Modelos de Computación

Los Del DGIIM, `losdeldgiim.github.io`

Doble Grado en Ingeniería Informática y Matemáticas
Universidad de Granada

Modelos de Computación

Los Del DGIIM, `losdeldgiim.github.io`

Arturo Olivares Martos

Granada, 2024-2025

Índice general

1. Relaciones de Problemas

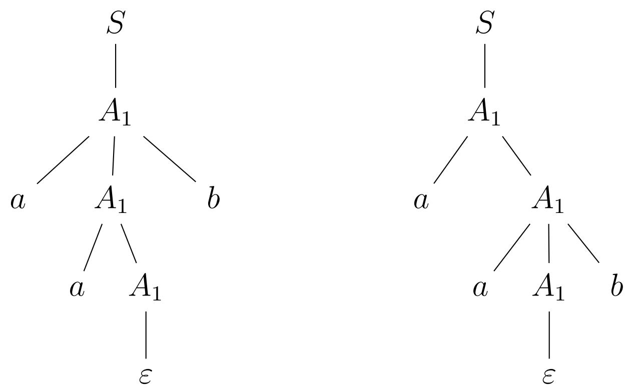

Figura 1.1: Árboles de derivación para aab usando la Gramática del Ejercicio 1.1.1.

1.1. Gramáticas Independientes del Contexto

Observación. Salvo que se indique lo contrario, las letras en mayúsculas representan variables, las letras en minúsculas representan terminales y la S representa el símbolo inicial.

Ejercicio 1.1.1. Determinar si la siguiente gramática es ambigua y si el lenguaje generado es inherentemente ambiguo:

$$\begin{cases} S \to A_1 \mid A_2 \\ A_1 \to aA_1b \mid aA_1 \mid \varepsilon \\ A_2 \to aA_2b \mid A_2b \mid \varepsilon \end{cases}$$

La gramática dada es ambigua puesto que hay palabras con más de un árbol de derivación. Por ejemplo, la palabra aab tiene los dos posibles árboles de derivación que se muestran en la Figura 1.1.

Veamos ahora que no es inherentemente ambiguo. La producción de A_1 produce las palabras de la forma a^ib^j, con $i \geqslant j$. La producción de A_2 produce las palabras de la forma a^ib^j, con $i \leqslant j$. Por tanto, la gramática genera el lenguaje:

$$\begin{aligned} L &= \{a^ib^j \mid i,j \in \mathbb{N}\cup\{0\},\ i \geqslant j\} \cup \{a^ib^j \mid i,j \in \mathbb{N}\cup\{0\},\ i \leqslant j\} = \\ &= \{a^ib^j \mid i,j \in \mathbb{N}\cup\{0\}\} \end{aligned}$$

Ejercicio 1.1.2. Sea la gramática

$$\begin{cases} S \to aSA \mid \varepsilon \\ A \to bA \mid z \mid \varepsilon \end{cases}$$

1. Demostrar que es ambigua.

   Tomemos como palabra $aabb$. Esta palabra tiene dos árboles de derivación, como se muestra en la Figura 1.2.

2. Dar una expresión regular para el lenguaje generado.

   En primer lugar, hemos de considerar que $\varepsilon \in L$. Además, todas las palabras de longitud positiva empiezan por a. por tanto, la expresión regular para el lenguaje generado por la gramática es:

   $$a^+b^*(z+\varepsilon)+\varepsilon$$

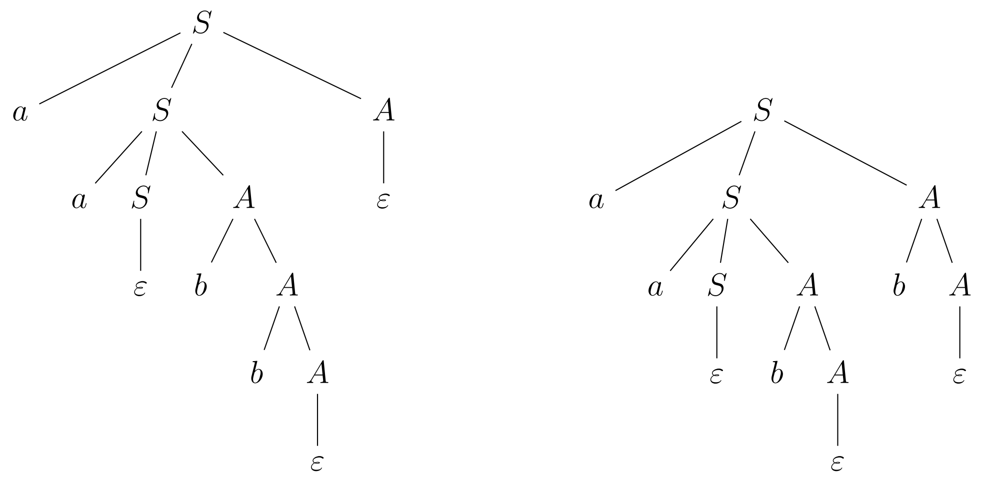

Figura 1.2: Árboles de derivación para $aabb$ usando la Gramática del Ejercicio 1.1.2.

