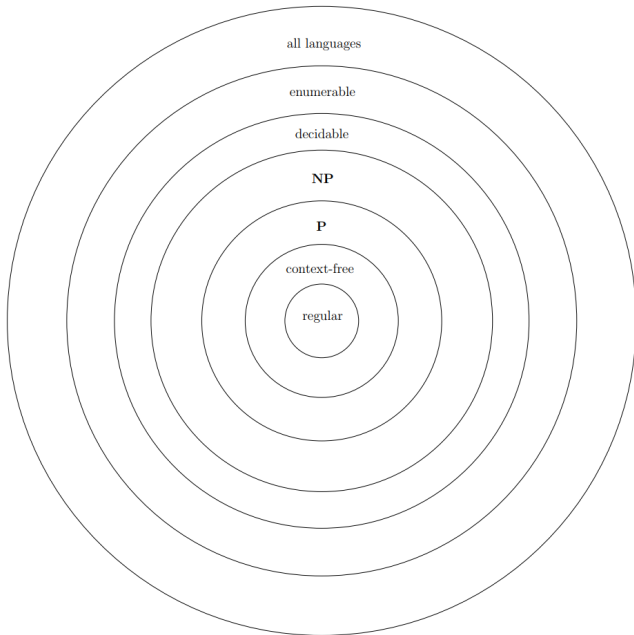


Gramáticas Libres de Contexto

Matemáticas Computacionales
(TC2020)

M.C. Xavier Sánchez Díaz
mail@tec.mx





El caso $a^n b^n$

Lenguajes libres de contexto

Tesis

Sean $L = \{a^n b^n, n \in \mathbb{N}\}$ y LR el conjunto de los lenguajes regulares. Entonces, $L \notin LR$.

Como un autómata debe tener un número *finito* de estados, no hay manera de hacer que recuerde cuántas *as* van para saber cuántas *bs* deberíamos tener.

El caso $a^n b^n$

Lenguajes libres de contexto

Tesis

Sean $L = \{a^n b^n, n \in \mathbb{N}\}$ y LR el conjunto de los lenguajes regulares. Entonces, $L \notin LR$.

Como un autómata debe tener un número *finito* de estados, no hay manera de hacer que recuerde cuántas *as* van para saber cuántas *bs* deberíamos tener.

GRs y GLCs

Diferencias entre GRs y GLCs

Las **reglas** en una **GR** son del modo $A \rightarrow bC$ o $A \rightarrow b$.

En una **Gramática Libre de Contexto** (GLC) son del modo $A \rightarrow bCd$ o $A \rightarrow b$.

Además, incluimos nuevas reglas: $A \rightarrow \varepsilon$

GRs y GLCs

Diferencias entre GRs y GLCs

Las **reglas** en una **GR** son del modo $A \rightarrow bC$ o $A \rightarrow b$.

En una **Gramática Libre de Contexto** (GLC) son del modo $A \rightarrow bCd$ o $A \rightarrow b$.

Además, incluimos nuevas reglas: $A \rightarrow \varepsilon$

GRs y GLCs

Diferencias entre GRs y GLCs

Las **reglas** en una **GR** son del modo $A \rightarrow bC$ o $A \rightarrow b$.

En una **Gramática Libre de Contexto** (GLC) son del modo $A \rightarrow bCd$ o $A \rightarrow b$.

Además, incluimos nuevas reglas: $A \rightarrow \varepsilon$

GRs y GLCs

Diferencias entre GRs y GLCs

Las **reglas** en una **GR** son del modo $A \rightarrow bC$ o $A \rightarrow b$.

En una **Gramática Libre de Contexto** (GLC) son del modo $A \rightarrow bCd$ o $A \rightarrow b$.

Además, incluimos nuevas reglas: $A \rightarrow \varepsilon$

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

¿Qué patrón podemos identificar en los ejemplos?

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: ε , ab , $aabb$, $aaabbb$, \dots

Estructura: Grupos anidados de una a al inicio y una b al final.

Volvemos al caso $a^n b^n$

GLCs bien formadas

Palabras aceptadas: $\varepsilon, ab, aabb, aaabbb, \dots$

Estructura: Grupos anidados de una a al inicio y una b al final.

