UNIDAD N°4 AUTÓMATAS FINITOS NO DETERMINISTAS

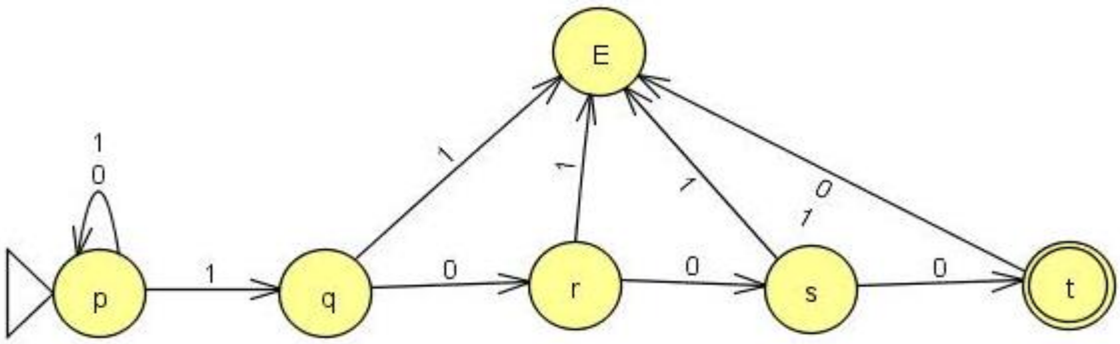
Recordemos la definición de un AFND
AFND = (Σ_E, Q, q_0, A, f) con $f: Q \times \Sigma_E \rightarrow \boldsymbol{P}(Q)$

donde **P(Q)** es el conjunto de todos los subconjuntos que se pueden formar con elementos del conjunto **Q**

Ejercicios propuestos referidos a construcción de AFND

Ejercicio 1

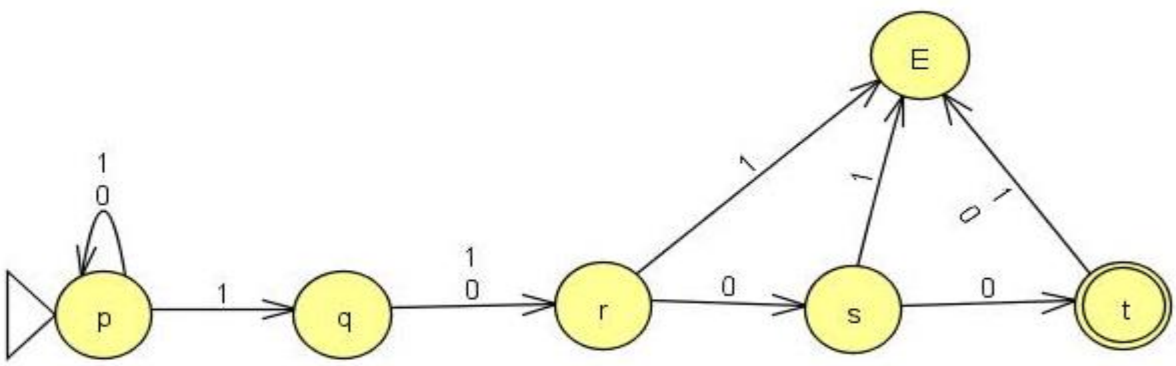
Reconocer las condiciones de error del AFND del Ejemplo 4.1 e incorporarlas en la definición formal del autómata y su grafo. El ejemplo 4.1 pide proponer y analizar un AFND destinado a reconocer cadenas que respondan a la forma general α = (0+1)*1000



AFND1 = ({0,1} , {p, q, r , s , t , E }, p, {t} , f)

Ejercicio 2

Reconocer las condiciones de error del AFND del Ejemplo 4.2 e incorporarlas en la definición formal del autómata y su grafo. El ejemplo 4.2 pide proponer y analizar un AFND destinado a reconocer cadenas que respondan a la forma general α = (0+1)*1 (0 + 1) 00.



