EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNM
TECNOLÓGICO NACIONAL DE MÉXICO

TECNOLÓGICO NACIONAL DE MEXICO

INSTITUTO TECNOLÓGICO DE IZTAPALAPA

INTEGRANTES:

CUANENEMI CUANALO MARIO ALBERTO	181080030
FERMIN CRUZ ERIK	181080007
GUTIERREZ ARELLANO RAFAEL	181080022
PEREZ ARMAS FAUSTO ISAAC	181080037

ISC-6AM

LENGUAJES Y AUTOMATAS I

M.C. ABIEL TOMÁS PARRA HERNÁNDEZ

SEP 2020 / FEB 2021

ACTIVIDAD SEMANA 14

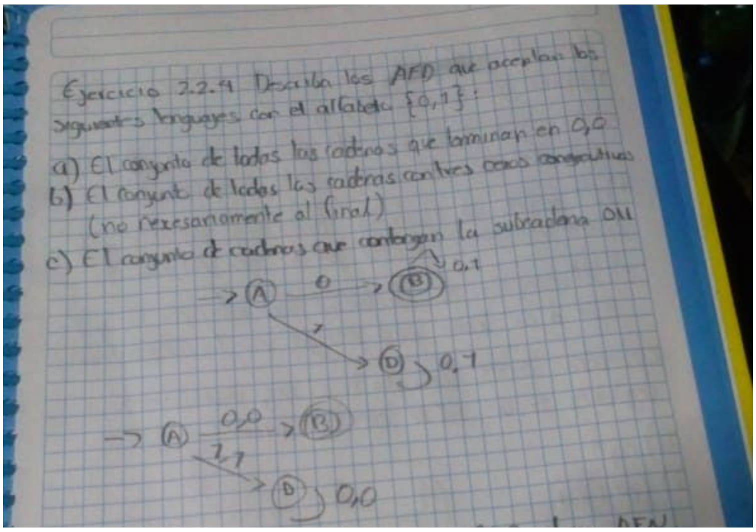
PRACTICAS DEL SEMESTRE

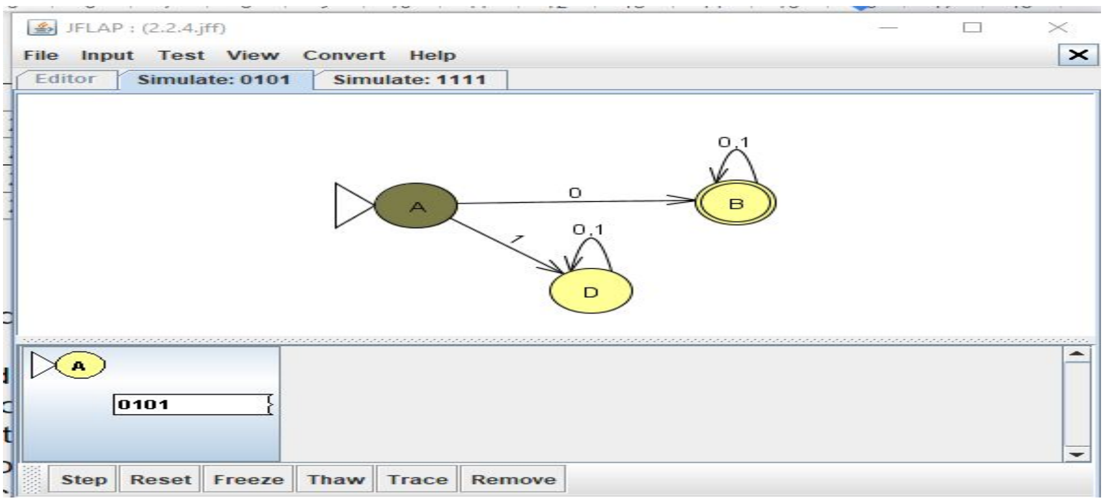
EJERCICIOS

Ejercicios de automatas				
Mario	2.2.4	2.3.2	3.1.1,	3.2.4
Fermin	2.2.5	2.3.7	3.1.2	3.3.1
Rafael	2.2.7	2.4.1	3.1.4	3.4.2
Fausto	2.2.8	2.5.3	3.2.3	3.4.3

Ejercicio 2.2.4

Ejercicio 2.2.4. Describa los AFD que aceptan los siguientes lenguajes con el alfabeto {0,1}: *
a) El conjunto de todas las cadenas que terminan en 00. b) El conjunto de todas las cadenas con tres ceros consecutivos (no necesariamente al final). c) El conjunto de cadenas que contengan la subcadena 011.





