



Instituto Politecnico Nacional



ESCOM “ESCUELA SUPERIOR DE CÓMPUTO”

TEORÍA COMPUTACIONAL

PRÁCTICA 8: TRANSFORMACIÓN GLC

PROFA: Luz María Sánchez García

ALUMMNO: Rojas Alvarado Luis Enrique

GRUPO: 2CM11

INTRODUCCIÓN

El propósito de la práctica consistió en que, en base a unos ejercicios propuestos por la maestra, y por medio de la aplicación de JFLAP fuéramos limpiando paso a paso dicha gramática y probáramos algunas cadenas para esa gramática así cómo describir el lenguaje que genera cada una de las gramáticas, para posteriormente convertir la gramática a su forma normal de Chomsky (FNC).

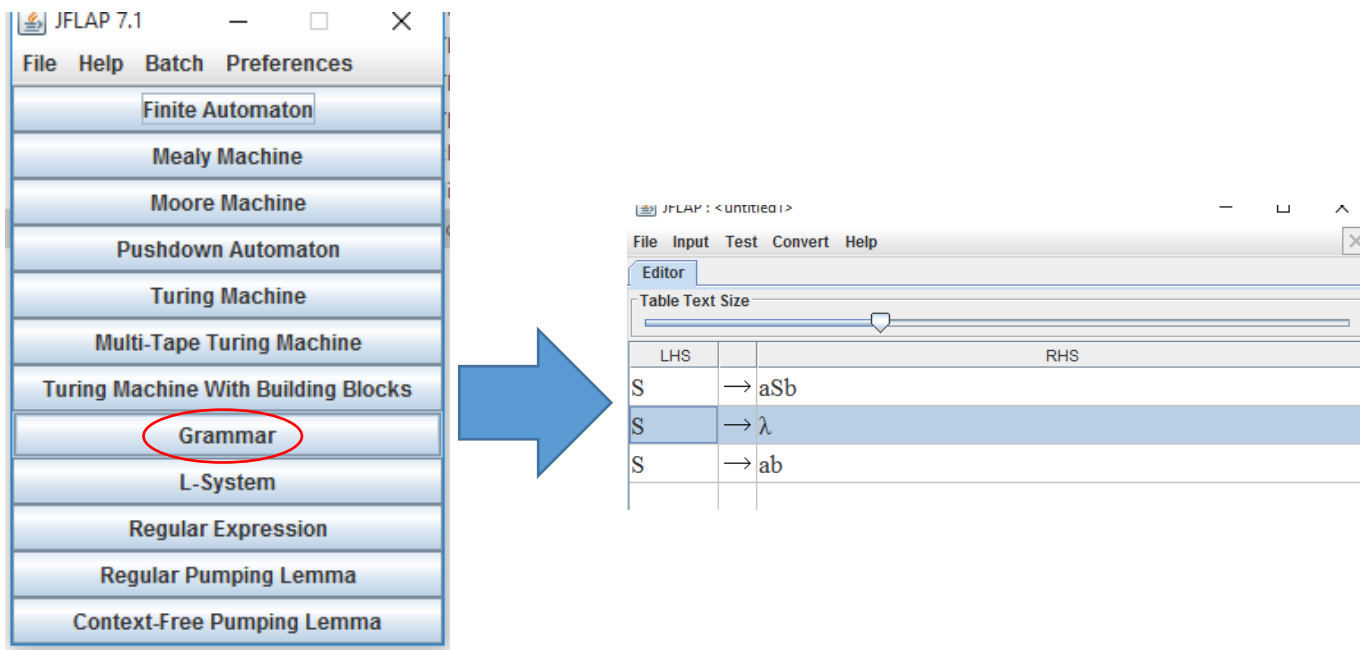
PLANTEAMIENTO DEL PROBLEMA

El problema que se presenta es que por medio de una gramática dada se tienen que limpiar y posteriormente convertir la gramática a su forma normal de Chomsky y probar cadenas que genera las derivaciones de las reglas de la gramática en su forma limpia y en su FNC.

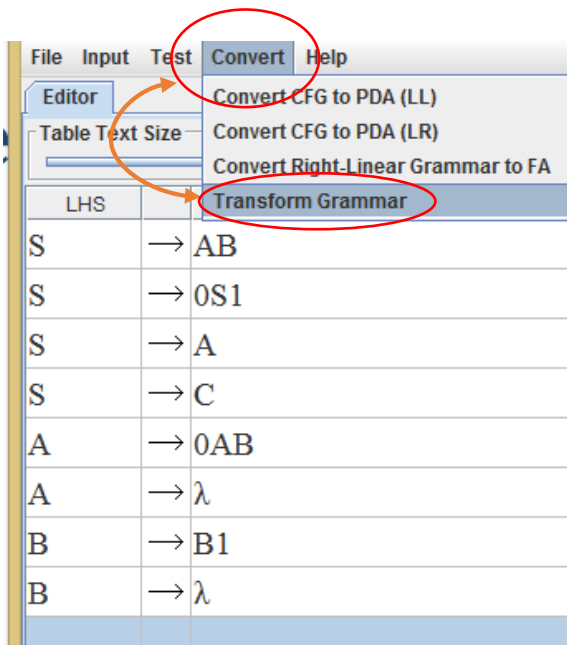
Para darle solución se optó por usar la herramienta de JFLAP ya que ofrece la opción para generar la gramática y paso a paso te dará la solución hasta que la gramática quede limpia y posteriormente se convierta a la forma normal de Chomsky.

DISEÑO DE LA SOLUCIÓN

Como ya se comentó anteriormente se utilizó la herramienta JFLAP en la interfaz de gramática y posteriormente se hace el diseño de las reglas de la gramática.

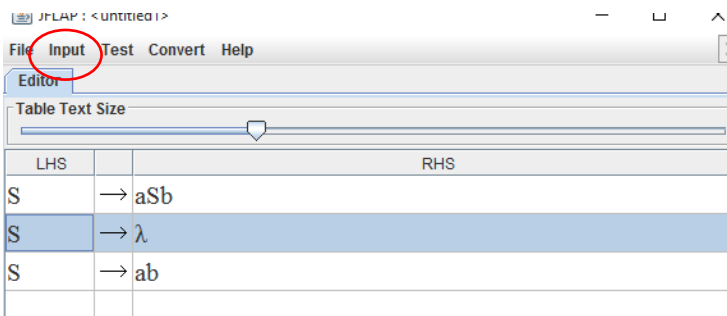


Una vez ingresada la gramática tenemos que ir a la pestaña de convert y posteriormente seleccionar la opción de convert to grammar para que empiece a limpiar la gramática.

