** y **b** sin importar su orden. Usaremos inducción sobre el número de pasos de derivación.

  Caso Base

  La única derivación de un paso a partir de S es $S \Rightarrow \lambda$ y λ tiene la misma cantidad de **a** y **b**.

  Hipótesis Inductiva

  Toda derivación de n o menos pasos produce una frase que tiene la misma cantidad de **a** y **b** sin importar su orden.

  Inducción

  Consideremos una derivación de $n+1$ pasos, entonces debemos considerar dos posibilidades:

  - Es de la forma
  $$S \Rightarrow aSbS \Rightarrow^* axbS \Rightarrow^* axby$$
  pero la derivación $S \Rightarrow^* x$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva x tiene la misma cantidad de **a** y **b** sin importar su orden. Así mismo la derivación $S \Rightarrow^* y$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva y tiene la misma cantidad de **a** y **b** sin importar su orden. En consecuencia $axby$ tiene la misma cantidad de **a** y **b** sin importar su orden.
  - Es de la forma
  $$S \Rightarrow bSaS \Rightarrow^* bxaS \Rightarrow^* bxay$$
  pero la derivación $S \Rightarrow^* x$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva x tiene la misma cantidad de **a** y **b** sin importar su orden. Así mismo la derivación $S \Rightarrow^* y$ tiene menos de n pasos y de acuerdo con la hipótesis inductiva y tiene la misma cantidad de **a** y **b** sin importar su orden. En consecuencia $bxay$ tiene la misma cantidad de **a** y **b** sin importar su orden.

  Ahora debemos demostrar que si la palabra w tiene la misma cantidad de **a** y **b** sin importar su orden, es posible derivarla a partir de S. Usaremos inducción sobre la longitud de la palabra w.

  Caso Base

  Si w tiene la misma cantidad de **a** y **b** sin importar su orden, y $|w| = 0$ entonces $w = \lambda$ y en ese caso podemos derivar $S \Rightarrow \lambda$.

  Hipótesis Inductiva

  Si w tiene la misma cantidad de **a** y **b** sin importar su orden entonces su longitud es necesariamente par. Entonces asumiremos que si$|w| \leq 2n$ entonces existe una derivación $S \Rightarrow^* w$.

  Inducción

  Consideremos w con la misma cantidad de **a** y **b** sin importar su orden tal que $|w| = 2(n+1)$. Debemos considerar dos posibilidades:

  - Si w comienza con una **a**, por tener la misma cantidad de **a** y **b** sin importar su orden, entonces podemos escribir $w = \mathbf{a}x\mathbf{b}y$. Más aún, escojamos x tal que $\mathbf{a}x\mathbf{b}$ sea el prefijo izquierdo más corto con la misma cantidad de **a** y **b**, así x e y tienen la misma cantidad de **a** y **b** por pertenecer a una cadena con la misma propiedad. Pero notemos que
  $$|w| = 2(n+1) = 2n+2$$
  y como $w = \mathbf{a}x\mathbf{b}y$ entonces $|x| \leq 2n \wedge |y| \leq 2n$. Por la hipótesis inductiva
  $$\begin{aligned} |x| \leq 2n &\implies S \Rightarrow^* x \\ |y| \leq 2n &\implies S \Rightarrow^* y \end{aligned}$$
  y usando G podemos derivar
  $$S \Rightarrow \mathbf{a}S\mathbf{b}S \Rightarrow^* \mathbf{a}x\mathbf{b}S \Rightarrow^* \mathbf{a}x\mathbf{b}y = w$$

- Si w comienza con una **b**, por tener la misma cantidad de **a** y **b** sin importar su orden, entonces podemos escribir $w = \mathbf{b}x\mathbf{a}y$. Más aún, escojamos x tal que $\mathbf{b}x\mathbf{a}$ sea el prefijo izquierdo más corto con la misma cantidad de **a** y **b**, así x e y tienen la misma cantidad de **a** y **b** por pertenecer a una cadena con la misma propiedad. Pero notemos que

$$|w| = 2(n+1) = 2n+2$$

y como $w = \mathbf{b}x\mathbf{a}y$ entonces $|x| \leq 2n \wedge |y| \leq 2n$. Por la hipótesis inductiva

$$\begin{array}{rcl} |x| \leq 2n & \Longrightarrow & S \Rightarrow^* x \\ |y| \leq 2n & \Longrightarrow & S \Rightarrow^* y \end{array}$$

y usando G podemos derivar

$$S \Rightarrow \mathbf{b}S\mathbf{a}S \Rightarrow^* \mathbf{b}x\mathbf{a}S \Rightarrow^* \mathbf{b}x\mathbf{a}y = w$$

- Demuestre que la gramática G es ambigua

Consideremos la frase **abab** y apliquemos derivaciones más izquierdas:

Caso 1

$$\begin{array}{rcl} S & \Rightarrow & \mathbf{a}S\mathbf{b}S \\ & \Rightarrow & \mathbf{ab}S\mathbf{a}S\mathbf{b}S \\ & \Rightarrow & \mathbf{aba}S\mathbf{b}S \\ & \Rightarrow & \mathbf{abab}S \\ & \Rightarrow & \mathbf{abab} \end{array}$$

Caso 2

$$\begin{array}{rcl} S & \Rightarrow & \mathbf{a}S\mathbf{b}S \\ & \Rightarrow & \mathbf{ab}S \\ & \Rightarrow & \mathbf{aba}S\mathbf{b}S \\ & \Rightarrow & \mathbf{abab}S \\ & \Rightarrow & \mathbf{abab} \end{array}$$

- Reescriba la gramática para eliminar la ambiguedad.

En este caso es posible reescribir la gramática cuyo funcionamiento intuitivo es el siguiente:

1. El lenguaje contiene a λ de modo que el símbolo inicial debe ser capaz de producirlo.
2. Si a partir del símbolo inicial encontramos una **a** entonces estamos en la necesidad de una **b** para "balancear". Denotaremos B esa necesidad.
3. Si a partir del símbolo inicial encontramos una **b** entonces estamos en la necesidad de una **a** para "balancear". Denotaremos A esa necesidad.
4. Si estamos en B y encontramos una **b** entonces estamos "balanceados", por lo que podemos regresar al símbolo inicial. Pero si encontramos una **a** entonces necesitamos **dos b** para "balancear" por lo que hemos de cumplir dos veces con B.
5. Si estamos en A y encontramos una **a** entonces estamos "balanceados", por lo que podemos regresar al símbolo inicial. Pero si encontramos una **b** entonces necesitamos **dos a** para "balancear" por lo que hemos de cumplir dos veces con A.

La gramática nos queda como

$$\begin{array}{rcl}
S & \rightarrow & \mathbf{a}B \\
S & \rightarrow & \mathbf{b}A \\
S & \rightarrow & \lambda \\
A & \rightarrow & \mathbf{a}S \\
A & \rightarrow & \mathbf{b}AA \\
B & \rightarrow & \mathbf{b}S \\
B & \rightarrow & \mathbf{a}BB
\end{array}$$

y si volvemos a considerar la frase **abab** encontraremos una sola derivación izquierda, a saber

$$\begin{array}{rcl}
S & \Rightarrow & \mathbf{a}B \\
 & \Rightarrow & \mathbf{ab}S \\
 & \Rightarrow & \mathbf{aba}B \\
 & \Rightarrow & \mathbf{abab}S \\
 & \Rightarrow & \mathbf{abab}
\end{array}$$