Ejercicio 2.2.5

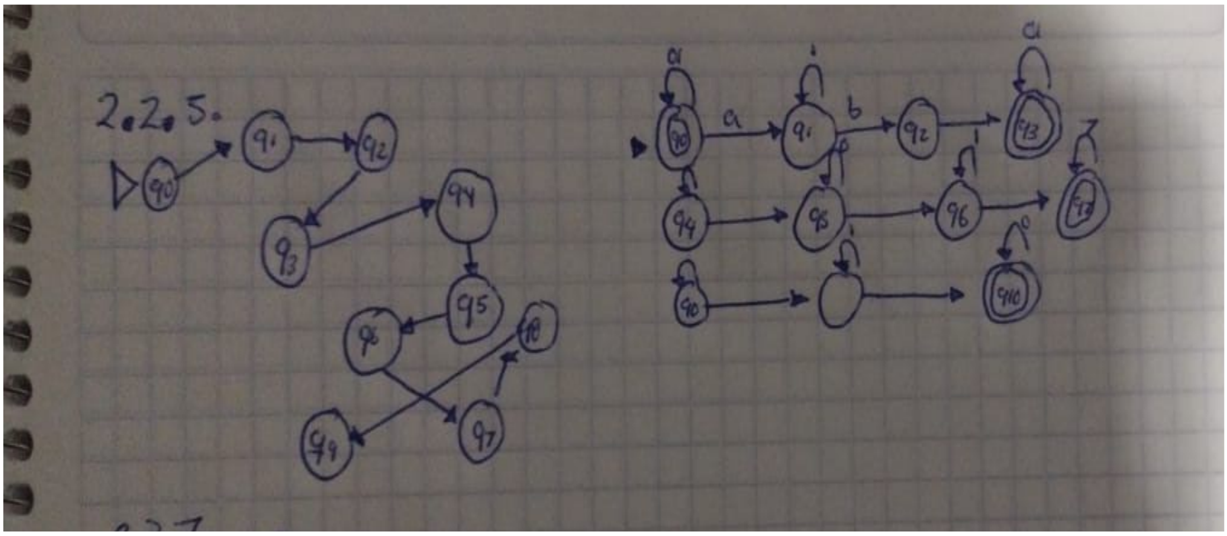
Describa los AFD que acepta los siguientes lenguajes con el alfabeto {0,1}:

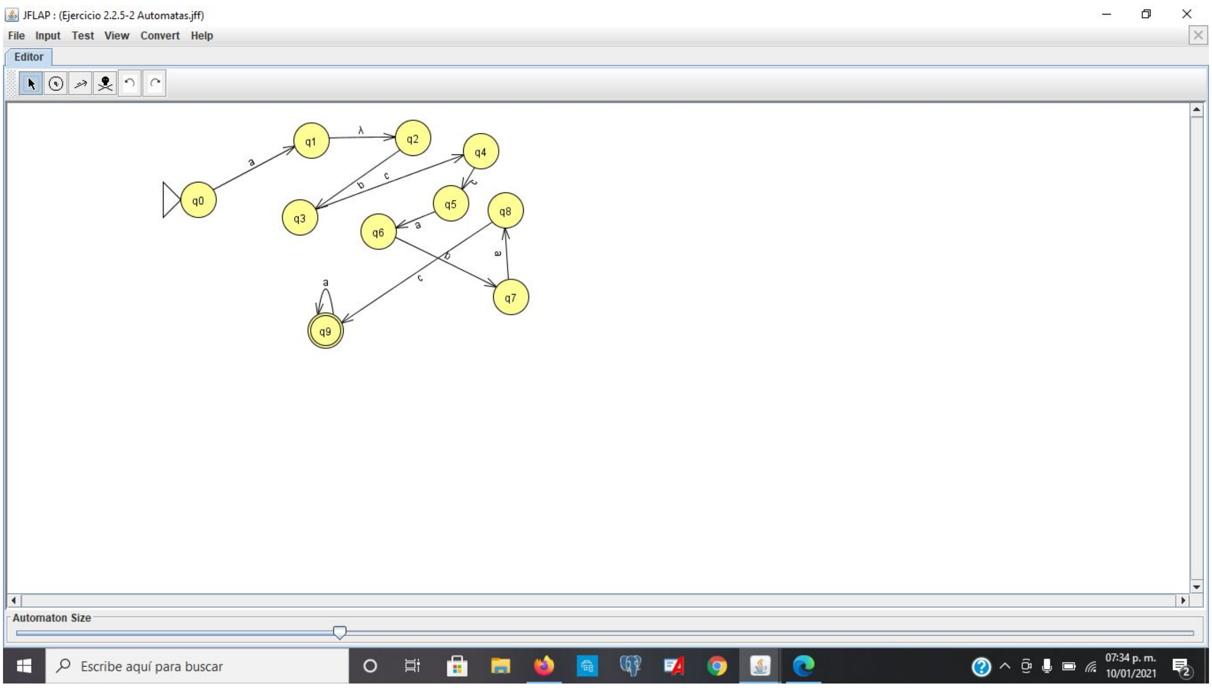
a) El conjunto de todas las cadenas tales que cada bloque de cinco símbolos consecutivos contenga al menos dos ceros.