AFND2 = ({0,1} , {p, q, r , s , t , E }, p, {t} ,f)

Ejercicios propuestos de conversión de AFND a AFD

La equivalencia entre AFND y AFD puede obtenerse por medio de dos Teoremas que fueron presentados en el teórico y también a través de otros métodos como el *procedimiento sobre la tabla de transición*. La resolución de ejercicios prácticos es a través del Teorema 2 de la siguiente manera:

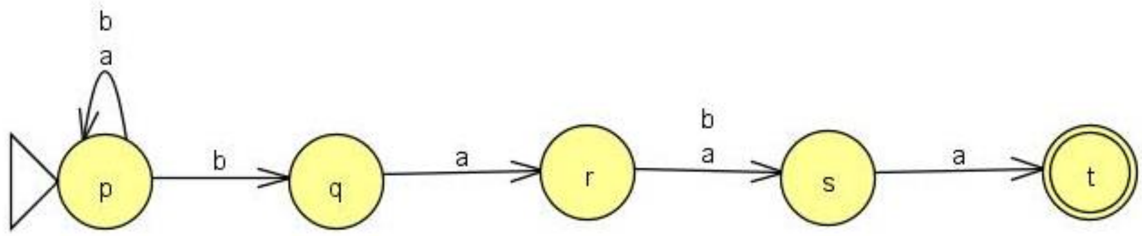
- Ejercicios prácticos de AFN sin λ, aplicamos el Teorema 2, Caso 1.
- Ejercicios prácticos de AFN con λ, aplicamos el Teorema 2, Caso 2.

CONVERSIÓN DE AFND A AFD SIN LAMBDA

Ejercicio 3

Proponer un AFDN que reconozca cadenas de la forma general **α = (a+b) * ba (a+b) a** y luego convertirlo a un AFD equivalente. Para ambos autómatas presentar las definiciones formales y grafos.

AFND3 = ({a,b} , {p, q, r , s , t }, p, {t} , f)



f	a	b
->p	{p}	{p,q}
q	{r}	
r	{s}	{s}
s	{t}	
*t		

Procedimiento de conversión de AFND a AFD:

1° Paso: determinar el nuevo estado inicial que será el estado inicial original { p } renombrado en C_0 , quedando C_0 = {p}

2° Paso: a partir de este nuevo estado inicial se comienzan a determinar las transiciones y con ello los nuevos estados hasta que no se generen nuevos estados.

f(C_0, a) = es lo mismo que decir f(p,a) ya que {p } =C_0 y como vemos en la tabla de transición transita al conjunto de estados { p } que ahora se denomina C_0

f(C_0, b) = puede transitar al conjunto { p,q }, como el conjunto no es el mismo de C_0 entonces a este conjunto lo llamamos estado C_1 por cada conjunto nuevo de estados voy creando nuevos estados y renombrando.

En la tabla de transición del AFD podemos visualizar las transiciones del primer estado

AFD	**a**	**b**
→ C_0	C_0	C_1

Continuamos con el siguiente estado que es C_1

$f(C_1, a) = \{p,r\}$ C_2 para completar las transiciones tomo cada estado por separado y me fijo en la tabla original dónde transita cada uno y así formo el conjunto { p,r }

f	a
->p	{p}
q	{r}

como es un conjunto diferente a los anteriores le asigno el nombre de estado C_2

$f(C_1, b) = \{p,q\}$ C_1 en este caso solo tiene transición el estado f(p,b) = {p,q}

En la tabla de transición del AFD podemos visualizar las transiciones del 2do estado

AFD	**a**	**b**
→ C_0	C_0	C_1
C_1	C_2	C_1

a continuación se muestran directamente el resultado de las nuevas transiciones

$C_0 = \{p\}$

$f(C_0, a) = \{p\}$ C_0

$f(C_0, b) = \{p,q\}$ C_1

$f(C_1, a) = \{p,r\}$ C_2

$f(C_1, b) = \{p,q\}$ C_1

$f(C_2, a) = \{p,s\}$ C_3

$f(C_2, b) = \{p,q,s\}$ C_4

$f(C_3, a) = \{p,t\}$ C_5

$f(C_3, b) = \{p,q\}$ C_1

$f(C_4, a) = \{p,r,t\}$ C_6

$f(C_4, b) = \{p,q\}$ C_1

$f(C_5, a) = \{p\}$ C_0

$f(C_5, b) = \{p,q\}$ C_1

$f(C_6, a) = \{p,s\}$ C_3

$f(C_6, b) = \{p,q,s\}$ C_4

f	a	b
->p	{p}	{p,q}
q	{r}	
r	{s}	{s}
s	{t}	
*t		

f	a	b
->C_0	C_0	C_1
C_1	C_2	C_1
C_2	C_3	C_4
C_3	C_5	C_1
C_4	C_6	C_1
*C_5	C_0	C_1
*C_6	C_3	C_4

AFDE3 = ({a,b} , { C_0, C_1 , C_2 , C_3 , C_4 , C_5 , C_6 }, C_0 , { C_5 , C_6} ,f)