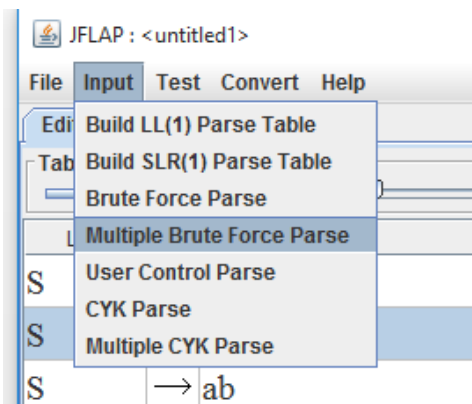


Una vez ahí, se selecciona la opción de do step para ir viendo paso a paso cómo se limpia la gramática ya que en la pestaña de arriba se dice que paso es el que estáoms realizando.

Posteriormente se selecciona la opción de export junto Do step para que podamos probar las cadenas pueden ser validas o no válidas. Para eso tenemos que ir a la parte de input.

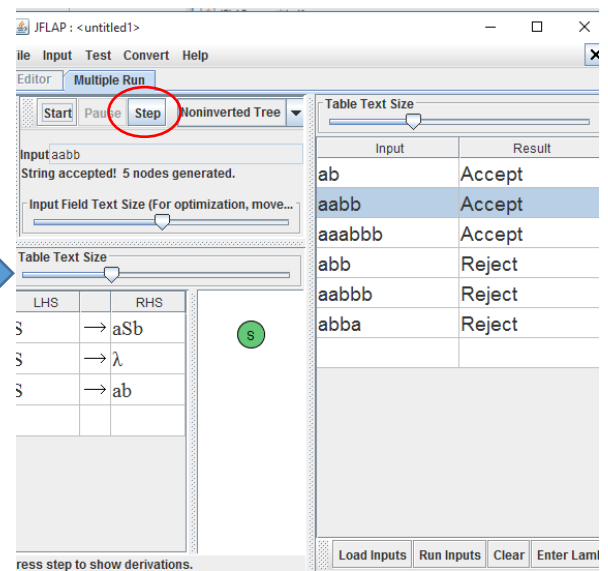
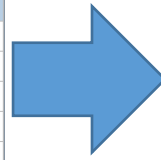
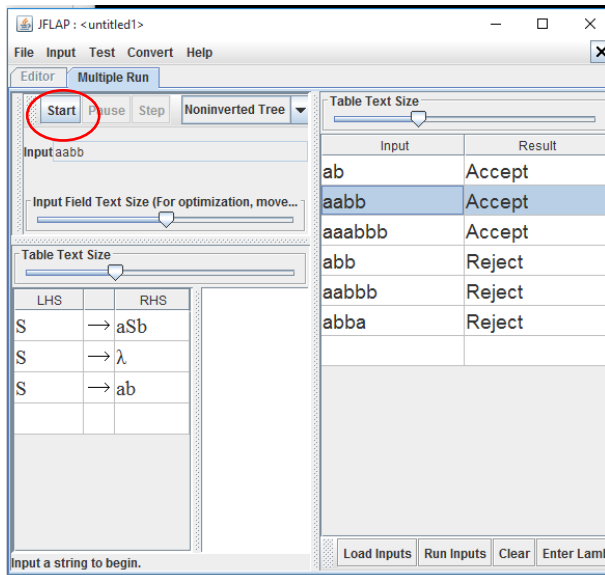


Una vez ahí se selecciona la opción de Multiple Brute Force Parse para agregar múltiples cadenas y verificar si son válidas o no válidas para la gramática declarada.

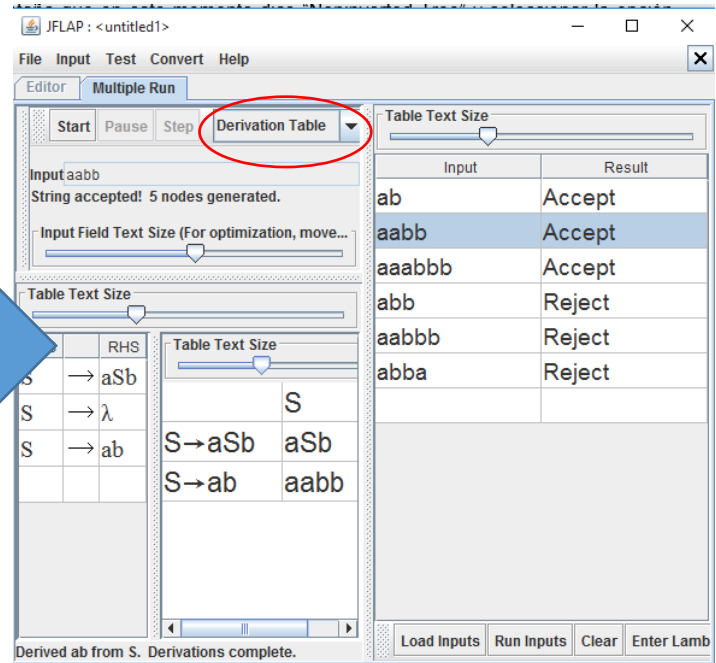
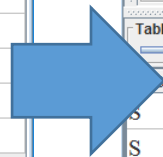
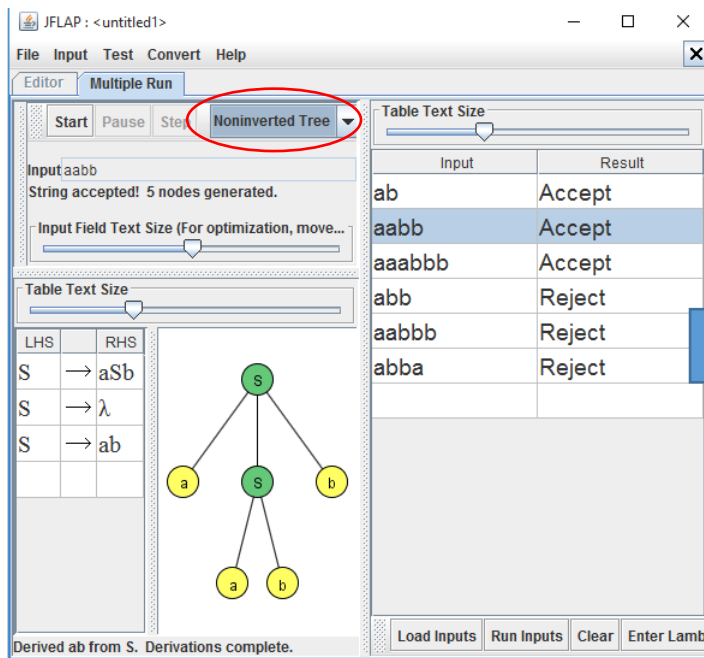


La pestaña generará una nueva ventana dentro de la aplicación para que se puedan ingresar las cadenas deseadas.