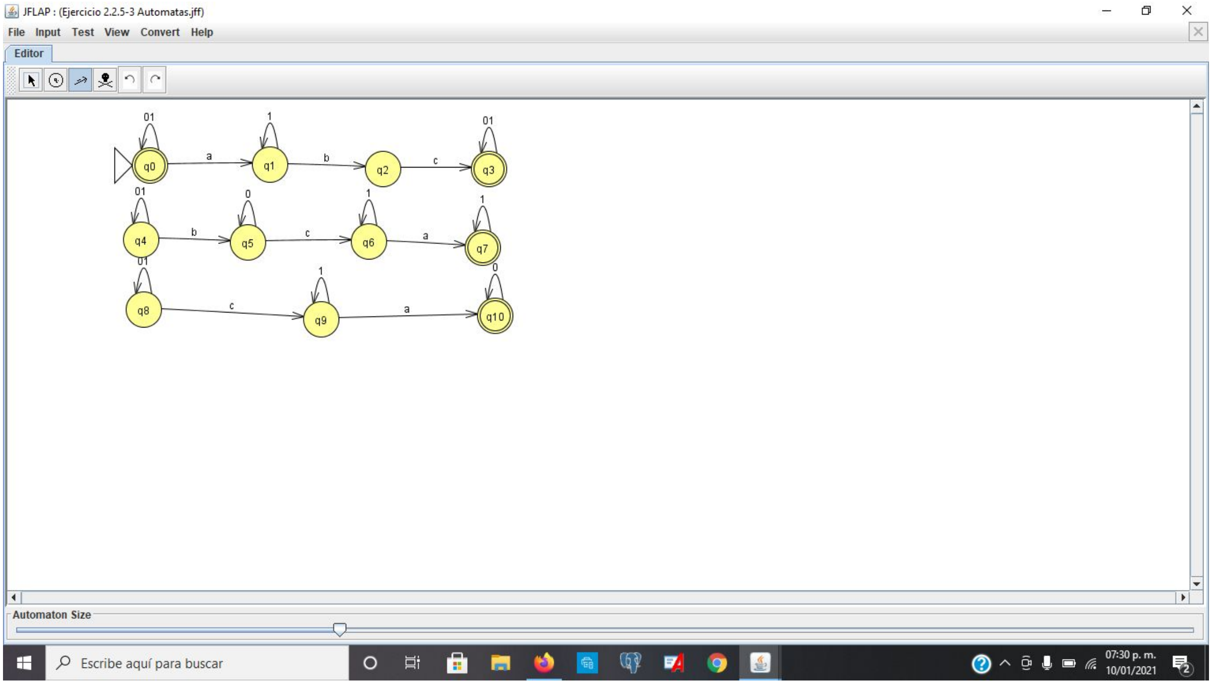
b) El conjunto de todas las cadenas cuyo símbolo está en la décima posición respecto del extremo derecho sea un 1.

c) El conjunto de cadenas que empiecen o terminen (o ambas cosas) con 01.

d) El conjunto de las cadenas tales que el número de ceros es divisible por cinco y el número de unos es divisible por 3.

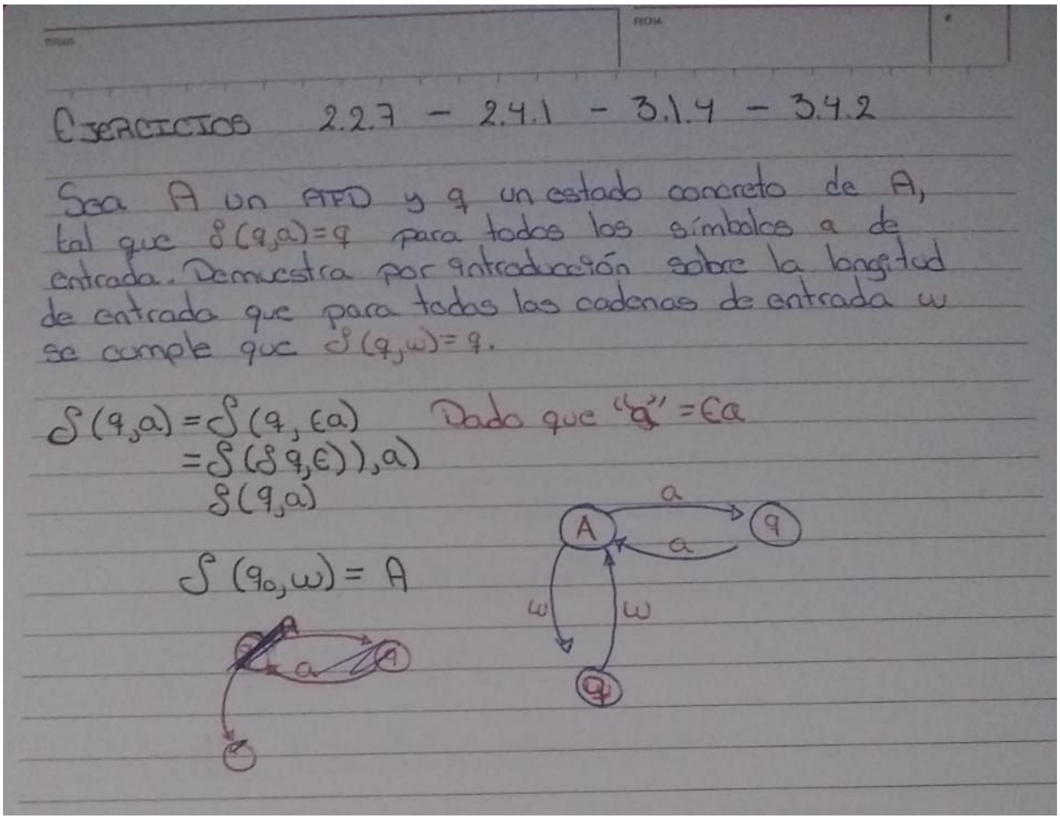






Ejercicio 2.2.7

Ejercicio 2.2.7. Sea A un AFD y q un estado concreto de A, tal que δ(q,a) = q para todos los símbolos a de entrada. Demuestre por inducción sobre la longitud de la entrada que para todas las cadenas de entrada w, se cumple que δ(q,w) = q.