GLC:

$$\textcircled{1} S \rightarrow aSb$$

$$\textcircled{2} S \rightarrow \varepsilon$$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $(((\dots)))$
- Cada grupo de $(((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- $S \rightarrow (S)$
- $S \rightarrow ()$
- $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $((((\dots)))$
- Cada grupo de $((((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- $S \rightarrow (S)$
- $S \rightarrow ()$
- $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $(((\dots)))$
- Cada grupo de $(((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- $S \rightarrow (S)$
- $S \rightarrow ()$
- $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $((((\dots)))$
- Cada grupo de $((((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- 1 $S \rightarrow (S)$
- 2 $S \rightarrow ()$
- 3 $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $((((\dots)))$
- Cada grupo de $((((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

① $S \rightarrow (S)$

② $S \rightarrow ()$

③ $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $(((\dots)))$
- Cada grupo de $(((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

¿Cómo concatenamos dos grupos?

Reglas:

- 1 $S \rightarrow (S)$
- 2 $S \rightarrow ()$
- 3 $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $((((\dots)))$
- Cada grupo de $((((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- 1 $S \rightarrow (S)$
- 2 $S \rightarrow ()$
- 3 $S \rightarrow SS$

Ejemplo: *Matching Parenthesis*

GLCs bien formadas

Ejemplos: $()$, $(())$, $()()$, $(())()$, \dots

Estructura:

- Grupos concatenados de paréntesis anidados $((((\dots)))$
- Cada grupo de $((((\dots)))$ es similar a $\{a^n b^n\}$
- Concatenamos usando una regla de tipo $S \rightarrow SS$

Reglas:

- 1 $S \rightarrow (S)$
- 2 $S \rightarrow ()$
- 3 $S \rightarrow SS$

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$① \quad E \rightarrow E + T$$

$$② \quad E \rightarrow T$$

$$③ \quad T \rightarrow T * F$$

$$④ \quad T \rightarrow F$$

$$⑤ \quad F \rightarrow CF$$

$$⑥ \quad F \rightarrow C$$

$$⑦ \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

1 $E \rightarrow E + T$

2 $E \rightarrow T$

3 $T \rightarrow T * F$

4 $T \rightarrow F$

5 $F \rightarrow CF$

6 $F \rightarrow C$

7 $C \rightarrow 0|1|2|3|4|5|6|7|8|9$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

❶ $E \rightarrow E + T$

❷ $E \rightarrow T$

❸ $T \rightarrow T * F$

❹ $T \rightarrow F$

❺ $F \rightarrow CF$

❻ $F \rightarrow C$

❼ $C \rightarrow 0|1|2|3|4|5|6|7|8|9$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

① $E \rightarrow E + T$

② $E \rightarrow T$

③ $T \rightarrow T * F$

④ $T \rightarrow F$

⑤ $F \rightarrow CF$

⑥ $F \rightarrow C$

⑦ $C \rightarrow 0|1|2|3|4|5|6|7|8|9$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$① \quad E \rightarrow E + T$$

$$② \quad E \rightarrow T$$

$$③ \quad T \rightarrow T * F$$

$$④ \quad T \rightarrow F$$

$$⑤ \quad F \rightarrow CF$$

$$⑥ \quad F \rightarrow C$$

$$⑦ \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$\textcircled{1} \quad E \rightarrow E + T$$

$$\textcircled{2} \quad E \rightarrow T$$

$$\textcircled{3} \quad T \rightarrow T * F$$

$$\textcircled{4} \quad T \rightarrow F$$

$$\textcircled{5} \quad F \rightarrow CF$$

$$\textcircled{6} \quad F \rightarrow C$$

$$\textcircled{7} \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$\textcircled{1} \quad E \rightarrow E + T$$

$$\textcircled{2} \quad E \rightarrow T$$

$$\textcircled{3} \quad T \rightarrow T * F$$

$$\textcircled{4} \quad T \rightarrow F$$

$$\textcircled{5} \quad F \rightarrow CF$$

$$\textcircled{6} \quad F \rightarrow C$$

$$\textcircled{7} \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$\textcircled{1} \quad E \rightarrow E + T$$