Los estados de aceptación del AFDE van a ser aquellas clases que contengan al estado de aceptación del AFND original, en este caso C_5 y C_6 que contienen al estado t.

Ejercicio 4

Dado el siguiente AFND, se pide:

a) Construir un árbol de descripciones instantáneas para cada una de las siguientes cadenas: 1000, 1001, 1101

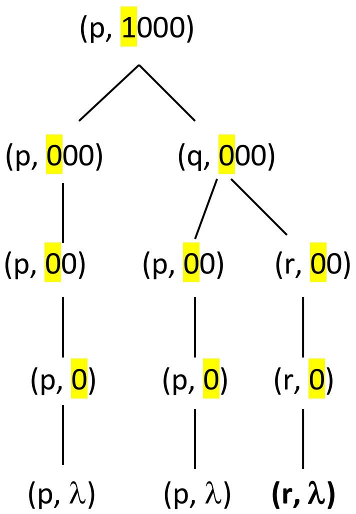
b) Realizar la conversión del AFND a AFD

f	0	1
->p	{p}	{p,q}
q	{p,r}	{q,r}
*r	{r}	{q}

AFND4 = ({0,1], {p, q, r}, p , {r}, f)

a) para construir el árbol de transición de una cadena se deben crear tantas ramas como transiciones tenga cada símbolo de la palabra

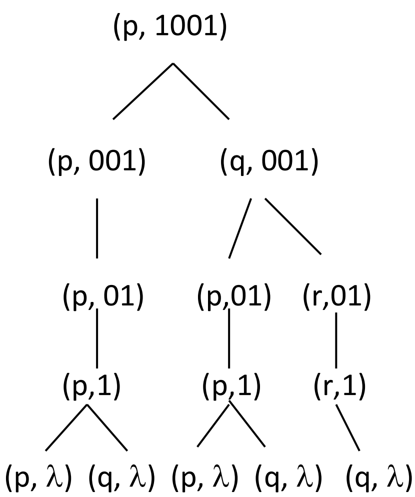
cadena 1000



f	0	1
->p	{p}	{p,q}
q	{p,r}	{q,r}
*r	{r}	{q}

El estado r es estado de aceptación por lo tanto la cadena es aceptada.

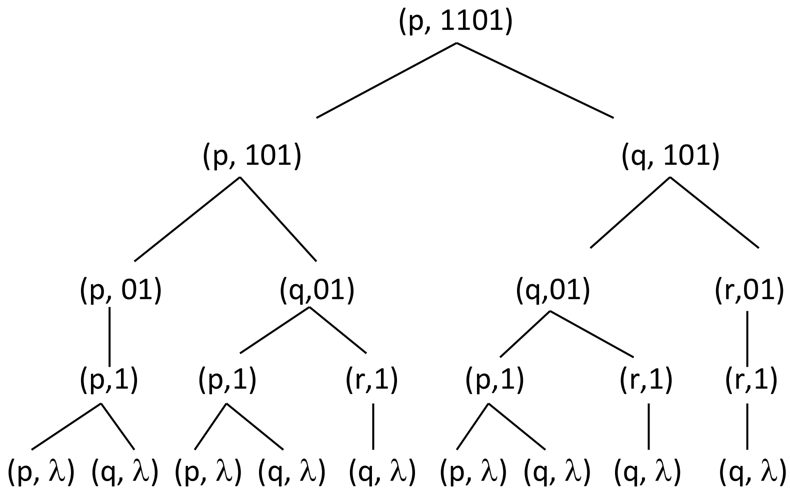
cadena 1001



f	0	1
->p	{p}	{p,q}
q	{p,r}	{q,r}
*r	{r}	{q}

El estado r es estado de aceptación y como se puede ver, por ninguna rama se llega a dicho estado, por lo tanto la cadena es rechazada.