Ejercicios 2.2.7 – 2.4.1 – 3.1.4 – 3.4.2

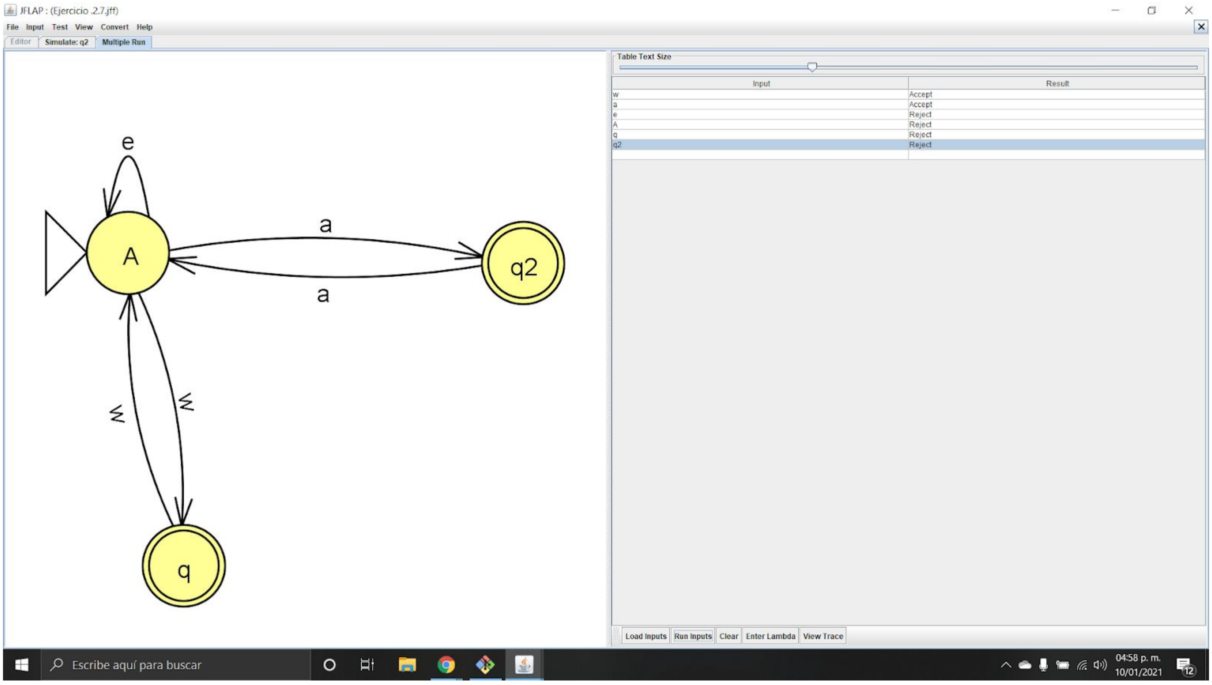
Sea A un AFD y q un estado concreto de A, tal que $\delta(q,a)=q$ para todos los símbolos a de entrada. Demuestra por introducción sobre la longitud de entrada que para todas las cadenas de entrada w se cumple que $\delta(q,w)=q$.

