

Práctica 07: Diseño y simplificación de gramáticas

Computabilidad y Algoritmia

Cheuk Kelly Ng Pante (alu0101364544@ull.edu.es)

29/10/2024

Índice general

1. Ejercicios de diseño de gramáticas	1
1.1. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^n \mid n \geq 0\}$	1
1.2. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$	4
1.3. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{ww^I \mid w \in \{a, b\}^*\}$	7
1.4. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m c^n \mid n \geq 0, m \text{ impar}\}$	10
1.5. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m c^m d^n \mid n, m \geq 1\}$	13
1.6. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m \mid n > m \geq 0\}$	16
1.7. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^i b^j c^{i+j} \mid i, j \geq 1, i + j \text{ par}\}$	19
1.8. Diseñar una gramática independiente del contexto que genere el lenguaje de las expresiones booleanas con los operadores AND, OR, y NOT, usando paréntesis para agrupar. Ejemplos: “(true AND false)”, “NOT (true OR false)”, “true AND (false OR true)”.	22

1. Ejercicios de diseño de gramáticas

1.1. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^n \mid n \geq 0\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^n b^n \mid n \geq 0\}$ acepta todas las cadenas que tienen el mismo número de símbolos de a's seguidos por el mismo número de b's. La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aSb \mid \varepsilon$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:

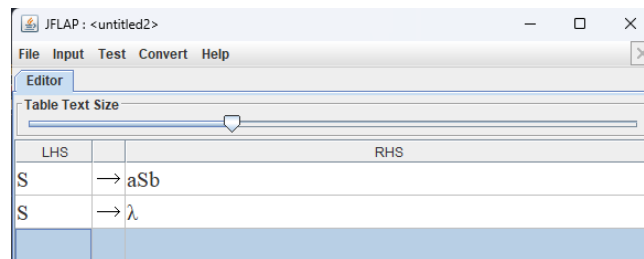


Figura 1.1: Gramática diseñada en JFLAP para el lenguaje $L = \{a^n b^n \mid n \geq 0\}$

- **Ejemplos de cadenas generadas:**

- **Cadena 1: $aabb$**

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $aabb$ se muestra en la siguiente figura:

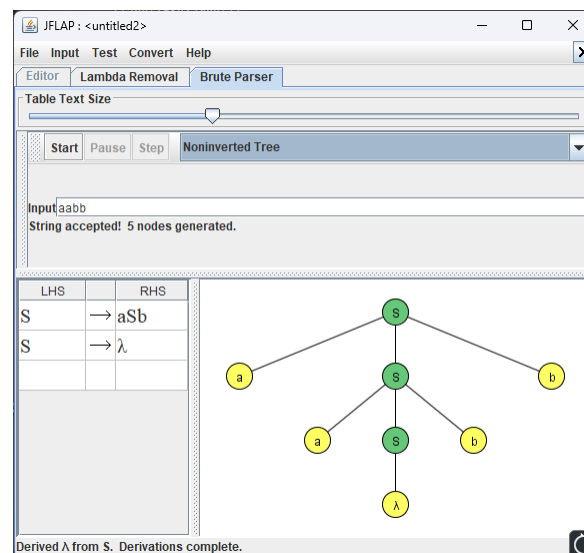


Figura 1.2: Árbol de análisis sintáctico para la cadena $aabb$

- **Cadena 2:** *aaabbb*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aaabbb* se muestra en la siguiente figura:

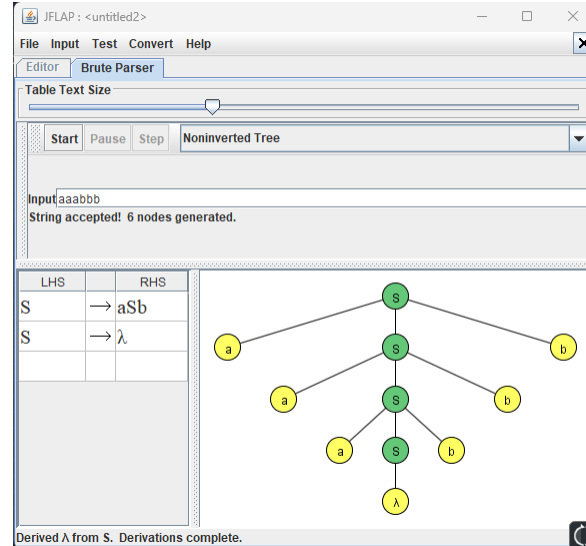


Figura 1.3: Árbol de análisis sintáctico para la cadena *aaabbb*

- **Cadena 3:** *aaaaaabbabbbb*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aaaaaabbabbbb* se muestra en la siguiente figura:

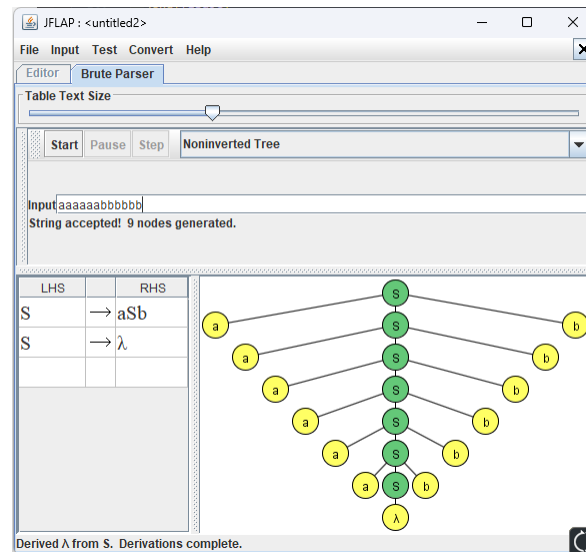


Figura 1.4: Árbol de análisis sintáctico para la cadena *aaaaaabbabbbb*

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:

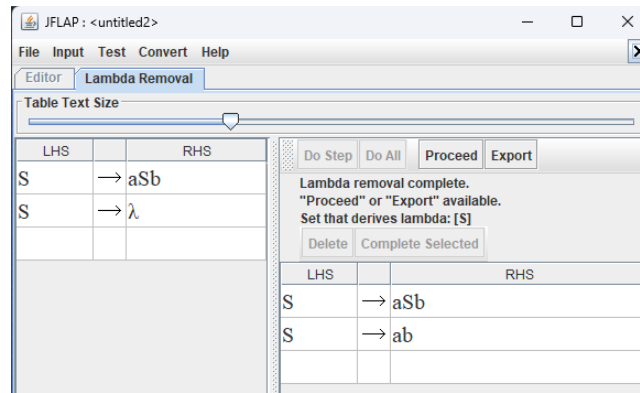


Figura 1.5: Gramática simplificada en JFLAP para el lenguaje $L = \{a^n b^n \mid n \geq 0\}$

1.2. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$ acepta todas las cadenas de a's seguidas de b's donde el número de a's es diferente del número de b's. Entonces se puede dividir en dos casos:

- $n > m$: En este caso, el número de a's es mayor que el número de b's.
- $n < m$: En este caso, el número de a's es menor que el número de b's.