$$\textcircled{2} \quad E \rightarrow T$$

$$\textcircled{3} \quad T \rightarrow T * F$$

$$\textcircled{4} \quad T \rightarrow F$$

$$\textcircled{5} \quad F \rightarrow CF$$

$$\textcircled{6} \quad F \rightarrow C$$

$$\textcircled{7} \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$\textcircled{1} \quad E \rightarrow E + T$$

$$\textcircled{2} \quad E \rightarrow T$$

$$\textcircled{3} \quad T \rightarrow T * F$$

$$\textcircled{4} \quad T \rightarrow F$$

$$\textcircled{5} \quad F \rightarrow CF$$

$$\textcircled{6} \quad F \rightarrow C$$

$$\textcircled{7} \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: expresiones aritméticas

GLCs bien formadas

Podemos formar expresiones aritméticas también, como $25 + 3 * 12$, donde la multiplicación debe tener precedencia sobre la suma:

$$\textcircled{1} \quad E \rightarrow E + T$$

$$\textcircled{2} \quad E \rightarrow T$$

$$\textcircled{3} \quad T \rightarrow T * F$$

$$\textcircled{4} \quad T \rightarrow F$$

$$\textcircled{5} \quad F \rightarrow CF$$

$$\textcircled{6} \quad F \rightarrow C$$

$$\textcircled{7} \quad C \rightarrow 0|1|2|3|4|5|6|7|8|9$$

¿Cómo derivamos $25 + 3 * 12$? ¿Y $5 * 2 + 26$?

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

- 1 $S \rightarrow aSa$
- 2 $S \rightarrow bSb$
- 3 $S \rightarrow a$
- 4 $S \rightarrow b$
- 5 $S \rightarrow \varepsilon$

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

¿Cómo hacemos si es par, e.g. *abba*?

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

- ① $S \rightarrow aSa$
- ② $S \rightarrow bSb$
- ③ $S \rightarrow a$
- ④ $S \rightarrow b$
- ⑤ $S \rightarrow \varepsilon$

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

① $S \rightarrow aSa$

② $S \rightarrow bSb$

③ $S \rightarrow a$

④ $S \rightarrow b$

⑤ $S \rightarrow \varepsilon$

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

¿Cómo hacemos si es impar, e.g. *abbba*?

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

① $S \rightarrow aSa$

② $S \rightarrow bSb$

③ $S \rightarrow a$

④ $S \rightarrow b$

⑤ $S \rightarrow \varepsilon$

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

① $S \rightarrow aSa$

② $S \rightarrow bSb$

③ $S \rightarrow a$

④ $S \rightarrow b$

⑤ $S \rightarrow \varepsilon$

Ejemplo: palíndromos en $\{a, b\}$

GLCs bien formadas

Ejemplos: $\varepsilon, a, aa, aaa, aba, abba, \dots$

Caso Par: parecido a $\{a^n b^n\}$ pero con lo mismo en cada lado: $S \rightarrow aSa$ y $S \rightarrow bSb$, y usando $S \rightarrow \varepsilon$ para salir del *loop*.

Caso impar: idéntico al caso par, sólo que el *loop* se termina con un terminal no vacío: $S \rightarrow a, S \rightarrow b$.

① $S \rightarrow aSa$

② $S \rightarrow bSb$

③ $S \rightarrow a$

④ $S \rightarrow b$

⑤ $S \rightarrow \varepsilon$

Combinación de GLCs

Operaciones en GLCs

¿Cómo **unimos** dos GLCs?

Ejemplo: GLC para las palabras de forma $a^n b^m$ tal que $n \neq m$

Solución:

$$\{a^n b^m, n \neq m\} = \{a^n b^m, n > m\} \cup \{a^n b^m, n < m\}$$

Usamos un **símbolo inicial nuevo** para hacer dos nuevas reglas: $S_0 \rightarrow S_1$ y $S_0 \rightarrow S_2$, donde S_1 y S_2 son los símbolos iniciales de las GLCs 1 y 2, respectivamente.

La **concatenación** es similar. Agregamos una nueva regla: $S_0 \rightarrow S_1 S_2$.