Para generar el árbol se selecciona la opción de start en la parte superior de la gramática y después seleccionar step hasta que el árbol esté completamente formado.



Una vez generado el árbol, para generar la tabla de derivación se requiere desplegar la pestaña que en este momento dice “Noninverted Tree” y seleccionar la opción “Derivation Table”



De ésta manera obtendremos tanto las cadenas válidas y no válidas así como el árbol de derivación y la tabla de derivación. Para así poder sacar una conclusión sobre el lenguaje que maneja la gramática, que tipo de expresión es y si es ambigua o no.

IMPLEMENTACIÓN DE LA SOLUCIÓN

1. $G = (\{0,1\}, \{S,A,B,C\}, S, P)$
 - $S \rightarrow AB \mid 0S1 \mid A \mid C$
 - $A \rightarrow 0AB \mid \lambda$
 - $B \rightarrow B1 \mid \lambda$
2. $G = (\{i,+\}, \{Z,E,F,G,P,Q,S,T\}, Z, P)$
 - $Z \rightarrow E + T$
 - $E \rightarrow E \mid S + F \mid T$
 - $F \rightarrow F \mid FP \mid P$
 - $P \rightarrow G$
 - $G \rightarrow G \mid GG \mid F$
 - $T \rightarrow T * i \mid i$
 - $Q \rightarrow E \mid E + F \mid T \mid S$
 - $S \rightarrow i$
3. $S \rightarrow Aab \mid B \mid CSa \mid b$
 - $A \rightarrow aA \mid Cb \mid a \mid aBAE$
 - $B \rightarrow bB \mid aBC \mid F \mid \lambda$
 - $C \rightarrow CG \mid DC$
 - $D \rightarrow aCb \mid a$
 - $E \rightarrow aaE \mid bB$
 - $f \rightarrow aF \mid ab$
 - $G \rightarrow F$
4. $S \rightarrow aSb \mid aAb$
 - $A \rightarrow BA \mid \lambda \mid aCB$
 - $B \rightarrow ASb \mid abB$
 - $C \rightarrow aDF \mid aDb$
 - $D \rightarrow abC \mid aCB \mid aF$
 - $F \rightarrow Fb \mid aCb$
5. $S \rightarrow A \mid AA \mid AAA$
 - $A \rightarrow ABa \mid ACa \mid a$
 - $B \rightarrow ABa \mid Ab \mid \lambda$
 - $C \rightarrow Cab \mid CC$
 - $D \rightarrow CD \mid Cd \mid CEa$
 - $E \rightarrow b$
6. $S \rightarrow aAb \mid cHB \mid CH$
 - $A \rightarrow dBH \mid eeC$
 - $B \rightarrow ff \mid D$

C->gFB | ah

D->i

E->jF

F->deGGG | eF

G->kF

H->Hlm

7. S->aAB | bBA | ABb | SS

A->aAb | CCA | BB

B-> λ | bC

C->aCS | SCS

D->ab | SABC

8. S->CBa | D

A->bbC

B->Sc | ddd

C->eA | f | C

D->E | SABC

E->gh

FUNCIONAMIENTO

1. $G = (\{0,1\}, \{S, A, B, C\}, S, P)$

-S->AB | 0S1 | A | C

-A->0AB | λ

-B->B1 | λ

Table Text Size	
LHS	RHS
S	→ AB
S	→ 0S1
S	→ A
S	→ C
A	→ 0AB
A	→ λ
B	→ B1
B	→ λ

Do Step | Do All | Proceed | Export

Lambda removal complete.
"Proceed" or "Export" available.
Set that derives lambda: {A, B, S}

Delete | Complete Selected

LHS	RHS
S	→ AB
S	→ 0S1
S	→ A
S	→ C
A	→ 0AB
B	→ B1
S	→ B
S	→ 01
A	→ 0
A	→ 0A
A	→ 0B
B	→ 1

Remover lamda

Start | Pause | Stop | Noninverted Tree

Input: 0011
String accepted: 94 nodes generated.

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing the text field)

Table Text Size

Input	Result
01	Accept
0011	Accept
000111	Accept
00001111	Accept

Table Text Size

LHS	RHS
S	→ AB
S	→ CF
F	→ SD
S	→ CD
S	→ CB
S	→ CE
S	→ CA
S	→ 1
S	→ 0
S	→ BD
A	→ CE
E	→ AB
A	→ 0
A	→ CA
A	→ CB
C	→ 0
B	→ 1
B	→ BD
D	→ 1

Derived 1 from D. Derivations complete.

Load Inputs | Run Inputs | Clear | Enter Lambda

Forma normal de Chomsky

$$L = \{0^n 1^m \mid n, m = 0, 1, 2, 3, \dots\}$$

$$2. G = (\{i, +\}, \{Z, E, F, G, P, Q, S, T\}, Z, P)$$

$$-Z \rightarrow E + T$$

$$-E \rightarrow E \mid S + F \mid T$$

$$-F \rightarrow F \mid FP \mid P$$

$$-P \rightarrow G$$

$$-G \rightarrow G \mid GG \mid F$$

$$-T \rightarrow T * i \mid i$$

$$-Q \rightarrow E \mid E + F \mid T \mid S$$

$$S \rightarrow i$$

LHS	RHS
Z	→ E+T
E	→ E
E	→ S+F
E	→ T
F	→ F
F	→ FP
F	→ P
P	→ G
G	→ G
G	→ GG
G	→ F
T	→ T*i
T	→ i
Q	→ E
Q	→ E+F
Q	→ T
Q	→ S
S	→ i

Unit removal complete.
"Proceed" or "Export" available.