cadena 1101



f	0	1
->p	{p}	{p,q}
q	{p,r}	{q,r}
*r	{r}	{q}

El estado r es estado de aceptación y como se puede ver, por ninguna rama se llega a dicho estado, por lo tanto la cadena es rechazada.

b) C_0 = { p }

f(C_0, 0) = { p } C_0
f(C_0, 1) = { p,q } C_1
f(C_1, 0) = { p,r } C_2
f(C_1, 1) = { p,q,r } C_3
f(C_2, 0) = { p ,r} C_2
f(C_2, 1) = { p,q } C_1
f(C_3, 0) = { p,r } C_2
f(C_3, 1) = { p,q,r } C_3

f	0	1
->p	{p}	{p,q}
q	{p,r}	{q,r}
*r	{r}	{q}

AFDE4 = ({0,1} , { C_0, C_1 , C_2 , C_3 }, C_0 , { C_2 , C_3} , f)

f	0	1
->C_0	C_0	C_1
C_1	C_2	C_3
*C_2	C_2	C_1
*C_3	C_2	C_3

CONVERSIÓN DE AFND-λ A AFD

La conversión de un AFND con λ se comienza por formar el conjunto T y T*.

T es el conjunto de todos los pares de estados que están vinculados con λ. Se considera que cada estado consigo mismo puede tener una transición λ.

T* es el conjunto T que se le aplica la propiedad transitiva, puede que se generen o no nuevos pares de estados.

El método de conversión es a través del Teorema 2, caso 2 basados en T*.

Ejercicios propuestos de conversión de AFND-λ a AFD

Ejercicio 6

Dado el siguiente AFND, se pide:

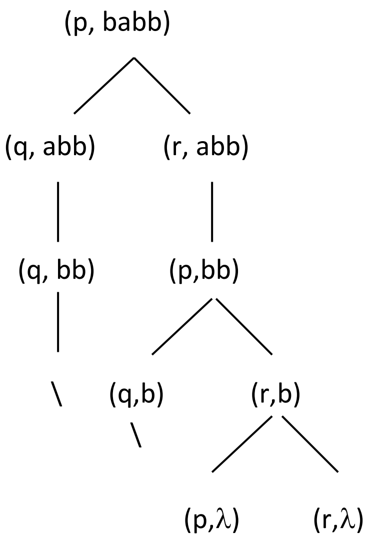
a) Construir un árbol de descripciones instantáneas para cada una de las siguientes cadenas: babb, bbbb, bbbaa

b) Realizar la conversión del AFND a AFD.

AFND-λ = ({a,b}, {p, q , r, s }, p, {r}, f)

f	a	b	λ
->p		{q,r}	{q}
q	{q}		{s}
*r	{p}	{p,r}	
s			

a) cadena babb es aceptada (solo se construye el árbol de esta cadena, se aclara que no están graficadas las transiciones λ, debido a que el gráfico del árbol se torna muy voluminoso) la cadena bbbb es aceptada y cadena bbbaa no es aceptada.



f	a	b	λ
->p		{q,r}	{q}
q	{q}		{s}
*r	{p}	{p,r}	
s			

b) En presencia de lambda es conveniente agrupar a todos los estados que están vinculados por una transición lambda en un conjunto T:

1° Paso: definir el conjunto T = {(p,p), (p,q), (q,q), (q,s), (r,r), (s,s)}

Luego se calcula las relaciones de transitividad de estos pares de estados

2° Paso: Buscar relaciones de transitividad en el conjunto T:

T* = { (p,p),(p,q), (p,s), (q,q), (q,s) (r,r), (s,s) }

se agrega (p,s) por transitividad ya que si p se relaciona con λ a través de q y q se relaciona a través de λ con s por lo tanto hay una relación por transitividad de p con s a través de q.