La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aSb \mid A \mid B$
 - $A \rightarrow aA \mid a$
 - $B \rightarrow bB \mid b$
- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:

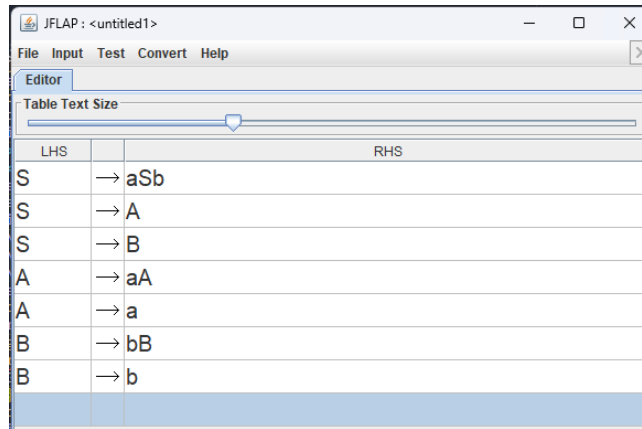


Figura 1.6: Gramática diseñada en JFLAP para el lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$

■ Ejemplos de cadenas generadas:

• Cadena 1: *aab*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aab* se muestra en la siguiente figura:

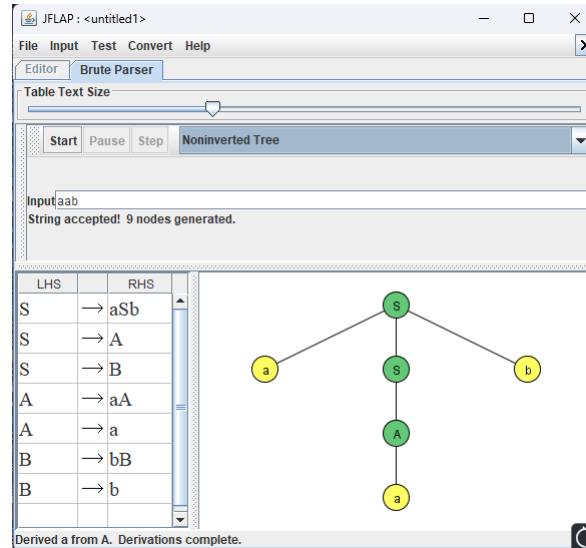


Figura 1.7: Árbol de análisis sintáctico para la cadena *aab*

• Cadena 2: *abbb*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *abbb* se muestra en la siguiente figura:

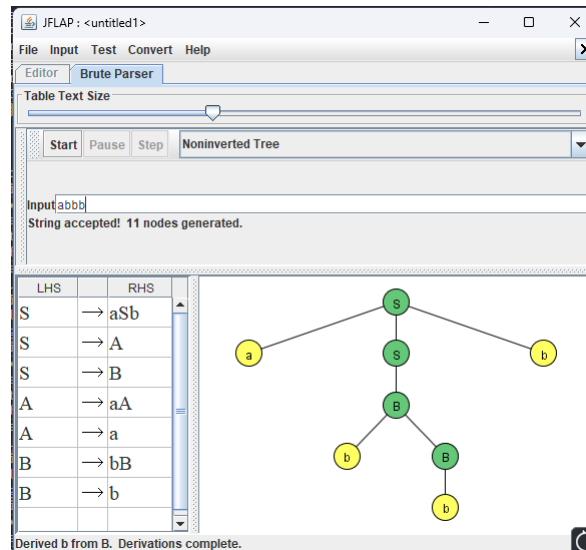


Figura 1.8: Árbol de análisis sintáctico para la cadena *abbb*

- **Cadena 3:** *aaaaaaaaab*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aaaaaaab* se muestra en la siguiente figura:

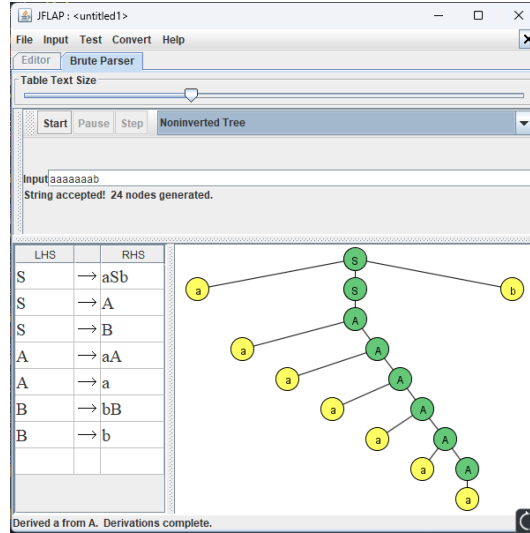
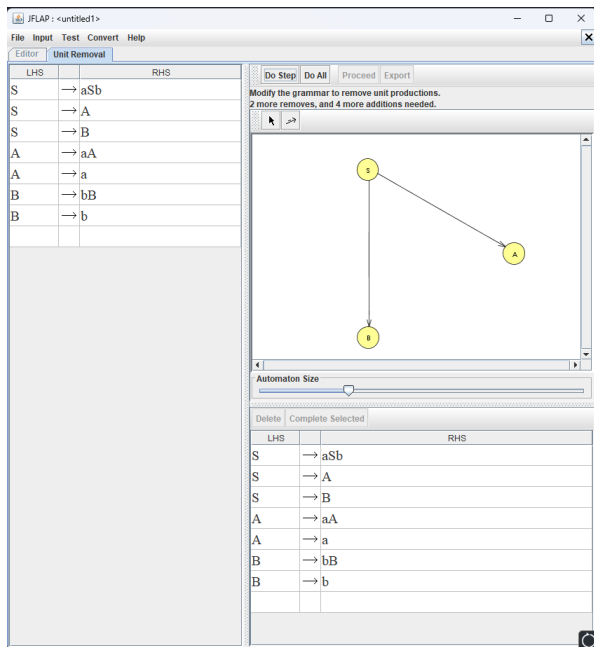
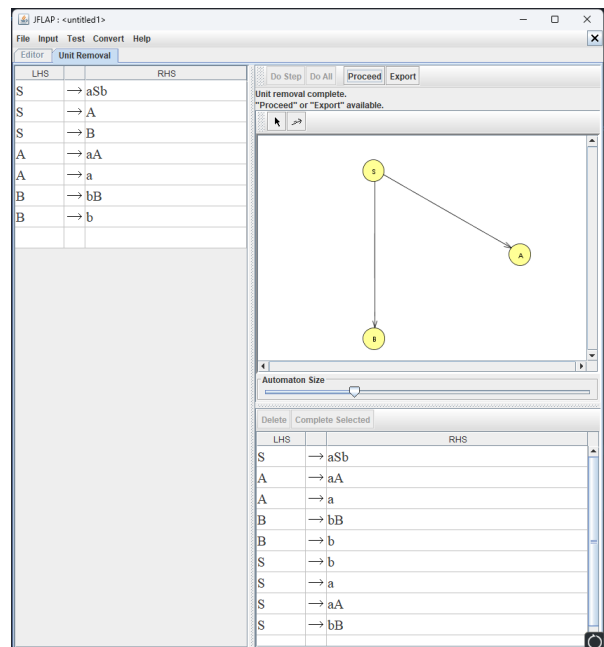


Figura 1.9: Árbol de análisis sintáctico para la cadena *aaaaaaab*

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:



(a) Eliminación de producciones unitarias

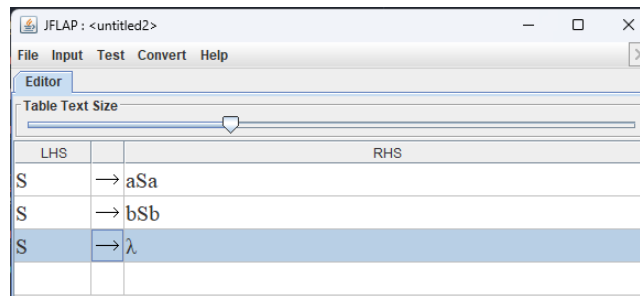


(b) Gramática simplificada

Figura 1.10: Gramática simplificada en JFLAP para el lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$

1.3. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{ww^I \mid w \in \{a,b\}^*\}$

- **Explicación de la gramática:** El lenguaje $L = \{ww^I \mid w \in \{a,b\}^*\}$ acepta todas las cadenas que son palíndromos sobre el alfabeto $\Sigma = \{a,b\}$. La gramática diseñada para el lenguaje L sigue la siguiente estructura:
 - $S \rightarrow aSa \mid bSb \mid \varepsilon$
- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:



The screenshot shows the JFLAP Editor window with the following grammar rules:

LHS	RHS
S	$\rightarrow aSa$
S	$\rightarrow bSb$
S	$\rightarrow \lambda$

Figura 1.11: Gramática diseñada en JFLAP para el lenguaje $L = \{ww^I \mid w \in \{a,b\}^*\}$

- **Ejemplos de cadenas generadas:**
 - **Cadena 1: aa**
 - **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena aa se muestra en la siguiente figura:

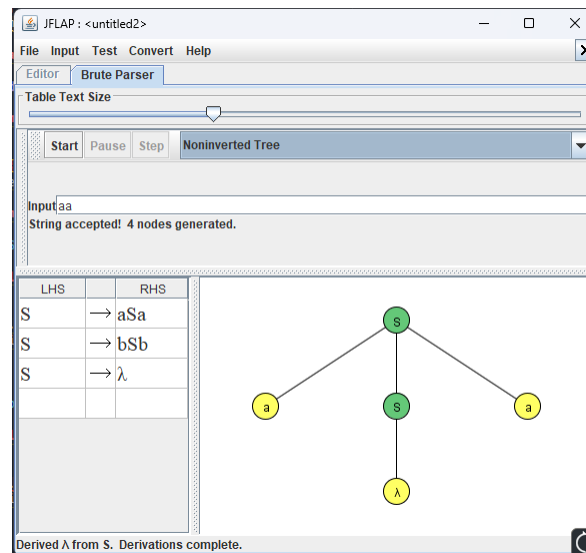


Figura 1.12: Árbol de análisis sintáctico para la cadena aa

- **Cadena 2: *abba***

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *abba* se muestra en la siguiente figura:

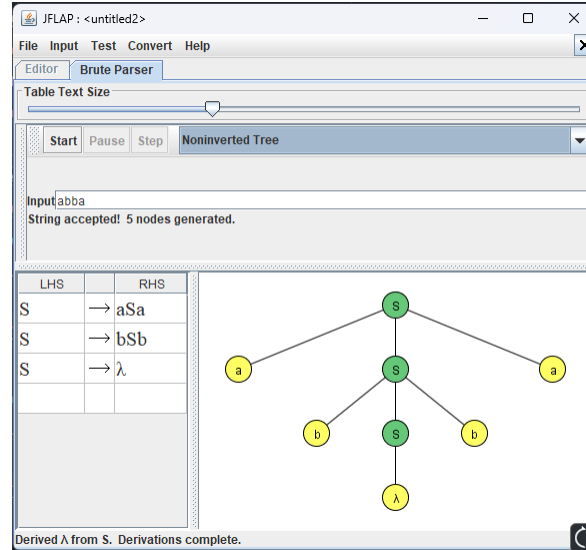


Figura 1.13: Árbol de análisis sintáctico para la cadena *abba*

- **Cadena 3: *abbaabba***

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *abbaabba* se muestra en la siguiente figura:

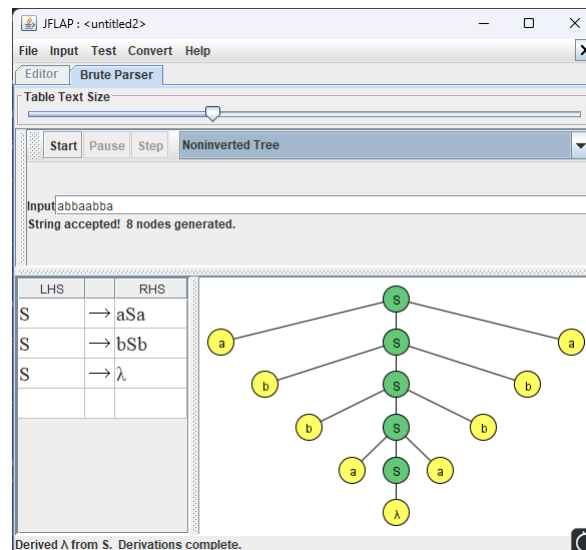


Figura 1.14: Árbol de análisis sintáctico para la cadena *abbaabba*

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:

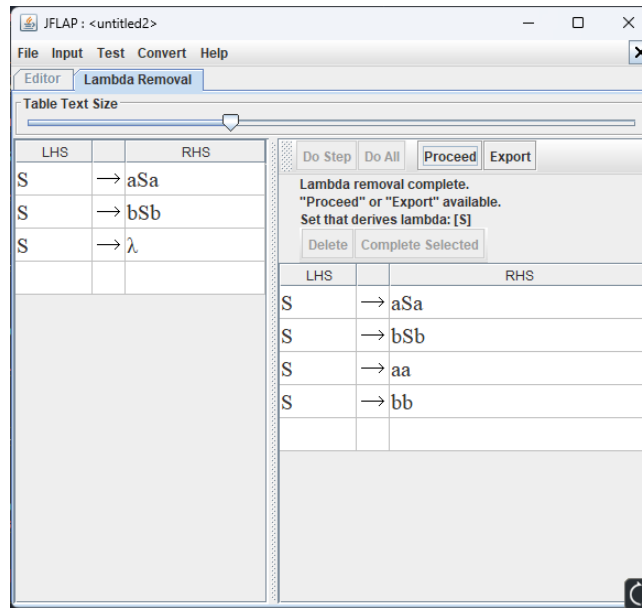


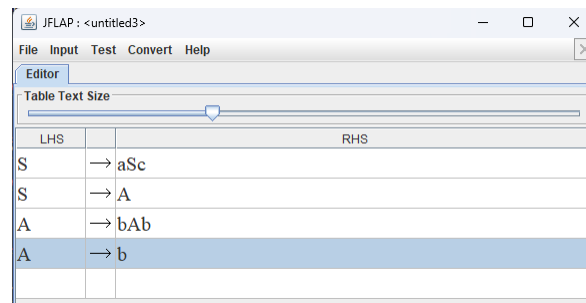
Figura 1.15: Gramática simplificada en JFLAP para el lenguaje $L = \{ww^I \mid w \in \{a,b\}^*\}$

1.4. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m c^n \mid n \geq 0, m \text{ impar}\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^n b^m c^n \mid n \geq 0, m \text{ impar}\}$ acepta todas las cadenas que empiezan con una cantidad n de símbolos a , luego tienen una cantidad m de símbolos b , y terminan con una cantidad n de símbolos c . La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aSc \mid A$
- $A \rightarrow bAb \mid b$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:



LHS	RHS
S	→ aSc
S	→ A
A	→ bAb
A	→ b

Figura 1.16: Gramática diseñada en JFLAP para el lenguaje $L = \{a^n b^m c^n \mid n \geq 0, m \text{ impar}\}$

- **Ejemplos de cadenas generadas:**

- **Cadena 1: b**

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena b se muestra en la siguiente figura:

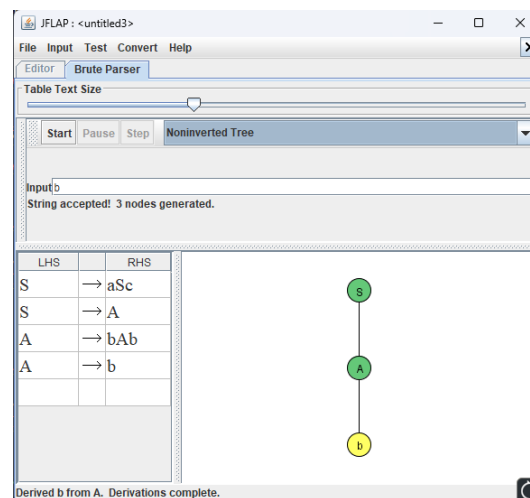


Figura 1.17: Árbol de análisis sintáctico para la cadena b

- **Cadena 2: abc**
 - **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena abc se muestra en la siguiente figura:

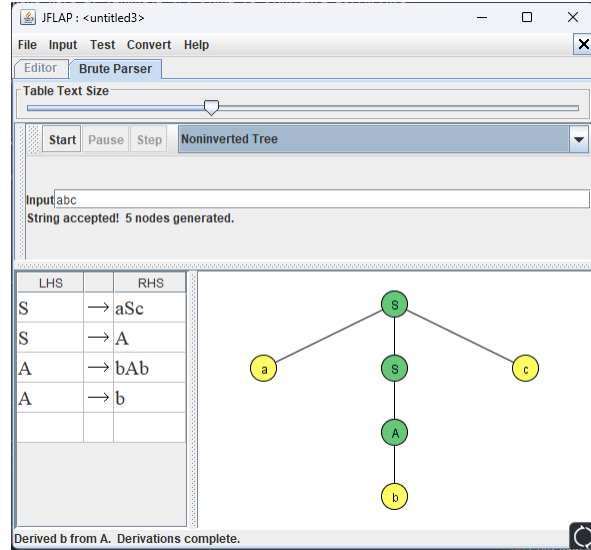


Figura 1.18: Árbol de análisis sintáctico para la cadena abc

- **Cadena 3: $aaaabbbcccc$**
 - **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $aaaabbbcccc$ se muestra en la siguiente figura:

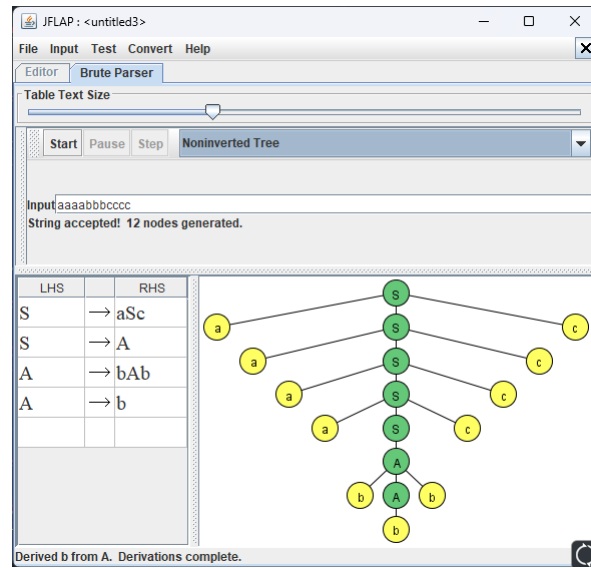
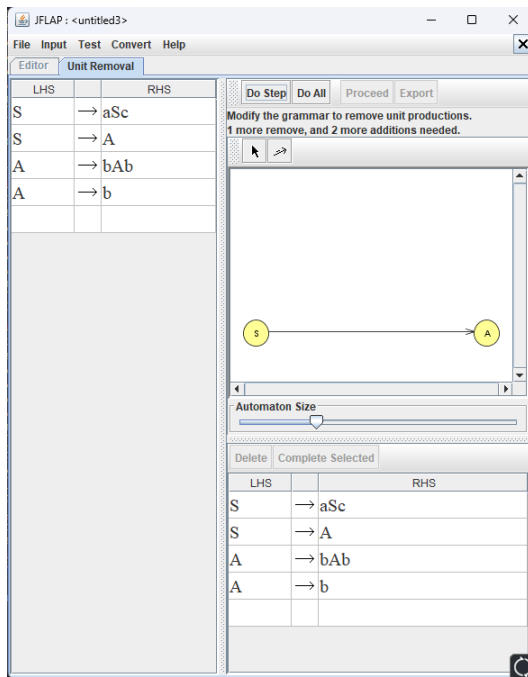
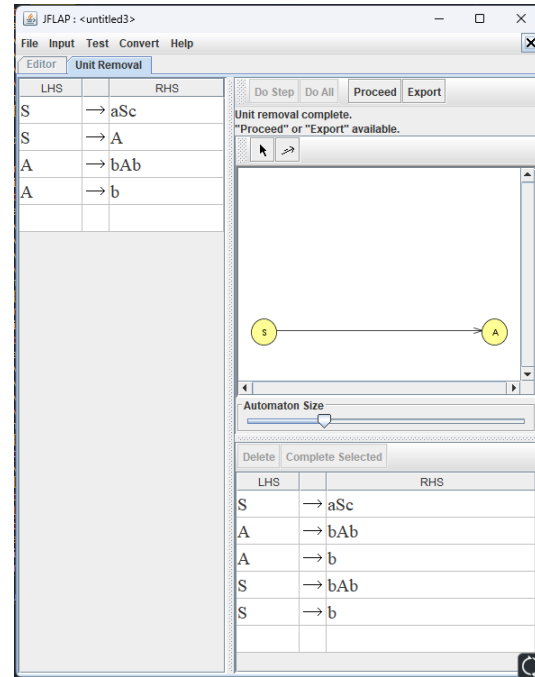


Figura 1.19: Árbol de análisis sintáctico para la cadena $aaaabbbcccc$

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:



(a) Eliminación de producciones unitarias



(b) Gramática simplificada

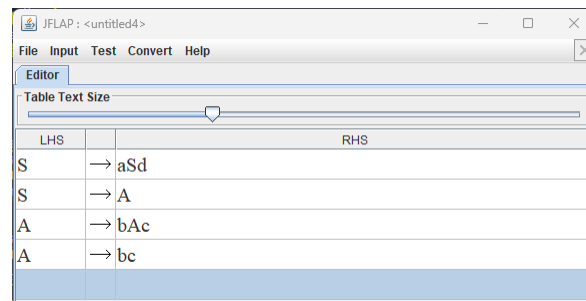
Figura 1.20: Gramática simplificada en JFLAP para el lenguaje $L = \{a^n b^m \mid n, m \geq 0, n \neq m\}$

1.5. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m c^m d^n \mid n, m \geq 1\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^n b^m c^m d^n \mid n, m \geq 1\}$ acepta todas las cadenas que empiezan con una cantidad n de símbolos a , seguido de una cantidad m de símbolos b , luego una cantidad m de símbolos c , y terminan con una cantidad n de símbolos d . La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aSd \mid A$
- $A \rightarrow bAc \mid bc$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:



LHS	RHS
S	→ aSd
S	→ A
A	→ bAc
A	→ bc

Figura 1.21: Gramática diseñada en JFLAP para el lenguaje $L = \{a^n b^m c^m d^n \mid n, m \geq 1\}$

- **Ejemplos de cadenas generadas:**

- **Cadena 1:** $abcd$

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $abcd$ se muestra en la siguiente figura:

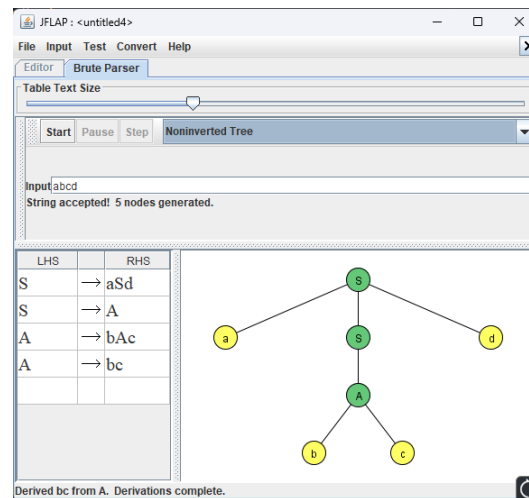


Figura 1.22: Árbol de análisis sintáctico para la cadena $abcd$

- **Cadena 2:** *aabbccdd*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aabbccdd* se muestra en la siguiente figura:

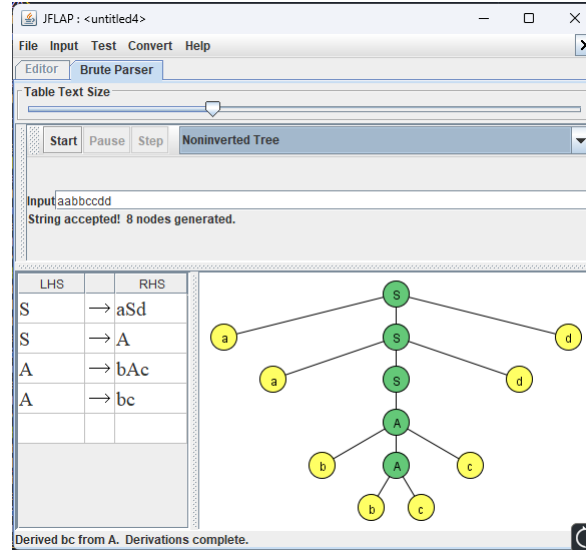


Figura 1.23: Árbol de análisis sintáctico para la cadena *aabbccdd*

- **Cadena 3:** *abbbcccd*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *abbbcccd* se muestra en la siguiente figura:

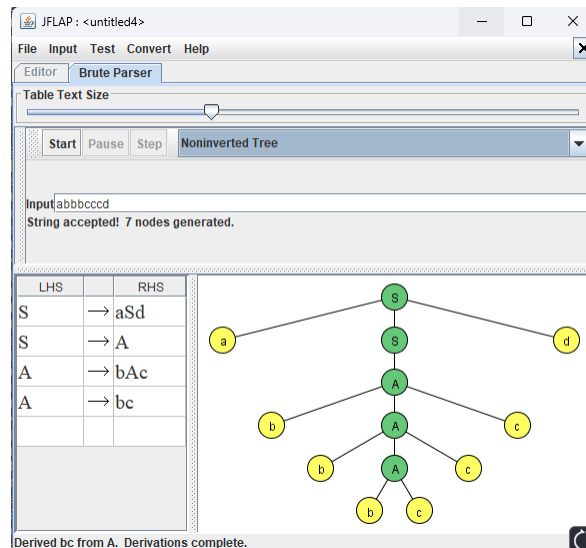
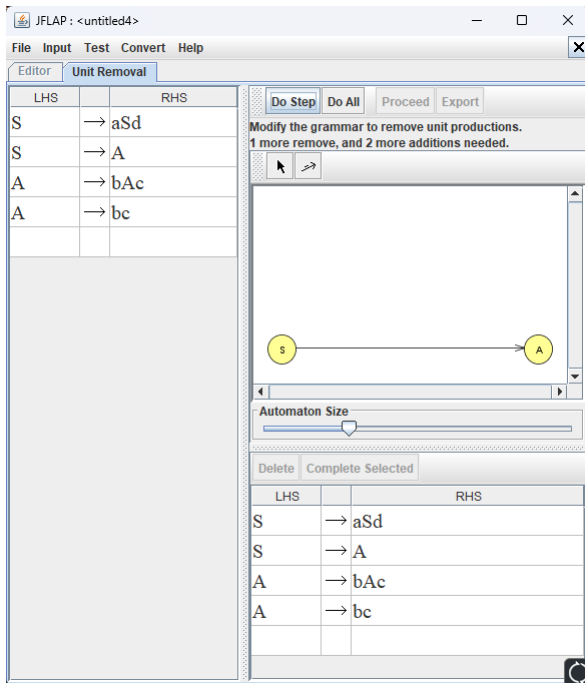
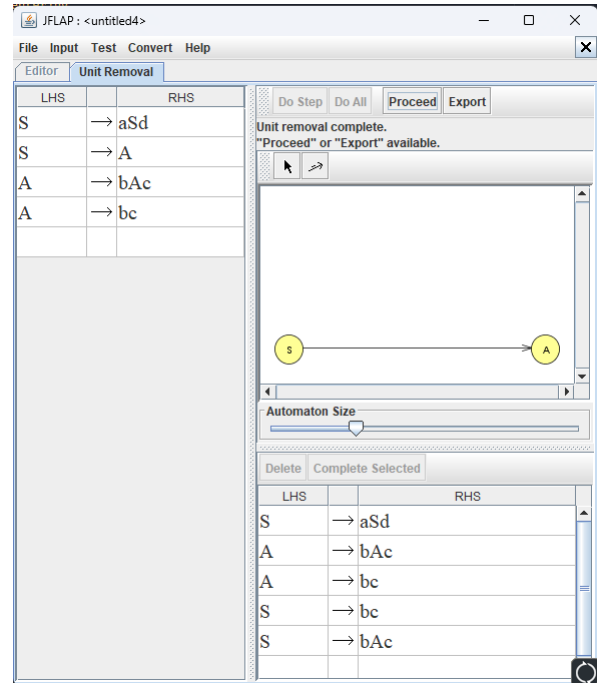


Figura 1.24: Árbol de análisis sintáctico para la cadena *abbbcccd*

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:



(a) Eliminación de producciones unitarias



(b) Gramática simplificada

Figura 1.25: Gramática simplificada en JFLAP para el lenguaje $L = \{a^n b^m c^m d^n \mid n, m \geq 1\}$

1.6. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^n b^m \mid n > m \geq 0\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^n b^m \mid n > m \geq 0\}$ acepta todas las cadenas que tienen más símbolos de a's que de b's. La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aS \mid aSb \mid \varepsilon$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:

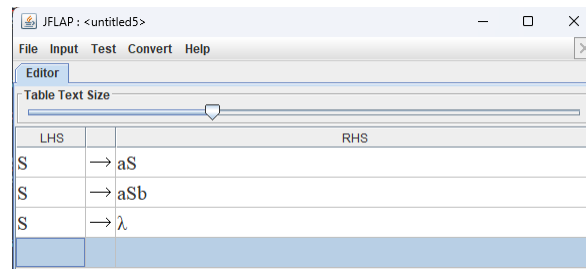


Figura 1.26: Gramática diseñada en JFLAP para el lenguaje $L = \{a^n b^m \mid n > m \geq 0\}$