Automation Size

LHS	RHS
Z	→ E+T
E	→ S+F
F	→ FP
G	→ GG
T	→ T*i
T	→ i
Q	→ E+F
S	→ i
Q	→ T*i
F	→ GG
E	→ i
G	→ FP
P	→ GG
E	→ T*i
Q	→ i
P	→ FP
Q	→ S+F

Elimina unitarios

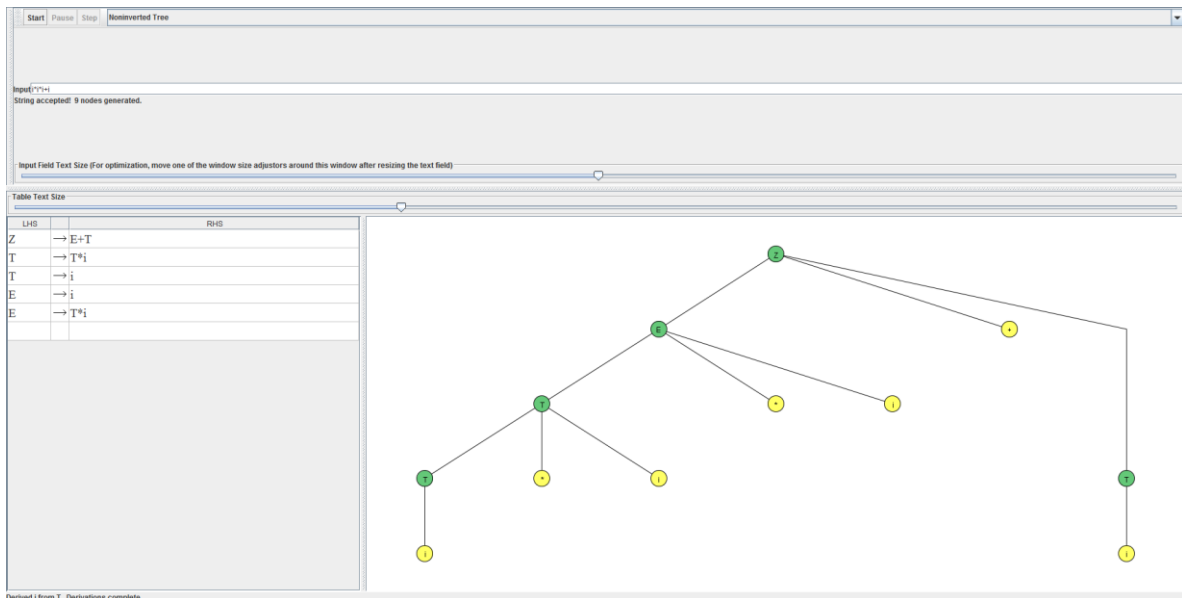
LHS	RHS
Z	→ E+T
E	→ T*i
E	→ i
F	→ GG
F	→ FP
G	→ GG
Q	→ E+F
G	→ FP
P	→ GG
P	→ FP
Q	→ S+F
Q	→ i
Q	→ T*i
S	→ i
E	→ S+F
T	→ i
T	→ T*i

Unitless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: {E, Q, S, T, Z}

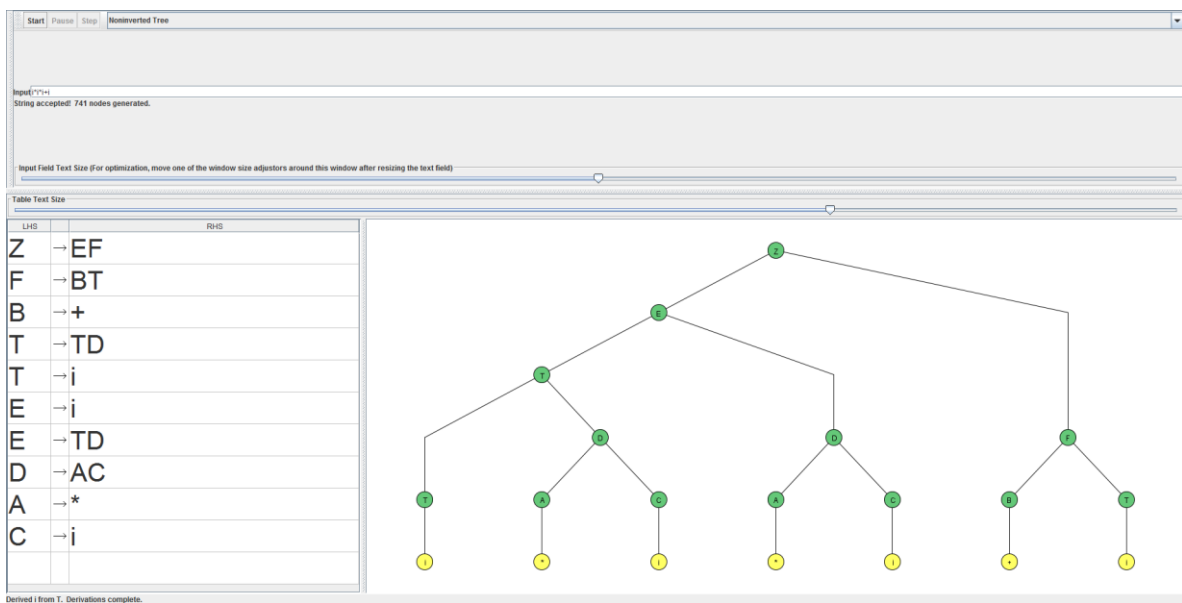
Automation Size

LHS	RHS
Z	→ E+T
E	→ T*i
E	→ i
T	→ i
T	→ T*i

Elimina los inaccesibles



Gramática Limpia



FNC

$$L = \{i*i, i+i, i+*i, \dots\}$$

3. $S \rightarrow Aab \mid B \mid CSa \mid b$
 $A \rightarrow aA \mid Cb \mid a \mid aBAE$
 $B \rightarrow bB \mid aBC \mid F \mid \lambda$
 $C \rightarrow CG \mid DC$
 $D \rightarrow aCb \mid a$
 $E \rightarrow aaE \mid bB$

f->aF | ab
G->F

LHS	RHS
S	→ Aab
S	→ B
S	→ CSa
S	→ b
A	→ aA
A	→ Cb
A	→ a
A	→ aBC
A	→ F
C	→ CG
C	→ DC
D	→ aCb
D	→ a
E	→ aaE
E	→ bB
F	→ aF
F	→ ab
G	→ F
S	→ ab

Do Step Do All Proceed Export

Unit removal complete.
"Proceed" or "Export" available.