3° Paso: es conveniente completar la columna de λ en la tabla antes de continuar el proceso:

f	a	b	λ
->p		{q,r}	{p,q,s}
q	{q}		{q,s}
*r	{p}	{p,r}	{r}
s			{s}

4° Paso: definir el nuevo estado inicial, para ello se busca en la tabla con quién se relaciona p a través de λ, entonces

$f(p,\lambda) = \{ p,q,s \}$ C_0

Recordamos que T* = { (p,p),(p,q), (p,s), (q,q), (q,s) (r,r), (s,s) }

5° Paso: determinar dónde van a transitar los estados C_i

$f(C_0, a) = \{ q \} \xrightarrow{\lambda} \{q,s\}$ C_1

$f(C_0, b) = \{ q,r \} \xrightarrow{\lambda} \{q,r,s\}$ C_2

$f(C_1, a) = \{ q \} \xrightarrow{\lambda} \{q,s\}$ C_1

$f(C_1, b) = \theta$

$f(C_2, a) = \{ p,q \} \xrightarrow{\lambda} \{ p,q,s \}$ C_0

$f(C_2, b) = \{ p,r \} \xrightarrow{\lambda} \{p,q,r,s\}$ C_3

$f(C_3, a) = \{ p,q \} \xrightarrow{\lambda} \{ p,q,s \}$ C_0

$f(C_3, b) = \{ p,q,r \} \xrightarrow{\lambda} \{p,q,r,s\}$ C_3

f	a	b	λ
->p		{q,r}	{q}
q	{q}		{s}
*r	{p}	{p,r}	
s			

f	a	b
->C_0	C_1	C_2
C_1	C_1	
*C_2	C_0	C_3
*C_3	C_0	C_3

AFDE6 = ({a,b} , { C_0, C_1 , C_2 , C_3 }, C_0 , { C_2 , C_3} , f)

Ejercicio 7

Dado el siguiente AFND, se pide:

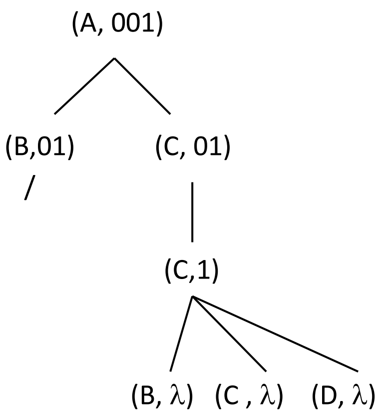
a) Construir un árbol de descripciones instantáneas para cada una de las siguientes cadenas: 001, 1100, 101.

b) Realizar la conversión del AFND a AFD.

AFND7-λ = ({0,1}, {A, B, C, D}, A, {D}, f)

f	0	1	λ
->A	{B,C}	{A}	{B,D}
B		{B,D}	{D}
*C	{C}	{B,C,D}	
D		{A,C}	{B}

a) cadena 001 es aceptada. (solo se construye el árbol de esta cadena, se aclara que no están graficadas las transiciones λ, debido a que el gráfico del árbol se torna muy voluminoso); las cadenas 1100 y 101 también son aceptadas.



b) T= { (A,A), (A,B), (A,D), (B,B), (B,D), (C,C), (D,D), (D,B)}
T* = T

f	0	1	λ
->A	{B,C}	{A}	{A,B,D}
B		{B,D}	{B,D}
*C	{C}	{B,C,D}	{C}
D		{A,C}	{D,B}

f(A,λ) = { A,B,D } C_0

f(C_0, 0) = {B,C } $\xrightarrow{\lambda}$ {B,C,D} C_1

f(C_0, 1) = { A,B,C,D } $\xrightarrow{\lambda}$ {A,B,C,D} C_2

f(C_1, 0) = { C } $\xrightarrow{\lambda}$ {C} C_3

f(C_1, 1) = { A,B,C,D } $\xrightarrow{\lambda}$ {A,B,C,D} C_2

f(C_2, 0) = { B,C} $\xrightarrow{\lambda}$ {B,C,D } C_1

f(C_2, 1) = { A,B,C,D } $\xrightarrow{\lambda}$ {A,B,C,D} C_2

f(C_3, 0) = { C } $\xrightarrow{\lambda}$ {C} C_3

f(C_3, 1) = { B,C,D } $\xrightarrow{\lambda}$ {B,C,D} C_1

f	0	1
->C_0	C_1	C_2
*C_1	C_3	C_2
*C_2	C_1	C_2
*C_3	C_3	C_1

AFDE7 = ({0, 1} , { C_0, C_1 , C_2 , C_3 }, C_0 , {C_1, C_2 , C_3} , f)

Ejercicio 8

Dado el siguiente AFND, se pide:

a) Construir un árbol de descripciones instantáneas para cada una de las siguientes cadenas: 1000, 111, 1001.

b) Realizar la conversión del AFND a AFD.