- **Ejemplos de cadenas generadas:**

- **Cadena 1: aab**

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena aab se muestra en la siguiente figura:

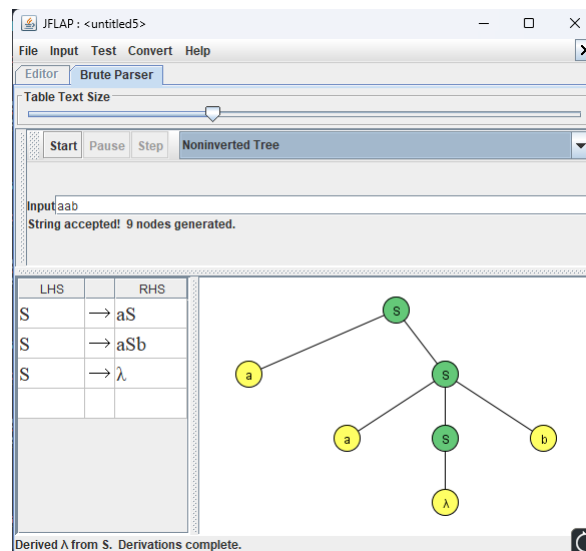


Figura 1.27: Árbol de análisis sintáctico para la cadena aab

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:

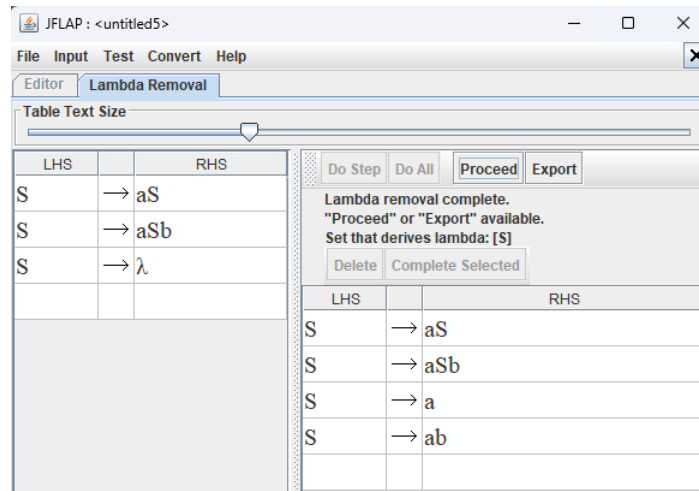


Figura 1.30: Gramática simplificada en JFLAP para el lenguaje $L = \{a^n b^m \mid n > m \geq 0\}$

1.7. Diseñar una gramática independiente del contexto que genere el lenguaje $L = \{a^i b^j c^{i+j} \mid i, j \geq 1, i + j \text{ par}\}$

- **Explicación de la gramática:** El lenguaje $L = \{a^i b^j c^{i+j} \mid i, j \geq 1, i + j \text{ par}\}$ acepta todas las cadenas que tienen una cantidad i de a's, seguido de una cantidad j de b's, y terminan con una cantidad $i + j$ de c's, donde la suma de i y j es un número par. La gramática diseñada para el lenguaje L sigue la siguiente estructura:

- $S \rightarrow aSc \mid A$
- $A \rightarrow bAc \mid bc$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:

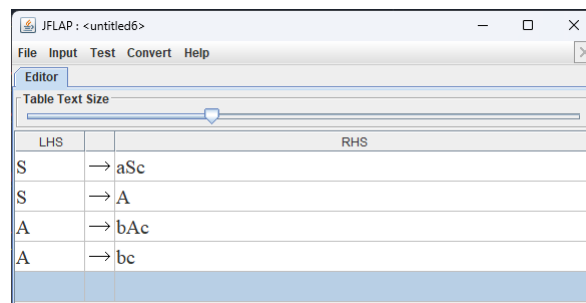


Figura 1.31: Gramática diseñada en JFLAP para el lenguaje $L = \{a^i b^j c^{i+j} \mid i, j \geq 1, i + j \text{ par}\}$

- **Ejemplos de cadenas generadas:**

- **Cadena 1:** $abcc$

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $abcc$ se muestra en la siguiente figura:

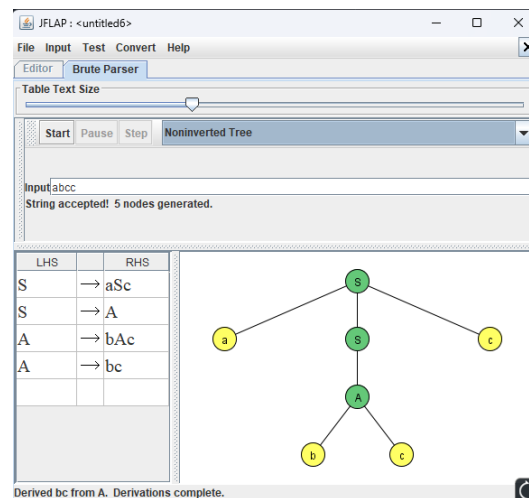


Figura 1.32: Árbol de análisis sintáctico para la cadena $abcc$

- **Cadena 2:** *aabbcccc*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *aabbcccc* se muestra en la siguiente figura:

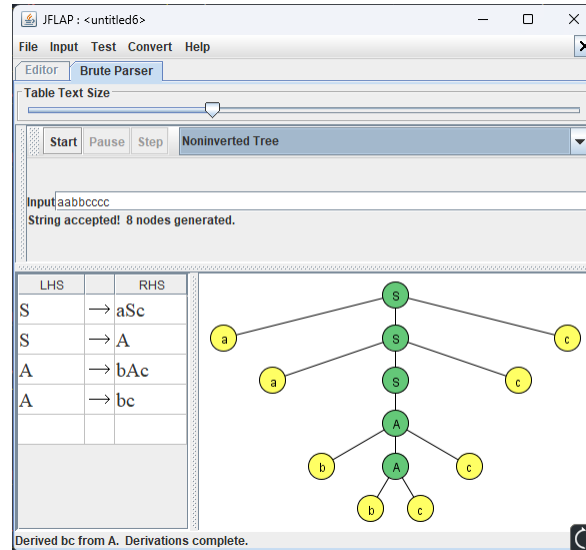


Figura 1.33: Árbol de análisis sintáctico para la cadena *aabbcccc*

- **Cadena 3:** *abbbcccc*

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena *abbbcccc* se muestra en la siguiente figura:

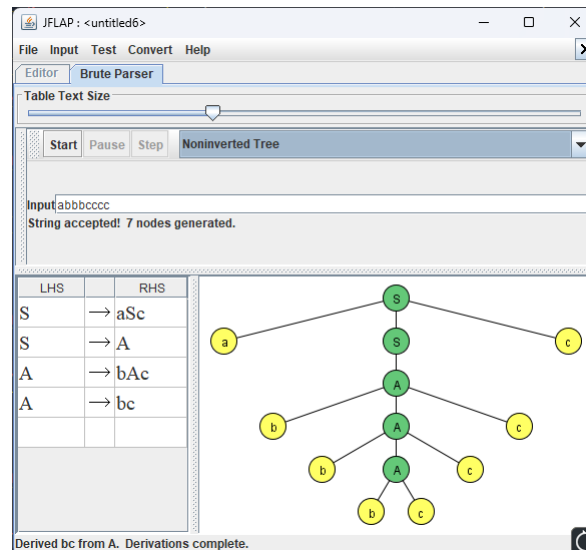


Figura 1.34: Árbol de análisis sintáctico para la cadena *abbbcccc*

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:

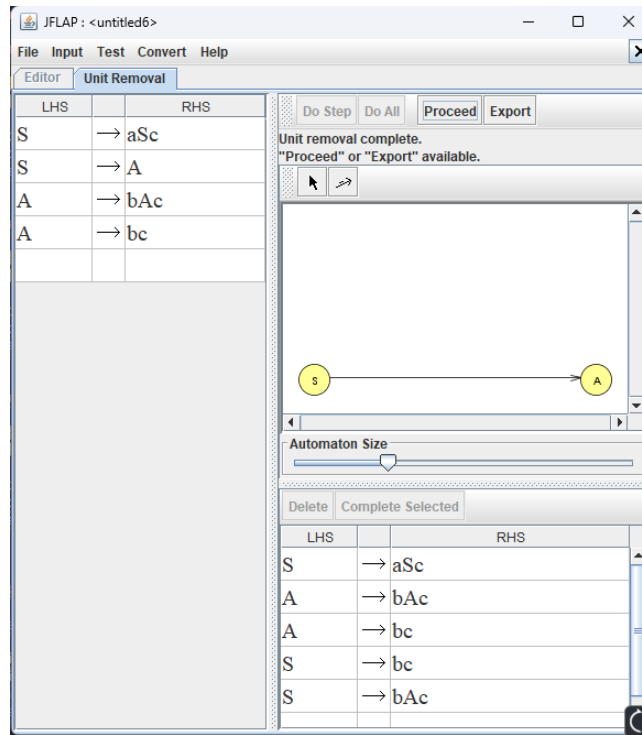


Figura 1.35: Gramática simplificada en JFLAP para el lenguaje $L = \{a^i b^j c^{i+j} \mid i, j \geq 1, i + j \text{ par}\}$

1.8. Diseñar una gramática independiente del contexto que genere el lenguaje de las expresiones booleanas con los operadores AND, OR, y NOT, usando paréntesis para agrupar. Ejemplos: “(true AND false)”, “NOT (true OR false)”, “true AND (false OR true)”.

- **Explicación de la gramática:** El lenguaje de las expresiones booleanas con los operadores AND, OR, y NOT, usando paréntesis para agrupar, acepta todas las cadenas que siguen la siguiente estructura:

$$S \rightarrow true \mid false \mid NOT\ S \mid S\ AND\ S \mid S\ OR\ S \mid (S)$$

- **Imagen de la gramática en JFLAP:** La gramática diseñada en JFLAP se muestra en la siguiente figura:

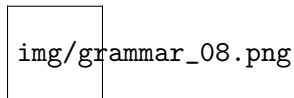


Figura 1.36: Gramática diseñada en JFLAP para el lenguaje de las expresiones booleanas con los operadores AND, OR, y NOT, usando paréntesis para agrupar

- **Ejemplos de cadenas generadas:**

- **Cadena 1:** $(trueANDfalse)$

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $(trueANDfalse)$ se muestra en la siguiente figura:

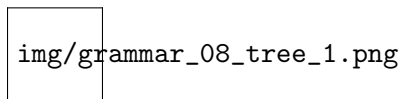


Figura 1.37: Árbol de análisis sintáctico para la cadena $(trueANDfalse)$

- **Cadena 2:** $NOT(trueORfalse)$

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $NOT(trueORfalse)$ se muestra en la siguiente figura:

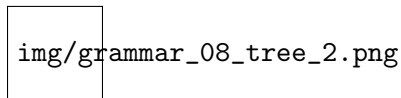


Figura 1.38: Árbol de análisis sintáctico para la cadena $NOT(trueORfalse)$

- **Cadena 3:** $trueAND(falseORtrue)$

- **Árbol de análisis sintáctico:** El árbol de análisis sintáctico para la cadena $trueAND(falseORtrue)$ se muestra en la siguiente figura:

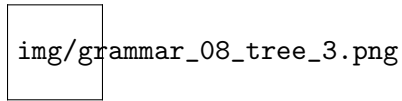


Figura 1.39: Árbol de análisis sintáctico para la cadena $trueAND(falseORtrue)$

- **Imagen de la gramática simplificada en JFLAP:** La gramática simplificada en JFLAP se muestra en la siguiente figura:

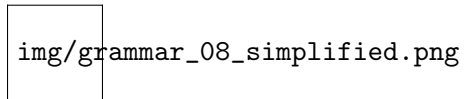


Figura 1.40: Gramática simplificada en JFLAP para el lenguaje de las expresiones booleanas con los operadores AND, OR, y NOT, usando paréntesis para agrupar