Automaton Size

LHS	RHS
S	→ Aab
S	→ CSa
S	→ b
A	→ aA
A	→ Cb
A	→ a
A	→ aBC
C	→ CG
C	→ DC
D	→ aCb
D	→ a
E	→ aaE
E	→ bB
F	→ aF
F	→ ab

Elimina unitarios

LHS	RHS
S	→ Aab
S	→ CSa
S	→ b
S	→ ab
G	→ ab
A	→ aF
G	→ aF
A	→ ab
C	→ DC
C	→ CG
F	→ ab
F	→ aF
E	→ bB
E	→ aaE
D	→ a
A	→ aBC
A	→ a
A	→ Cb
D	→ aCb
A	→ aA

Do Step Do All Proceed Export

Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, b, F, G, S]

Automaton Size

LHS	RHS
S	→ Aab
S	→ b
S	→ ab
A	→ aF
A	→ ab
F	→ ab
F	→ aF
A	→ a
A	→ aA

Elimina inaccesibles

Start Pause Step Noninverted Tree

Input aab
String accepted! 5 nodes generated.

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...

Table Text Size

LHS	RHS
S	→ Aab
S	→ b
S	→ ab
A	→ aA
A	→ a
F	→ aF
F	→ ab
A	→ ab
A	→ aF

Table Text Size

Input	Result
aab	Accept
aaab	Accept
ab	Accept

Load Inputs Run Inputs Clear Enter Lambda

Gramática Limpia

Start Pause Step Noninverted Tree

Input aaab
String accepted! 49 nodes generated.

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...

Table Text Size

LHS	RHS
S	→ AD
D	→ BC
S	→ b
S	→ BC
A	→ BA
A	→ a
F	→ BF
F	→ BC
A	→ BC
C	→ b
A	→ BF
B	→ a

Table Text Size

Input	Result
aab	Accept
aaab	Accept
ab	Accept

Load Inputs Run Inputs Clear Enter Lambda

FNC

$$L = \{a^n b \mid n=0,1,2,\dots\}$$

4. $S \rightarrow aSb \mid aAb$
- $A \rightarrow BA \mid \lambda \mid aCB$
- $B \rightarrow ASb \mid abB$
- $C \rightarrow aDF \mid aDb$
- $D \rightarrow abC \mid aCB \mid aF$
- $F \rightarrow Fb \mid aCb$

Table Text Size

LHS	RHS
S	→ aSb
S	→ aAb
A	→ BA
A	→ λ
A	→ aCB
B	→ ASb
B	→ abB
C	→ aDF
C	→ aDb
D	→ abC
D	→ aCB
D	→ aF
F	→ Fb
F	→ aCb

Do Step Do All Proceed Export

Lambda removal complete.
 "Proceed" or "Export" available.
 Set that derives lambda: [A]

Delete Complete Selected

LHS	RHS
S	→ aSb
S	→ aAb
A	→ BA
A	→ aCB
B	→ ASb
B	→ abB
C	→ aDF
C	→ aDb
D	→ abC
D	→ aCB
D	→ aF
F	→ Fb
F	→ aCb
S	→ ab
A	→ B
B	→ Sb

Elimina cadenas vacías

LHS	RHS
S	→ aSb
S	→ aAb
A	→ BA
A	→ aCB
B	→ ASb
B	→ abB
C	→ aDF
C	→ aDb
D	→ abC
D	→ aCB
D	→ aF
F	→ Fb
F	→ aCb
S	→ ab
A	→ B
B	→ Sb

Do Step Do All Proceed Export

Unit removal complete.
 "Proceed" or "Export" available.

⏪ ⏩

Automaton Size

Delete Complete Selected

LHS	RHS
S	→ aSb
S	→ aAb
A	→ BA
A	→ aCB
B	→ ASb
B	→ abB
C	→ aDF
C	→ aDb
D	→ abC
D	→ aCB
D	→ aF

Elimina Unitarios

LHS	RHS
S	→ aSb
S	→ aAb
S	→ ab
A	→ ASb
A	→ Sb
A	→ abB
B	→ ASb
B	→ Sb
F	→ aCb
F	→ Fb
D	→ aF
D	→ aCB
D	→ abC
C	→ aDb
C	→ aDF
B	→ abB
A	→ aCB
A	→ BA

Do Step Do All Proceed Export
Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, B, S]

Automaton Size

LHS	RHS
S	→ aSb
S	→ aAb
S	→ ab
A	→ ASb
A	→ Sb
A	→ abB
B	→ ASb
B	→ Sb
B	→ abB
A	→ BA

Elimina inaccesibles

Start Pause Step Noninverted Tree
Input aaabbb
String accepted! 9 nodes generated.
Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing L...
Table Text Size

Input	Result
ab	Accept
aaabbb	Accept
aabb	Accept
aaaabbbb	Accept

LHS	RHS
S	→ aSb
S	→ aAb
S	→ ab
A	→ BA
B	→ ASb
B	→ abB
B	→ Sb
A	→ abB
A	→ Sb
A	→ ASb

Derived ab from S. Derivations complete.

Load Inputs Run Inputs Clear Enter Lambda

Gramática Limpia

Start Pause Step Noninverted Tree

Input String: **aabb**

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...)

Table Text Size

LHS	RHS
S	→ CE
S	→ CG
G	→ AD
S	→ CD
A	→ BA
B	→ AE
B	→ CF
B	→ SD
A	→ CF
F	→ DB
C	→ a
A	→ SD

Derived b from D. Derivations complete.