AFND8-λ = ({0,1}, {p, q , r, s }, p, {q}, f)

f	0	1	λ
->p		{q,r}	{r}
*q	{p}	{p,q}	
r	{r}		{s}
s		{s}	

a) cadena 1000 no es aceptada (solo se construye el árbol de esta cadena, se aclara que no están graficadas las transiciones λ, debido a que el gráfico del árbol se torna muy voluminoso); la cadena 111 si es aceptada y la cadena 1001 no es aceptada.

(p, 1000)

(q, 000) (r, 000)

(p,00) (r,00)

\

(r,0)

(r,λ)

b)

T= { (p,p), (p,r), (q,q), (r,r), (r,s), (s,s)}

T* = { (p,p),(p,r), (p,s), (q,q), (r,r), (r,s), (s,s) }

se agrega (p,s) por transitividad ya que el estado p se relaciona con el estado r a través de λ y el estado r se relación con el estado s a través de λ, por lo que surge la relación por transitividad de p con s a través del estado r

f	0	1	λ
->p		{q,r}	{p,r,s}
*q	{p}	{p,q}	{q}
r	{r}		{r,s}
s		{s}	{s}

f(p,λ) = { p,r,s } C_0

f(C_0, 0) = { r } $\xrightarrow{\lambda}$ {r,s} C_1

f(C_0, 1) = { q,r ,s} $\xrightarrow{\lambda}$ {q,r,s} C_2

f(C_1, 0) = { r } $\xrightarrow{\lambda}$ {r,s} C_1

f(C_1, 1) = { s } $\xrightarrow{\lambda}$ { s} C_3

f(C_2, 0) = { p ,r} $\xrightarrow{\lambda}$ { p,r,s } C_0

f(C_2, 1) = { p,q,s } $\xrightarrow{\lambda}$ {p,q,r,s} C_4

f(C_3, 0) = θ

f(C_3, 1) = { s } $\xrightarrow{\lambda}$ { s} C_3

f(C_4, 0) = { p ,r} $\xrightarrow{\lambda}$ { p,r,s } C_0

f(C_4, 1) = { p,q,r,s } $\xrightarrow{\lambda}$ {p,q,r,s} C_4

f	0	1
->C_0	C_1	C_2
C_1	C_1	C_3
*C_2	C_0	C_4
C_3		C_3
*C_4	C_0	C_4

AFDE8 = ({0,1} , { C_0, C_1 , C_2 , C_3, C_4 }, C_0 , { C_2 , C_4} , f)

DEFINICIÓN DE AUTOMATAS A PARTIR DE GRAMÁTICA REGULARES

La construcción de Autómatas a partir de una gramática regular consiste en determinar el AF que valide el lenguaje que genera la gramática. El procedimiento es el siguiente partiendo de la gramática regular:

- El Σ_T de la gramática es el Σ_E del AFD.
- El Σ_N de la gramática es el conjunto Q del AF, es decir, que tendremos tantos estados como símbolos no terminales tenga la gramática
- Cada transición del AF se construye a través de cada regla de producción, es decir si tenemos X := aY da lugar a la transición f(X,a) = Y.
- Las reglas de producción formadas solo por símbolos terminales son las transiciones en el AF que se realizan al estado de aceptación, sería por cada X : =a tendremos f(X,a) = A siendo A el nuevo estado de aceptación que debe crearse

Ejercicios propuestos de definición del AF a partir de la gramática

Definir el autómata finito correspondiente, que reconozca el mismo lenguaje que genera cada una de las siguientes gramáticas regulares:

Ejercicio 9

Σ_T Σ_N S P

G_1 = ({0,1}, {D,E}, D, P_1)

P_1 = {D:=0E | 1, E := 0 | 1E |1 }

Σ_E Q q_0 A f

AF9 = ({ 0,1}, {D,E,F}, D, {F},f)