$\delta(q,a)=\delta(q,\epsilon a)$ Dado que "a" $=\epsilon a$

$=\delta(\delta(q,\epsilon)),a)$

$\delta(q,a)$

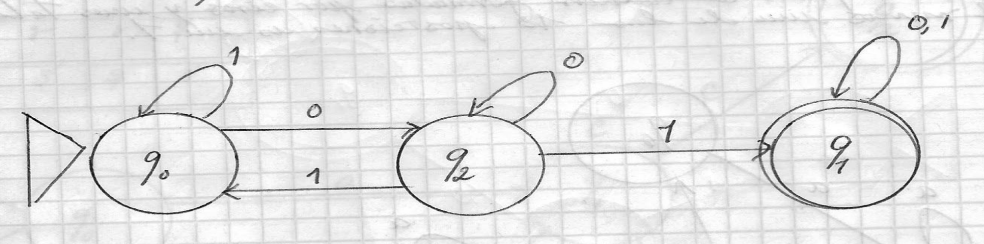
$\delta(q_0,w)=A$

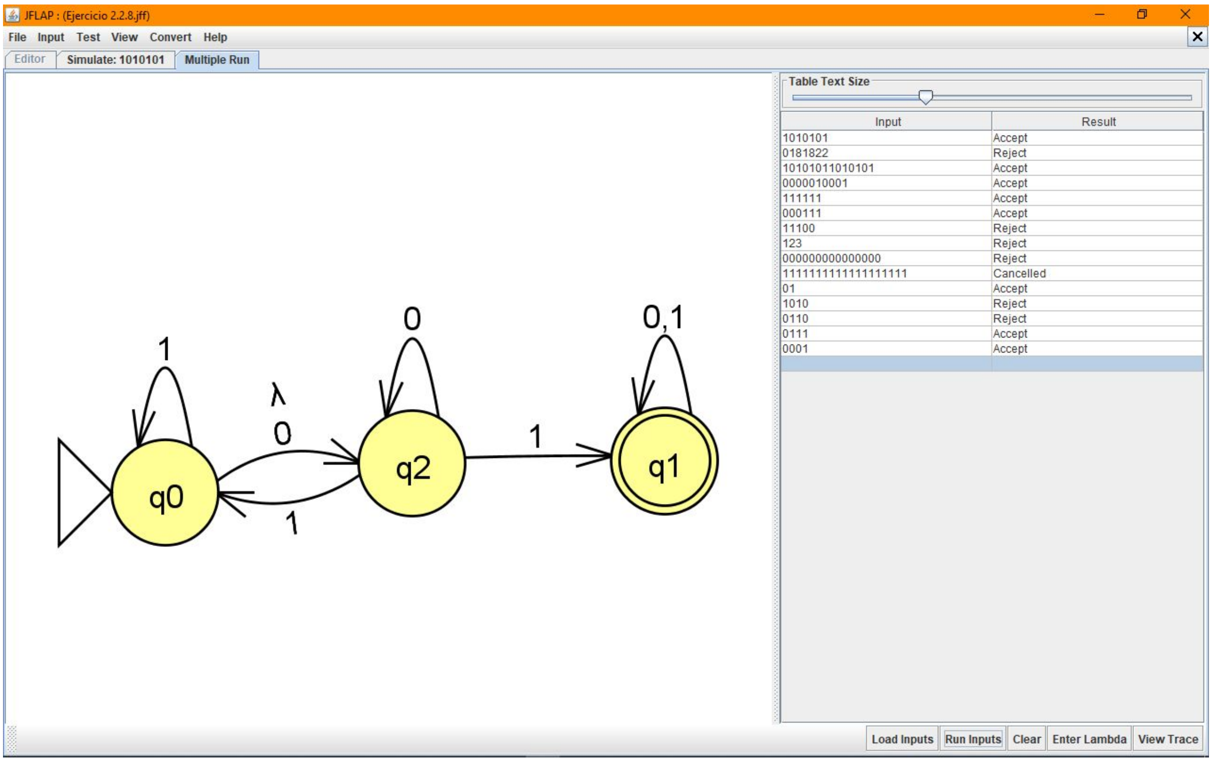


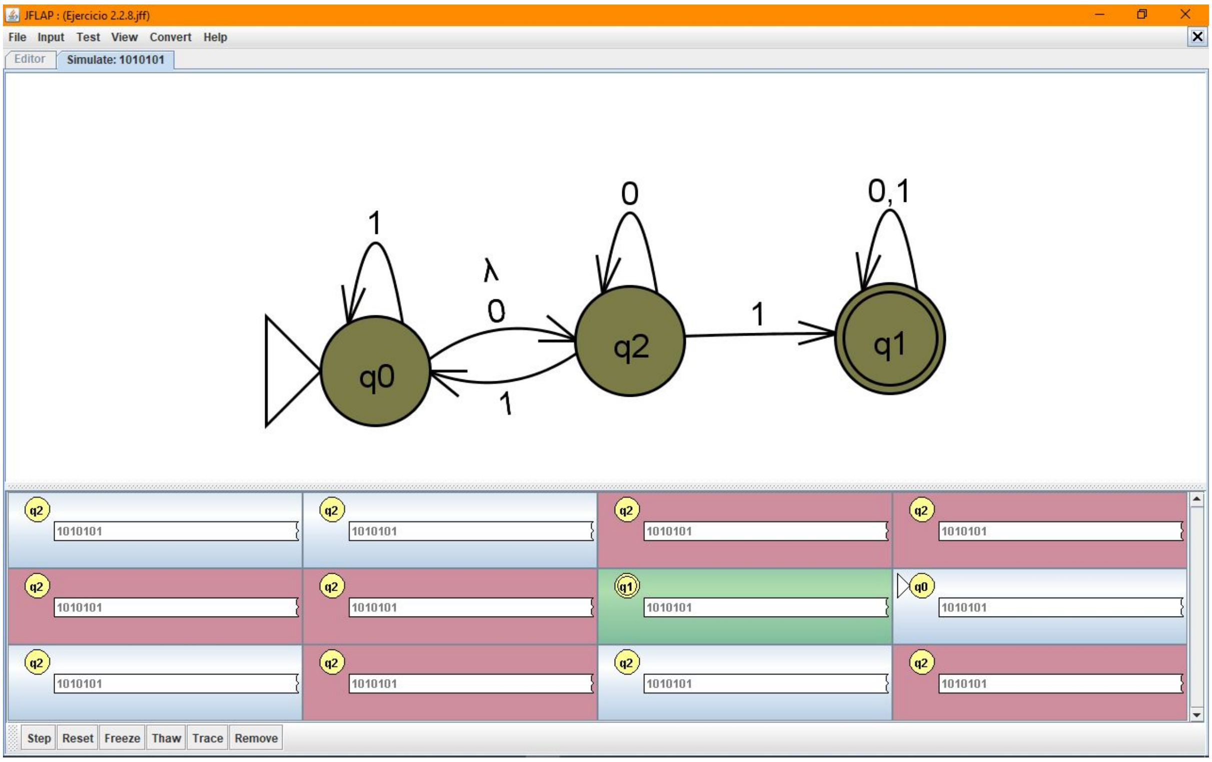
Ejercicio 2.2.8

Ejercicio 2.2.8. Sea A un AFD y a un símbolo de entrada particular de A, tal que para todos los estados q de
A tenemos que δ(q,a) = q.
a) Demuestre por inducción sobre n que para todo n ≥ 0, δ(q,an) = q, donde an es la cadena formada por n símbolos a.
b) Demuestre que {a}∗ ⊆ L(A) o {a}∗ ∩ L(A) = 0.