Table Text Size

Input	Result
ab	Accept
aaabbb	Accept
aabb	Accept
aaaabbbb	

Load Inputs Run Inputs Clear Enter Lambda

FNC

$$L = \{a^n b^m \mid n=m=1,2,3,\dots\}$$

5. $S \rightarrow A \mid AA \mid AAA$
 $A \rightarrow ABa \mid ACa \mid a$
 $B \rightarrow ABa \mid Ab \mid \lambda$
 $C \rightarrow Cab \mid CC$
 $D \rightarrow CD \mid Cd \mid CEa$
 $E \rightarrow b$

Table Text Size

LHS	RHS
S	→ A
S	→ AA
S	→ AAA
A	→ ABa
A	→ Ab
A	→ λ
B	→ ABa
B	→ Ab
B	→ λ
C	→ Cab
C	→ CC
D	→ CD
D	→ Cd
D	→ CEa
E	→ b

Do Step Do All Proceed Export


Lambda removal complete.
 "Proceed" or "Export" available.
 Set that derives lambda: [A, B, S]

Delete Complete Selected

LHS	RHS
S	→ A
S	→ AA
S	→ AAA
A	→ ABa
A	→ Ab
B	→ ABa
B	→ Ab
C	→ Cab
C	→ CC
D	→ CD
D	→ Cd
D	→ CEa
E	→ b
A	→ Aa
A	→ Ba
A	→ a

Elimina Cadenas Vacías

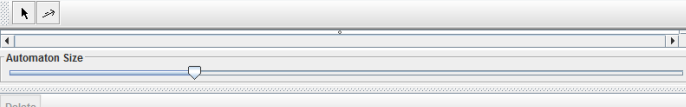
LHS		RHS
S	→	A
S	→	AA
S	→	AAA
A	→	ABa
A	→	Ab
B	→	ABa
B	→	Ab
C	→	Cab
C	→	CC
D	→	CD
D	→	Cd
D	→	CEa
E	→	b
A	→	Aa
A	→	Ba
A	→	a
A	→	b
B	→	Aa
B	→	Ba
B	→	a
B	→	b

Do Step Do All Proceed Export
Unit removal complete.
"Proceed" or "Export" available.

Automaton Size
Delete Complete Selected

LHS		RHS
S	→	AA
S	→	AAA
A	→	ABa
A	→	Ab
B	→	ABa
B	→	Ab
C	→	Cab
C	→	CC
D	→	CD
D	→	Cd
D	→	CEa

Elimina Unitarios

LHS		RHS
S	→	AA
S	→	AAA
S	→	Ba
S	→	Ab
S	→	Aa
S	→	b
S	→	a
S	→	ABa
A	→	b
A	→	a
A	→	Ba
A	→	Aa
A	→	Ab
A	→	ABa
B	→	ABa
B	→	Ab
B	→	Aa
B	→	b
B	→	a
B	→	Ba
C	→	CC

Do Step Do All Proceed Export
Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, B, E, S]

Automaton Size
Delete

LHS		RHS
S	→	AA
S	→	AAA
S	→	Ba
S	→	Ab
S	→	Aa
S	→	b
S	→	a
S	→	ABa
A	→	b
A	→	a
A	→	Ba
A	→	Aa
A	→	Ab
A	→	ABa

Elimina Inaccesibles

Start Pause Step Noninverted Tree

Input: **aaaabbbb**

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...)

Table Text Size

LHS	RHS
S	→ AA
S	→ AAA
S	→ Ba
S	→ Ab
S	→ Aa
S	→ b
S	→ a
S	→ ABa
A	→ ABa
A	→ Ab
A	→ Aa
B	→ ABa
B	→ Ab
B	→ Aa
B	→ Ba

Derived b from A. Derivations complete.

Input	Result
ab	Accept
aaabbb	Accept
aabb	Accept
aabbaa	Accept
bbaabb	Accept

Load Inputs Run Inputs Clear Enter Lambda

Gramática limpia

Start Pause Step Noninverted Tree

Input String: **aabbaa**

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...)

Table Text Size

LHS	RHS
S	→ AA
S	→ AE
E	→ AA
S	→ BC
S	→ AD
S	→ AC
S	→ b
S	→ a
S	→ AF
A	→ AF
A	→ AD
A	→ AC

Derived a from C. Derivations complete.

Input	Result
ab	Accept
aaabbb	Accept
aabb	Accept
aabbaa	Accept
bbaabb	Accept

Load Inputs Run Inputs Clear Enter Lambda

FNC

$$L = \{a^n b^m, b^n a^m, a^n b^m a^n, b^n a^m b^n \mid n=m=1,2,3,\dots\}$$

6. $S \rightarrow aAb \mid cHB \mid CH$

$A \rightarrow dBH \mid eeC$

$B \rightarrow ff \mid D$

$C \rightarrow gFB \mid ah$

$D \rightarrow i$

$E \rightarrow jF$

$F \rightarrow deGGG \mid eF$

G->kF
H->Hlm

LHS	RHS
S	→ aAb
S	→ cHB
S	→ CH
A	→ dBH
A	→ eeC
B	→ ff
B	→ D
C	→ gFB
C	→ ah
D	→ i
E	→ jF
F	→ deGGG
F	→ eF
G	→ kF
H	→ Hlm

Do Step Do All Proceed Export

Unit removal complete.
"Proceed" or "Export" available.

Automaton Size

Delete Complete Selected

LHS	RHS
S	→ aAb
S	→ cHB
S	→ CH
A	→ dBH
A	→ eeC
B	→ ff
C	→ gFB
C	→ ah
D	→ i
E	→ jF
F	→ deGGG
F	→ eF
G	→ kF
H	→ Hlm
R	→ i

Elimina Unitarios

LHS	RHS
S	→ aAb
S	→ cHB
S	→ CH
B	→ i
H	→ Hlm
G	→ kF
F	→ eF
F	→ deGGG
E	→ jF
D	→ i
C	→ ah
C	→ gFB
B	→ ff
A	→ eeC
A	→ dBH