Combinación de GLCs

Operaciones en GLCs

¿Cómo **unimos** dos GLCs?

Ejemplo: GLC para las palabras de forma $a^n b^m$ tal que $n \neq m$

Solución:

$$\{a^n b^m, n \neq m\} = \{a^n b^m, n > m\} \cup \{a^n b^m, n < m\}$$

Usamos un **símbolo inicial nuevo** para hacer dos nuevas reglas: $S_0 \rightarrow S_1$ y $S_0 \rightarrow S_2$, donde S_1 y S_2 son los símbolos iniciales de las GLCs 1 y 2, respectivamente.

La **concatenación** es similar. Agregamos una nueva regla: $S_0 \rightarrow S_1 S_2$.

Combinación de GLCs

Operaciones en GLCs

¿Cómo **unimos** dos GLCs?

Ejemplo: GLC para las palabras de forma $a^n b^m$ tal que $n \neq m$

Solución:

$$\{a^n b^m, n \neq m\} = \{a^n b^m, n > m\} \cup \{a^n b^m, n < m\}$$

Usamos un **símbolo inicial nuevo** para hacer dos nuevas reglas: $S_0 \rightarrow S_1$ y $S_0 \rightarrow S_2$, donde S_1 y S_2 son los símbolos iniciales de las GLCs 1 y 2, respectivamente.

La **concatenación** es similar. Agregamos una nueva regla: $S_0 \rightarrow S_1 S_2$.

Combinación de GLCs

Operaciones en GLCs

¿Cómo **unimos** dos GLCs?

Ejemplo: GLC para las palabras de forma $a^n b^m$ tal que $n \neq m$

Solución:

$$\{a^n b^m, n \neq m\} = \{a^n b^m, n > m\} \cup \{a^n b^m, n < m\}$$

Usamos un **símbolo inicial nuevo** para hacer dos nuevas reglas: $S_0 \rightarrow S_1$ y $S_0 \rightarrow S_2$, donde S_1 y S_2 son los símbolos iniciales de las GLCs 1 y 2, respectivamente.

La **concatenación** es similar. Agregamos una nueva regla: $S_0 \rightarrow S_1 S_2$.

Combinación de GLCs

Operaciones en GLCs

¿Cómo **unimos** dos GLCs?

Ejemplo: GLC para las palabras de forma $a^n b^m$ tal que $n \neq m$

Solución:

$$\{a^n b^m, n \neq m\} = \{a^n b^m, n > m\} \cup \{a^n b^m, n < m\}$$

Usamos un **símbolo inicial nuevo** para hacer dos nuevas reglas: $S_0 \rightarrow S_1$ y $S_0 \rightarrow S_2$, donde S_1 y S_2 son los símbolos iniciales de las GLCs 1 y 2, respectivamente.

La **concatenación** es similar. Agregamos una nueva regla: $S_0 \rightarrow S_1 S_2$.

Ejemplo de ambigüedad: $\#a = \#b$

Ambigüedad y Refinamiento de GLCs

Estructura: grupos de $a \dots b$ o $b \dots a$.

Reglas:

- 1 $S \rightarrow aSb$
- 2 $S \rightarrow bSa$
- 3 $S \rightarrow SS$
- 4 $S \rightarrow \varepsilon$

Probar **derivaciones** usando *DFS* para $abab = \langle 3, 1, 4, 1, 4 \rangle$ y $abab = \langle 1, 2, 4 \rangle$.

Ejemplo de ambigüedad: $\#a = \#b$

Ambigüedad y Refinamiento de GLCs

Estructura: grupos de $a \dots b$ o $b \dots a$.

Reglas:

- ① $S \rightarrow aSb$
- ② $S \rightarrow bSa$
- ③ $S \rightarrow SS$
- ④ $S \rightarrow \varepsilon$

Probar **derivaciones** usando *DFS* para $abab = \langle 3, 1, 4, 1, 4 \rangle$ y $abab = \langle 1, 2, 4 \rangle$.

Ejemplo de ambigüedad: $\#a = \#b$

Ambigüedad y Refinamiento de GLCs

Estructura: grupos de $a \dots b$ o $b \dots a$.