3. Construir una gramática no ambigua que genere el mismo lenguaje.

Ejercicio 1.1.3. Considera la gramática $G = (V, T, S, P)$ donde

$$\begin{aligned}
V &= \{\langle\text{expresión}\rangle, \langle\text{identificador}\rangle\}\\
T &= \{a, b, c, d, -\}\\
S &= \langle\text{expresión}\rangle\\
P &= \left\{\begin{array}{r}
\langle\text{expresión}\rangle \to \langle\text{identificador}\rangle \\
\langle\text{expresión}\rangle \to \langle\text{identificador}\rangle - \langle\text{expresión}\rangle \\
\langle\text{expresión}\rangle \to \langle\text{expresión}\rangle - \langle\text{identificador}\rangle \\
\langle\text{identificador}\rangle \to a \mid b \mid c \mid d
\end{array}\right.
\end{aligned}$$

1. Demuestra que esta gramática no puede ser empleada para describir un posible lenguaje de programación, teniendo en cuenta que la sustración no es una operación conmutativa, y que $(a - b) - d \neq a - (b - d)$.

2. ¿Es ambigua la gramática G? ¿Es la ambigüedad inherente al lenguaje generado por G? Justifica adecuadamente la respuesta.

3. ¿Es posible modificar G de manera que la nueva gramática pueda ser usada para generar el lenguaje de las expresiones aritméticas correctas con el operador de resta?

Ejercicio 1.1.4. Dada la gramática

$$\left\{\begin{array}{l}
S \to A \mid B \\
A \to aaA \mid \varepsilon \\
B \to aaaB \mid \varepsilon
\end{array}\right.$$

1. Demostrar que es ambigua.

   La variable A genera palabras de la forma a^{2i} y la variable B genera palabras de la forma a^{3i}. Por tanto, la palabras de la forma a^{6i} tienen dos árboles de derivación, como se muestra en la Figura 1.3.

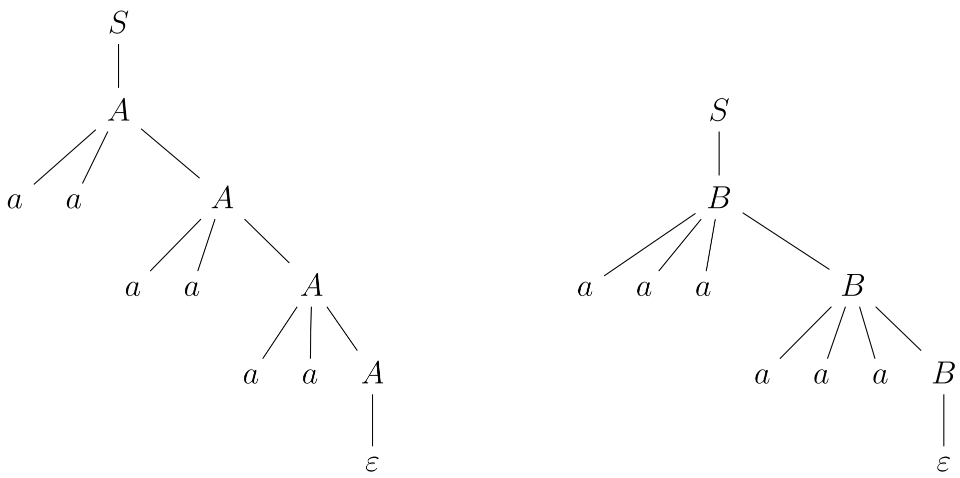

Figura 1.3: Árboles de derivación para a^6 usando la Gramática del Ejercicio 1.1.4.

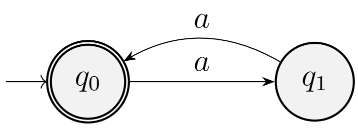

Figura 1.4: AFD que acepta el lenguaje de la variable A de la Gramática del Ejercicio 1.1.4.

2. Construir un autómata finito determinístico que acepte el mismo lenguaje.

   El autómata que genera las palabras de la forma a^{2i} es el que se muestra en la Figura 1.4, mientras que el autómata que genera las palabras de la forma a^{3i} es el que se muestra en la Figura 1.5. El autómata producto es el que se muestra en la Figura 1.6.