Do Step Do All Proceed Export

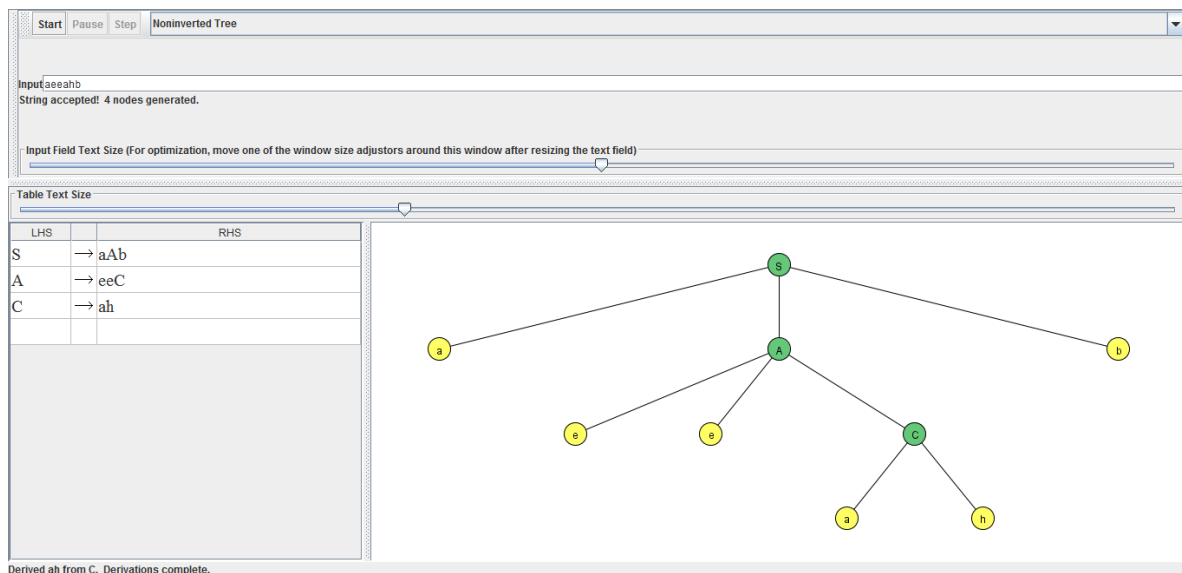
Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, B, C, D, S]

Automaton Size

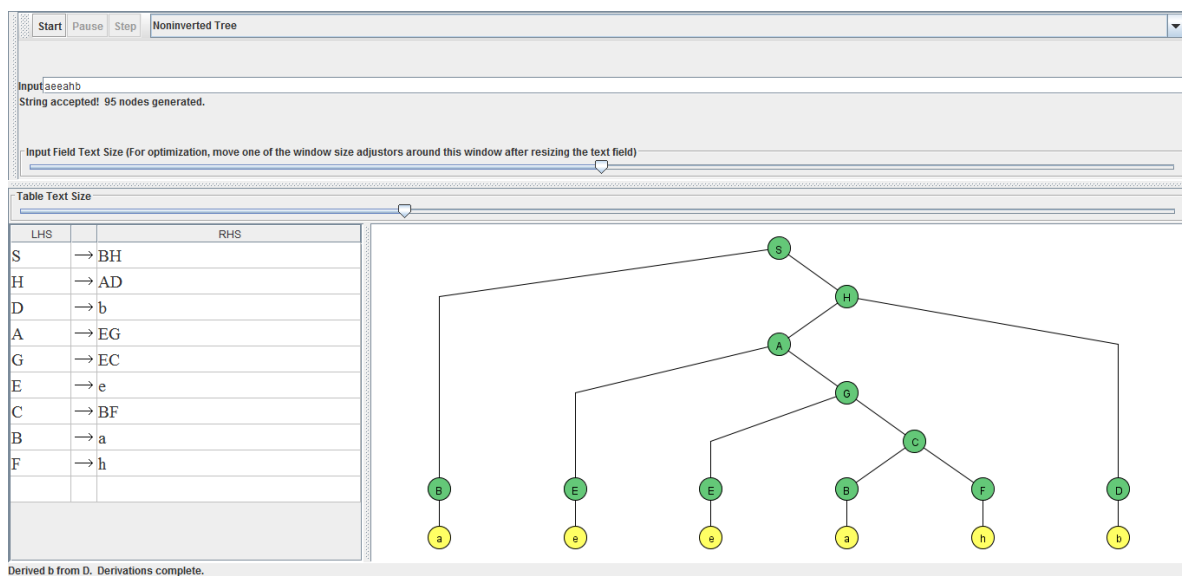
Delete

LHS	RHS
S	→ aAb
C	→ ah
A	→ eeC

Elimina inaccesibles



Gramática limpia



FNC

$L = \{aeeahb\}$ (Sólo acepta una cadena)

7. $S \rightarrow aAB \mid bBA \mid ABb \mid SS$

$A \rightarrow aAb \mid CCA \mid BB$

$B \rightarrow \lambda \mid bC$

$C \rightarrow aCS \mid SCS$

$D \rightarrow ab \mid SABC$

Table Text Size				
LHS		RHS	Do Step	Do All
S	→	aAB	Proceed	Export
S	→	bBA	Lambda removal complete. "Proceed" or "Export" available. Set that derives lambda: [A, B]	
S	→	ABb		
S	→	SS	Delete	Complete Selected
A	→	aSb	LHS	RHS
A	→	CCA	S	→ aAB
A	→	BB	S	→ bBA
B	→	λ	S	→ ABb
B	→	bC	S	→ SS
C	→	aCS	A	→ aSb
C	→	SCS	A	→ CCA
D	→	ab	A	→ BB
D	→	SABC	B	→ bC
			C	→ aCS
			C	→ SCS
			D	→ ab
			D	→ SABC
			S	→ a
			S	→ aA
			S	→ aB
			S	→ b

Elimina cadenas vacías

Table Text Size				
LHS		RHS	Do Step	Do All
S	→	aAB	Proceed	Export
S	→	bBA	Unit removal complete. "Proceed" or "Export" available.	
S	→	ABb		
S	→	SS	Automation Size	
A	→	aSb	Delete	Complete Selected
A	→	CCA	LHS	RHS
A	→	BB	S	→ aAB
B	→	bC	S	→ bBA
C	→	aCS	S	→ ABb
C	→	SCS	S	→ SS
D	→	ab	A	→ aSb
D	→	SABC	A	→ CCA
S	→	a	A	→ BB
S	→	aA	B	→ bC
S	→	aB	C	→ aCS
S	→	b	C	→ SCS
S	→	bA	D	→ ab
S	→	bB	D	→ SABC
S	→	Ab	S	→ a
S	→	Bb	S	→ aA
A	→	CC		

Elimina Unitarios

LHS	RHS
S	→ aAB
S	→ bBA
S	→ ABb
S	→ SS
S	→ a
S	→ aA
S	→ aB
S	→ b
S	→ bA
S	→ bB
S	→ Ab
S	→ Bb
A	→ bC
A	→ CC
D	→ SC
D	→ SBC
D	→ SAC
D	→ SABC
D	→ ab
C	→ SCS
C	→ aCS

Do Step Do All Proceed Export
Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, D, S]

Automaton Size

LHS	RHS
S	→ SS
S	→ a
S	→ aA
S	→ b
S	→ bA
S	→ Ab
A	→ aSb

Elimina Inaccessibles

Start Pause Step Noninverted Tree
Input aaabb
String accepted! 240 nodes generated.
Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...)