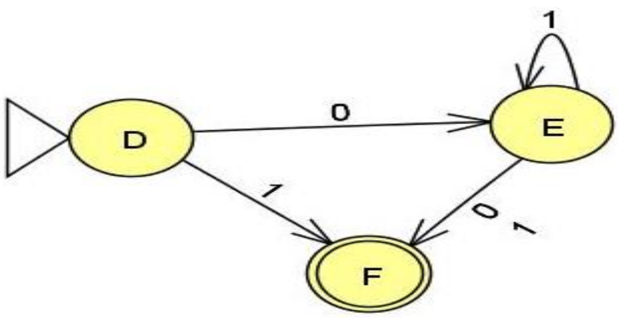
f(D,0)=E

f(D,1)=F

f(E,0)=F

f(E,1)=E

f(E,1)=F



Ejercicio 10

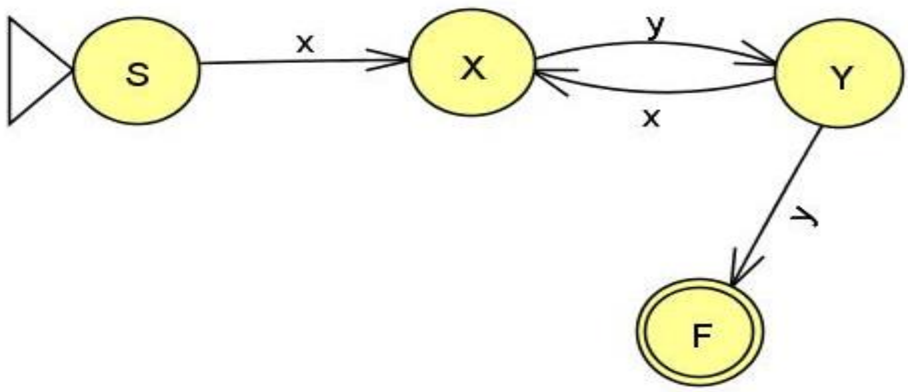
$\sum_T$ $\sum_N$ S P

G_2 = ({x,y}, {S,X,Y}, S, P_2)

P_2 = {S:=xX , X :=yY , Y:=xX | y }

$\sum_E$ Q q_0 A f

AF10 = ({x,y}, {S,X,Y,F}, S, {F},f)



Ejercicio 11

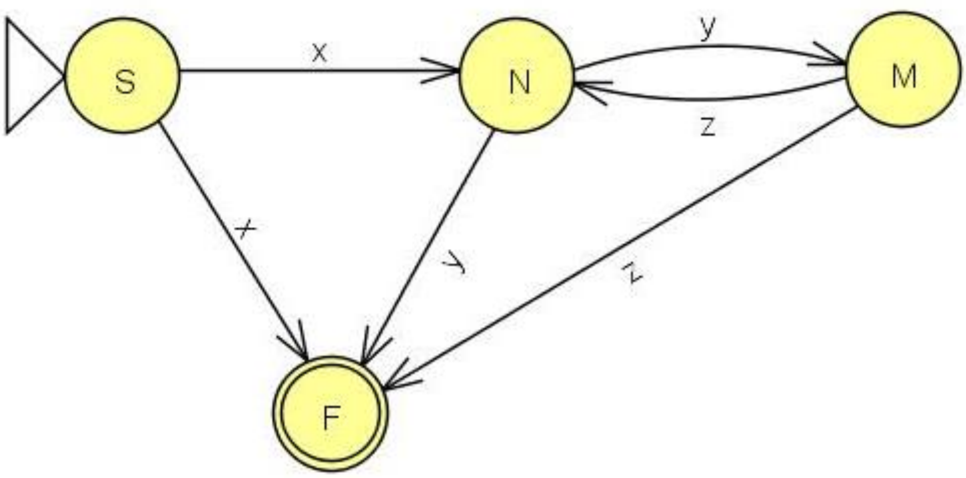
$\sum_T$ $\sum_N$ S P

G_3 = ({x,y,z}, {S,N,M}, S, P_3)

P_3 = {S:=xN | x, N := yM |y, M:= zN |z}

$\sum_E$ Q q_0 A f

AF10 = ({x,y,z}, {S,N,M,F}, S, {F},f)