3. Construir una gramática lineal por la derecha, a partir del autómata determinístico, que genere el mismo lenguaje.

4. Demostrar que la gramática resultante no es ambigua.

Ejercicio 1.1.5. Dar una gramática libre de contexto no ambigua que genere el lenguaje

$$L = \{a^i b^j a^k b^l \mid (i = j) \vee (k = l)\}$$

Ejercicio 1.1.6. Determinar cuales de las siguientes gramáticas son ambiguas y, en su caso, comprobar si los lenguajes generados son inherentemente ambiguos:

1. $S \to aSb \mid Sb \mid aS \mid a$

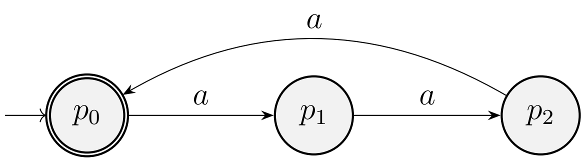

Figura 1.5: AFD que acepta el lenguaje de la variable B de la Gramática del Ejercicio 1.1.4.

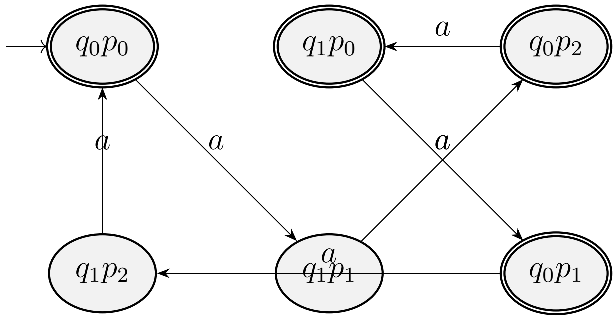

Figura 1.6: AFD que acepta el lenguaje de la Gramática del Ejercicio 1.1.4.

2. $S \to aaS \mid aaaS \mid a$

3. $S \to aS \mid aSb \mid X$,
   $X \to Xa \mid a$

Ejercicio 1.1.7. Dar gramáticas libres de contexto no ambiguas (cuando sea posible) para los siguientes lenguajes sobre el alfabeto $A = \{a, b, c\}$:

1. $L_1 = \{a^i b^j c^k \mid i \neq j \vee j \neq k\}$

2. $L_2 = \{(ab)^i (bc)^j \mid i, j \geqslant 0\}$

3. $L_3 = \{a^i b^{i+j} c^j \mid i, j \geqslant 0\}$

4. L_4 definido como el conjunto de palabras que comienzan por aab y terminan por bbc y tales que estas dos subcadenas no aparecen nunca en el interior de la palabra (sólo están al principio y al final).

Ejercicio 1.1.8. Dada la gramática

$$\left\{ S \to 01S \mid 010S \mid 101S \mid \varepsilon \right.$$

1. Determinar si es ambigua.

2. Construir un autómata finito determinista asociado.

3. Calcular la gramática lineal por la derecha que se obiene a partir del autómata. ¿Es ambigua la gramática resultante?

Ejercicio 1.1.9. Considerar la siguiente gramática:

$$\begin{cases} S \to A_1B \\ A \to 0A \mid \varepsilon \\ B \to 0B \mid 1B \mid \varepsilon \end{cases}$$

1. Demostrar que la gramática dada no es ambigua.

2. Encontrar una gramática para el mismo lenguaje que sea ambigua y demostrar su ambigüedad.

Ejercicio 1.1.10. Describe el lenguaje generado por la siguiente gramática $G = (\{S, A\}, \{a, b\}, P, S)$, con

$$P = \begin{cases} S \to aAa \mid bAa \\ A \to aAa \mid bAa \mid \varepsilon \end{cases}$$

1. Demuestra que el lenguaje generado por la gramática no es regular, pero si independiente del contexto.

2. Normaliza la gramática G en la Forma Normal de Greibach, y determina todas las derivaciones más a la izquierda para la cadena ab^2a^5.

Ejercicio 1.1.11. Obtener la forma normal de Greibach para la siguiente gramática:

$$G = \{\{S_1, S_2, S_3\}, \{a, b, c, d, e\}, S_1, P\}$$

donde:

$$P = \begin{cases} S_1 \to S_1S_2c \mid S_3 \mid S_3bS_3 \\ S_2 \to S_1S_1 \mid d \\ S_3 \to S_2e \end{cases}$$

Ejercicio 1.1.12. Pasar a forma normal de Greibach la gramática

$$\begin{cases} S \to AAA \mid B \\ A \to aA \mid B \\ B \to \varepsilon \end{cases}$$

Ejercicio 1.1.13. Determina si los siguientes lenguajes son regulares o independientes del contexto. Encuentra una gramática que los genere.