Ejercicio. 2.2.8.
Sea A un AFD y a un símbolo de entrada particular de A, tal que para todos los estados q de
A tenemos que δ(q,a) = q
a) Demuestre por inducción sobre n que para todo n ≥ 0, δ(q,an) = q, donde an es la cadena formada por n símbolos
b) Demuestre que {a}= ⊆ L(A) ó {a}= ∩ L(A) = 0.



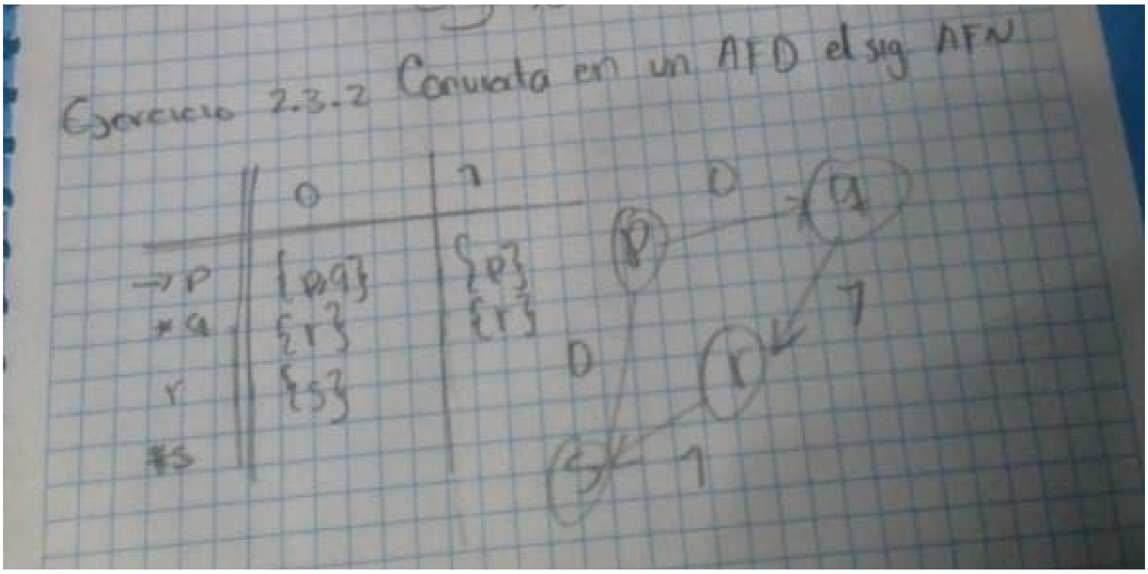

JFLAP : (Ejercicio 2.2.8.jff)
File Input Test View Convert Help
Editor
Simulate: 1010101
Multiple Run
Table Text Size
Input | Result
1010101 | Accept
0181822 | Reject
10101011010101 | Accept
0000010001 | Accept
111111 | Accept
000111 | Accept
11100 | Reject
123 | Reject
000000000000000 | Reject
1111111111111111111 | Cancelled
01 | Accept
1010 | Reject
0110 | Reject
0111 | Accept
0001 | Accept
q0
q2
q1
1
λ
0
1
0
1
0,1
Load Inputs
Run Inputs
Clear
Enter Lambda
View Trace

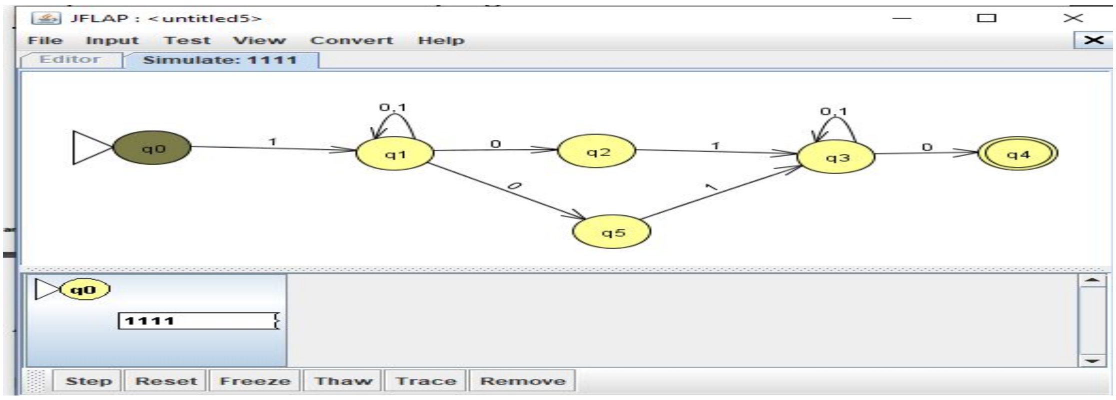

JFLAP : (Ejercicio 2.2.8.jff)
File Input Test View Convert Help
Editor
Simulate: 1010101
q0
q2
q1
1
λ
0
1
0
1
0,1
q2
1010101
q1
q0
Step
Reset
Freeze
Thaw
Trace
Remove