Table Text Size

LHS	RHS
S	→ SS
S	→ a
S	→ aA
S	→ b
S	→ bA
S	→ Ab
A	→ aSb

Derived b from S. Derivations complete.

Input	Result
a	Accept
b	Accept
aaabb	Accept
aaaabbb	Accept

Load Inputs Run Inputs Clear Enter Lambda

Gramática limpia

Start Pause Step Noninverted Tree

Input aaaabbb
String accepted! 11712 nodes generated.

Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...

Table Text Size

LHS	RHS
S	→ SS
S	→ a
S	→ BA
S	→ b
S	→ CA
S	→ AC
A	→ BD
D	→ SC
B	→ a
C	→ b

Derived b from S. Derivations complete.

Input	Result
a	Accept
b	Accept
aaabb	Accept
aaaabbb	Accept

Load Inputs Run Inputs Clear Enter Lambda

FNC

$$L = \{a^n b^m \mid n > m, n = 1, 2, 3, \dots, m = 1, 2, 3, \dots\}$$

8. $S \rightarrow CBa \mid D$
 $A \rightarrow bbC$
 $B \rightarrow Sc \mid ddd$
 $C \rightarrow eA \mid f \mid C$
 $D \rightarrow E \mid SABC$
 $E \rightarrow gh$

LHS	RHS
S	→ CBa
S	→ D
A	→ bbC
B	→ Sc
B	→ ddd
C	→ eA
C	→ f
C	→ C
D	→ E
D	→ SABC
E	→ gh

Do Step Do All Proceed Export

Unit removal complete.
"Proceed" or "Export" available.

Automaton Size

Delete Complete Selected

LHS	RHS
S	→ CBa
A	→ bbC
B	→ Sc
B	→ ddd
C	→ eA
C	→ f
D	→ SABC
E	→ gh
S	→ gh
D	→ gh
S	→ SABC

Elimina Unitarios

LHS	RHS
S	→ CBa
S	→ gh
S	→ SABC
D	→ gh
E	→ gh
D	→ SABC
C	→ f
C	→ eA
B	→ ddd
B	→ Sc
A	→ bbC

Do Step Do All Proceed Export
Useless removal complete.
"Proceed" or "Export" available.
Variables that predicate terminals: [A, B, C, D, E, S]

Automaton Size

Delete

LHS	RHS
S	→ CBa
S	→ gh
S	→ SABC
C	→ f
C	→ eA
B	→ ddd
B	→ Sc
A	→ bbC

Elimina inaccesibles

Start Pause Step Noninverted Tree
Input fddda
String accepted! 6 nodes generated.
Input Field Text Size (For optimization, move one of the window size adjusters around this window after resizing L...)

Table Text Size

LHS	RHS
S	→ CBa
S	→ gh
S	→ SABC
A	→ bbC
B	→ Sc
B	→ ddd
C	→ eA
C	→ f

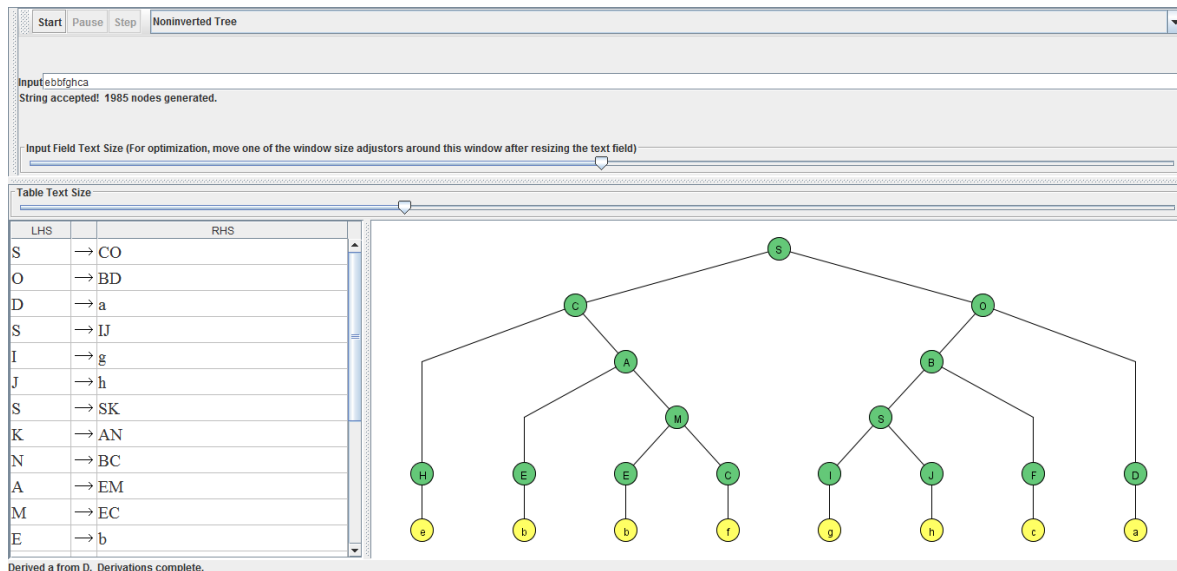
Derived ddd from B. Derivations complete.

Table Text Size

Input	Result
gh	Accept
ghbbfdddacf	Accept
fddda	Accept
ebbfghca	Accept

Load Inputs Run Inputs Clear Enter Lambda

Gramática limpia



FNC

L) {ghbbffddacf, fddda, ebbfghca, ...}

CONCLUSIÓN

En ésta práctica se realizó la conversión paso a paso para limpiar una gramática, y posteriormente convertirla en su forma normal de Chomsky, me di cuenta que visualizarlo paso a paso abre muchas posibilidades a cómo lo harías a papel, ya que aunque el procedimiento es el mismo, los pasos que se hace en el programa a como se hacen analíticamente son algo distintos, pero siempre se va a llegar a la misma conclusión o lo que es lo mismo a la misma gramática limpia.

BIBLIOGRAFÍA

- 1) <http://repositori.uji.es/xmlui/bitstream/handle/10234/5875/bolAuto5.pdf;jsessionid=44C764C197A3310BFCA622A5D5BF5B54?sequence=6>
- 2) <http://teodelacomp.blogspot.com/2011/03/forma-normal-de-chomsky.html>