1. $L_1 = \{a^ib^jc^k \mid i, j \geqslant 0, k < i + j\}$

2. $L_2 = \{(ab)^ic^jd \mid j = i - 1, i \geqslant 1\}$

3. $L_3 = \{ab^icd^j \mid j = 2 \cdot i, 1 \leqslant i \leqslant 10\}$

Elige una de ellas que sea independiente del contexto y pásala a forma normal de Chomsky.

Ejercicio 1.1.14. Dadas las siguientes gramáticas determinar si son ambiguas y, en caso de que lo sean, determinar una gramática no ambigua que genere el mismo lenguaje

1. $E \to E{+}E \mid E{*}E \mid (E) \mid x \mid y$ (alfabeto de símbolos terminales $\{x, y, +, *, (,)\}$ y símbolo inicial E).

2. $S \to SS{+} \mid SS{*} \mid x \mid y$ (alfabeto de símbolos terminales $\{x, y, +, *\}$ y símbolo inicial S)

Ejercicio 1.1.15. Una gramática independiente del contexto generalizada es una gramática en el que las producciones son de la forma $A \to r$ donde r es una expresión regular de variables y símbolos terminales. Una gramática independiente del contexto generalizada representa una forma compacta de representar una gramática con todas las producciones $A \to \alpha$, donde α es una palabra del lenguaje asociado a la expresión regular r y $A \to r$ es una producción de la gramática generalizada. Observemos que esta gramática asociada puede tener infinitas producciones, ya que una expresión regular puede representar un lenguaje con infinitas palabras. El concepto de lenguaje generado por una gramática generalizada se define de forma análoga al de las gramáticas independientes del contexto, pero teniendo en cuenta que ahora puede haber infinitas producciones. Demostrar que un lenguaje es independiente del contexto si y solo si se puede generar por una gramática generalizada.

Ejercicio 1.1.16. Demostrar que los siguientes lenguajes son independientes del contexto:

1. $L_1 = \{u\#w \mid u^{-1}$ es una subcadena de $w, u, w \in \{0,1\}^*\}$
2. $L_2 = \{u_1\#u_2\#\ldots\#u_k \mid k \geqslant 1,$ cada $u_i \in \{0,1\}^*,$ y para algún i y $j, u_i = u_j^{-1}\}$

Ejercicio 1.1.17. Sobre el alfabeto $\{0,1\}$ dar una gramática no ambigua que genere todas las palabras en las que el número de 0s es el doble que el de 1s.

Ejercicio 1.1.18. Sea el lenguaje $L = \{0^i1^k0^j \mid i \neq j, 2i \neq j\}$. Demostrar que L es independiente del contexto.

Ejercicio 1.1.19. Demostrar que si una gramática G está en forma normal de Chomsky, entonces si $w \in L(G)$ el número de pasos de derivación de toda generación de esta palabra es $2|w| - 1$.

Ejercicio 1.1.20. Dar gramáticas independientes del contexto no ambiguas para los siguientes lenguajes sobre el alfabeto $\{0,1\}$:

1. El conjunto de palabras w tal que en todo prefijo de w el número de 0s es mayor o igual que el número de 1s.
2. El conjunto de palabras w en las que el número de 0s es mayor o igual que el número de 1s.

Ejercicio 1.1.21. Sea $L = \{0^i1^j0^k \mid i \neq j, 2i \neq j\}$. Demostrar que L es independiente del contexto.

Ejercicio 1.1.22. Supongamos el conjunto de símbolos terminales $T = \{\text{if}, \text{condicion}, \text{then}, \text{else}, a := 1\}$, el alfabeto de variables $V = \{\langle\text{SENT}\rangle, \langle\text{IF - THEN}\rangle, \langle\text{IF - THEN - ELSE}\rangle, \langle\text{ASIG}\rangle\}$, y las producciones:

$$\begin{cases} \langle\text{SENT}\rangle \to \langle\text{ASIG}\rangle \mid \langle\text{IF - THEN}\rangle \mid \langle\text{IF - THEN - ELSE}\rangle \\ \langle\text{IF - THEN}\rangle \to \text{if condicion then}\langle\text{SENT}\rangle \\ \langle\text{IF - THEN - ELSE}\rangle \to \text{if condicion then}\langle\text{SENT}\rangle\text{else}\langle\text{SENT}\rangle \\ \langle\text{ASIG}\rangle \to \text{a := 1} \end{cases}$$

Suponiendo que el símbolo inicial es $\langle\text{SENT}\rangle$, demostrar que la gramática es ambigua. Dar una gramática no ambigua que genere el mismo lenguaje.