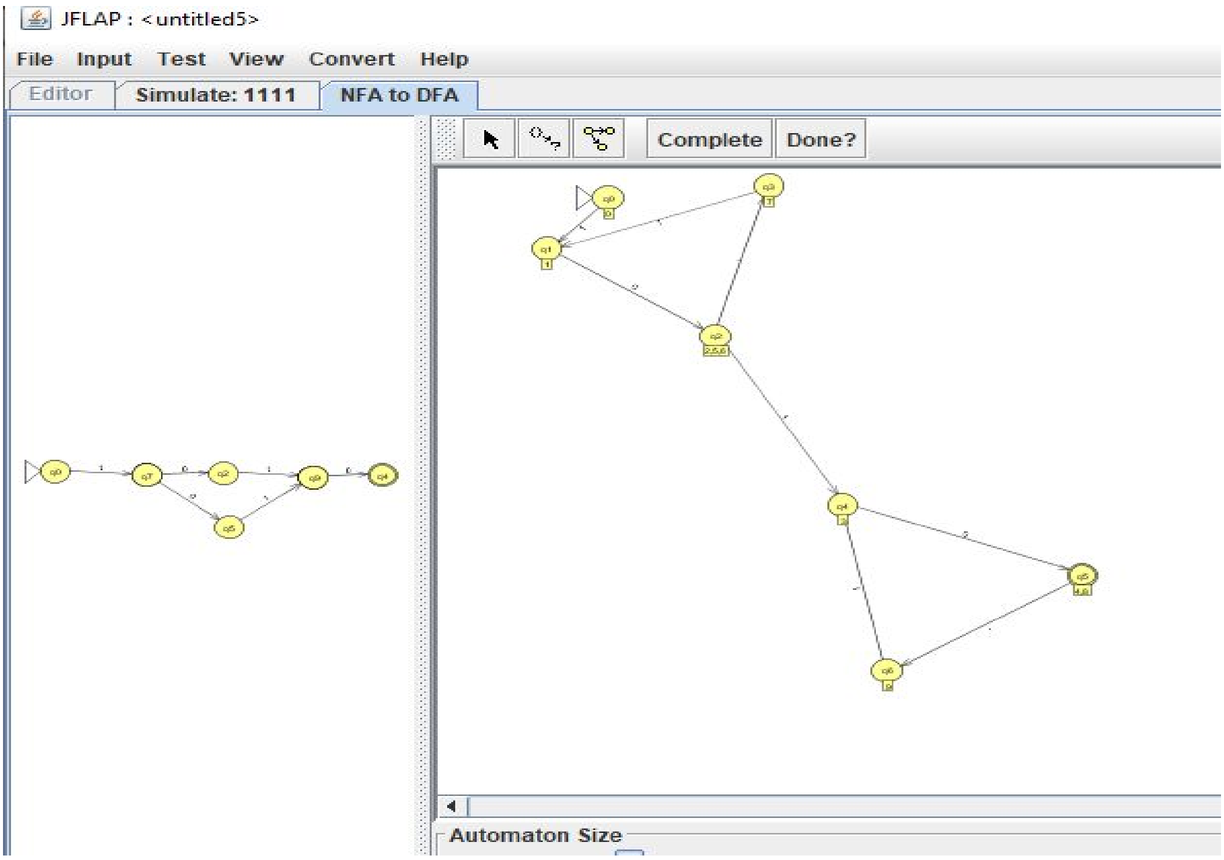
Ejercicio 2.3.2

Ejercicio 2.3.2. Convierta en un AFD el siguiente AFN:

	0	1
$\rightarrow p$	$\{q,s\}$	$\{q\}$
$*q$	$\{r\}$	$\{q,r\}$
r	$\{s\}$	$\{p\}$
$*s$	$\emptyset$	$\{p\}$