DEFINICIÓN DE GRAMÁTICA REGULARES A PARTIR DE AUTOMATAS

La construcción de Gramáticas Regulares a partir de Autómatas consiste en definir el conjunto de reglas de producción de una gramática que generará lenguajes que están destinados a ser reconocidos por un cierto AFD.
El procedimiento es el siguiente partiendo de un autómata finito determinista:

- El Σ_E del AF es el Σ_T de la gramática
- El conjunto Q del AF es el Σ_N en la gramática, tendremos tanto símbolos no terminales como estados tenga el AF
- Cada regla de producción se construye a través de cada transición del AF, es decir que por cada f(X,a) = Y tendremos X := aY
- Cada transición al estado de aceptación no solo será una regla de producción con la transición al estado sino que también será una regla de producción con el símbolo terminal, ejemplo: siendo A estado de aceptación entonces por cada f(X,a)=A tendremos dos reglas de producción X:=aA X:= a

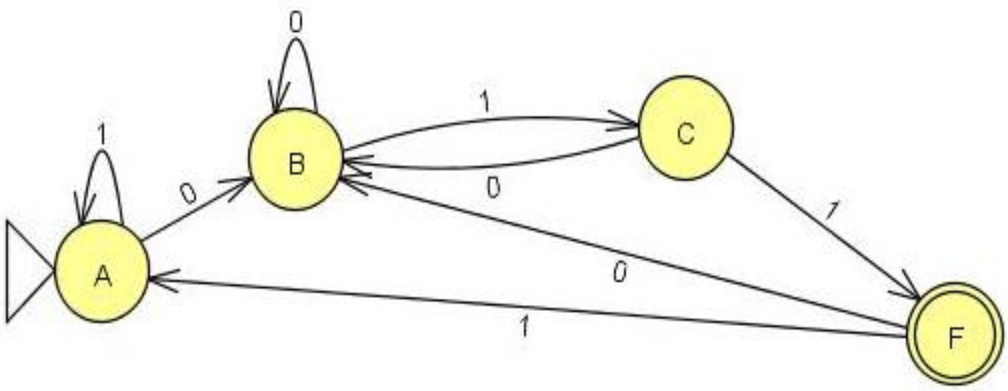
Ejercicios propuestos de definición de gramática a partir del AF

Definir la gramática regular correspondiente a los AF siguientes:

Ejercicio 12

Σ_E Q q_0 A f

AFD = ({0,1], {A,B,C,F}, A, {F} ,f)



Σ_T Σ_N S P

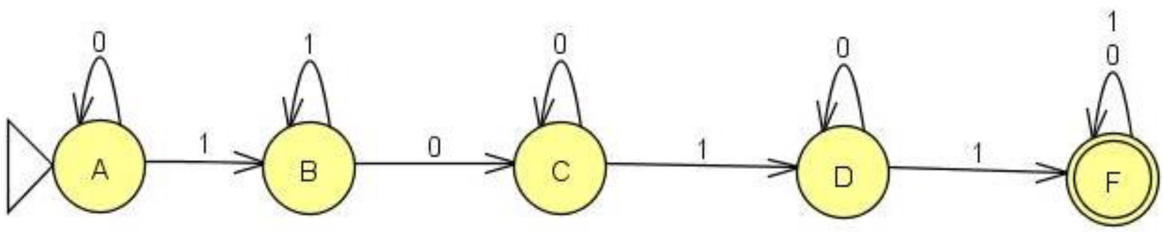
G_{12} = ({0,1}, {A,B,C,F}, A, P_{12})

P_{12} = {A:=1A | 0B, B := 0B | 1C , C:= 1F | 1 | 0B, F:= 0B | 1A }

Ejercicio 13

$\sum_E$ Q q_0 A f

AFD = ({0,1], {A,B,C,D,F}, A, {F} ,f)



$\sum_T$ $\sum_N$ S P

G_{13} = ({0,1}, {A,B,C,D,F}, A, P_{13})

P_{13} = {A:=0A | 1B, B := 1B | 0C , C:= 0C | 1D, D := 0D | 1F | 1, F:= 0F | 1F | 0 | 1 }