Reglas:

- ① $S \rightarrow aSb$
- ② $S \rightarrow bSa$
- ③ $S \rightarrow SS$
- ④ $S \rightarrow \varepsilon$

Probar **derivaciones** usando *DFS* para $abab = \langle 3, 1, 4, 1, 4 \rangle$ y $abab = \langle 1, 2, 4 \rangle$.

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Es posible reemplazar reglas en una GLC sin alterar el conjunto de palabras que se pueden formar con ellas.

Por ejemplo, aquellas que generen producciones vacías: $A \rightarrow \varepsilon$

Producciones 'inútiles' para reducir el tamaño de las derivaciones: $A \rightarrow B$ y $B \rightarrow a$.

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Es posible reemplazar reglas en una GLC sin alterar el conjunto de palabras que se pueden formar con ellas.

Por ejemplo, aquellas que generen producciones vacías: $A \rightarrow \varepsilon$

Producciones 'inútiles' para reducir el tamaño de las derivaciones: $A \rightarrow B$ y $B \rightarrow a$.

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Es posible reemplazar reglas en una GLC sin alterar el conjunto de palabras que se pueden formar con ellas.

Por ejemplo, aquellas que generen producciones vacías: $A \rightarrow \varepsilon$

Producciones 'inútiles' para reducir el tamaño de las derivaciones: $A \rightarrow B$ y $B \rightarrow a$.

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Existen algunos problemas con las reglas vacías:

Son **ineficientes**, pues crean un símbolo para después destruirlo.

Las reglas vacías pueden **crecer o decrecer** las derivaciones, e.g. derivaciones de ab

$$S \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow S \rightarrow aSb \rightarrow ab$$

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Existen algunos problemas con las reglas vacías:

Son **ineficientes**, pues crean un símbolo para después destruirlo.

Las reglas vacías pueden **crecer o decrecer** las derivaciones, e.g. derivaciones de ab

$$S \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow S \rightarrow aSb \rightarrow ab$$

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Existen algunos problemas con las reglas vacías:

Son **ineficientes**, pues crean un símbolo para después destruirlo.

Las reglas vacías pueden **crecer o decrecer** las derivaciones, e.g. derivaciones de ab

$$S \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow S \rightarrow aSb \rightarrow ab$$

Eliminación de reglas

Ambigüedad y Refinamiento de GLCs

Existen algunos problemas con las reglas vacías:

Son **ineficientes**, pues crean un símbolo para después destruirlo.

Las reglas vacías pueden **crecer o decrecer** las derivaciones, e.g. derivaciones de ab

$$S \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow SSS \rightarrow SS \rightarrow S \rightarrow aSb \rightarrow ab$$

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas vacías

Ambigüedad y Refinamiento de GLCs

Para eliminar reglas de producciones vacías $A \rightarrow \varepsilon$, se reemplaza el lado derecho de cada ocurrencia de una regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow \varepsilon$$

donde podemos reemplazar cada S por ε , para generar una nueva regla:

$$\textcircled{1} \quad S \rightarrow aSb$$

$$aabb = \langle 1, 2 \rangle$$

$$\textcircled{2} \quad S \rightarrow ab$$

¿Qué pasa si el lenguaje $L(G)$ sí contiene a la palabra vacía ε ?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow V$

④ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

❶ $S \rightarrow aSb$

❷ $S \rightarrow ab$

❸ $S \rightarrow V$

❹ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

❶ $S \rightarrow aSb$

❷ $S \rightarrow ab$

❸ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow V$

④ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow V$

④ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow V$

④ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?

Eliminación de reglas 'inútiles'

Ambigüedad y Refinamiento de GLCs

Para eliminar las reglas 'inútiles' (o pasos intermedios) en una GLC hay que ir **conectando** los extremos:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow V$

④ $V \rightarrow a$

En lugar de tener dos reglas $S \rightarrow V$ y $V \rightarrow a$ podemos tener directamente una sola, $S \rightarrow a$:

① $S \rightarrow aSb$

② $S \rightarrow ab$

③ $S \rightarrow a$

¿Qué pasa si hay ciclos en plan $S \rightarrow T$ y $T \rightarrow S$?