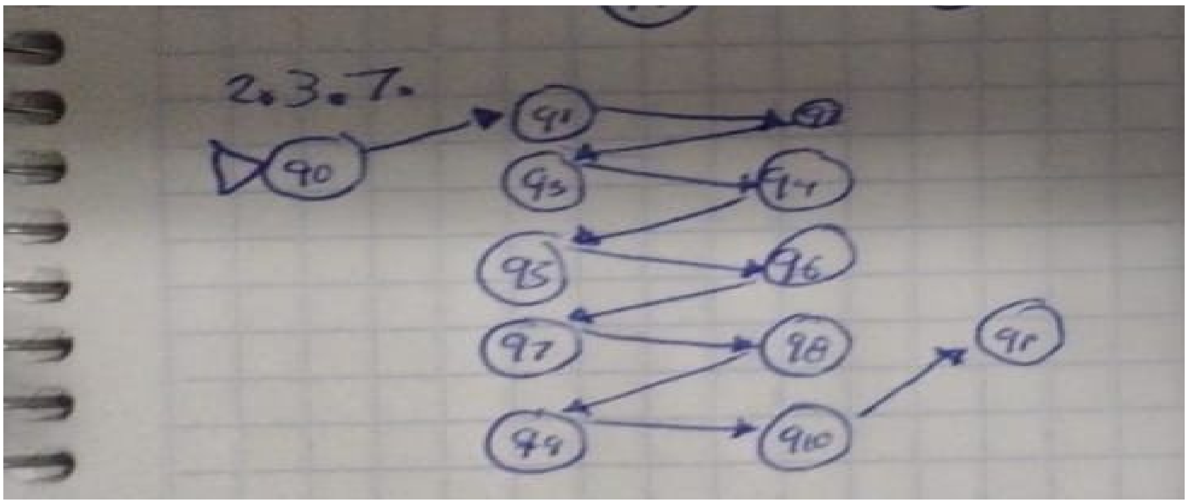


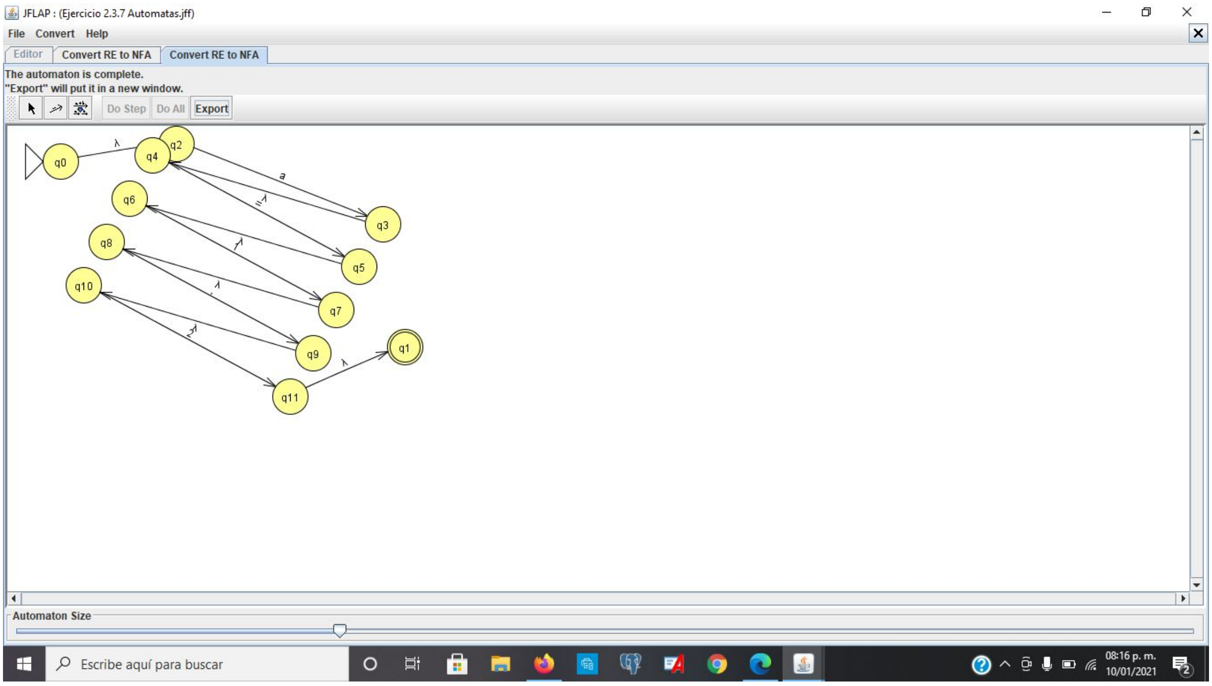




Ejercicio 2.3.7

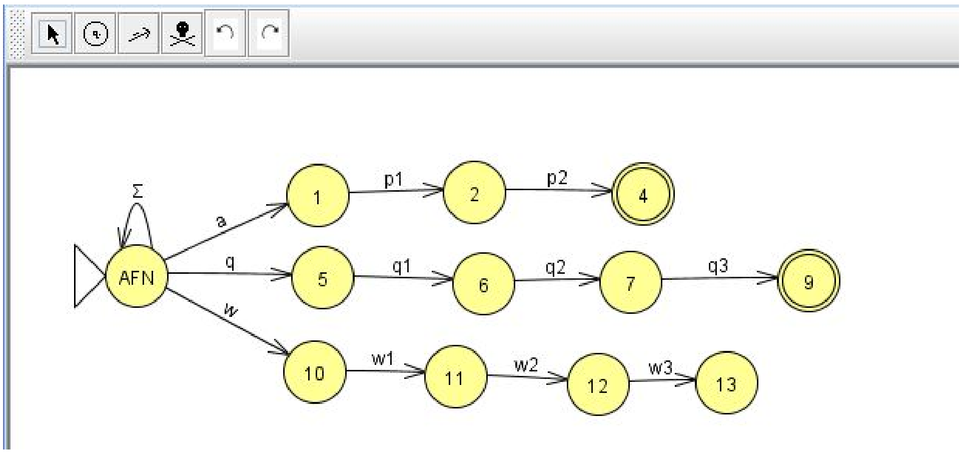
En el Ejemplo 2.13 hemos establecido que el AFN N se encuentra en el estado qi, para i = 1,2,...,n, después de leer la secuencia de entrada w si y sólo si el símbolo i-ésimo del final de w es 1. Demuestre esta afirmación.

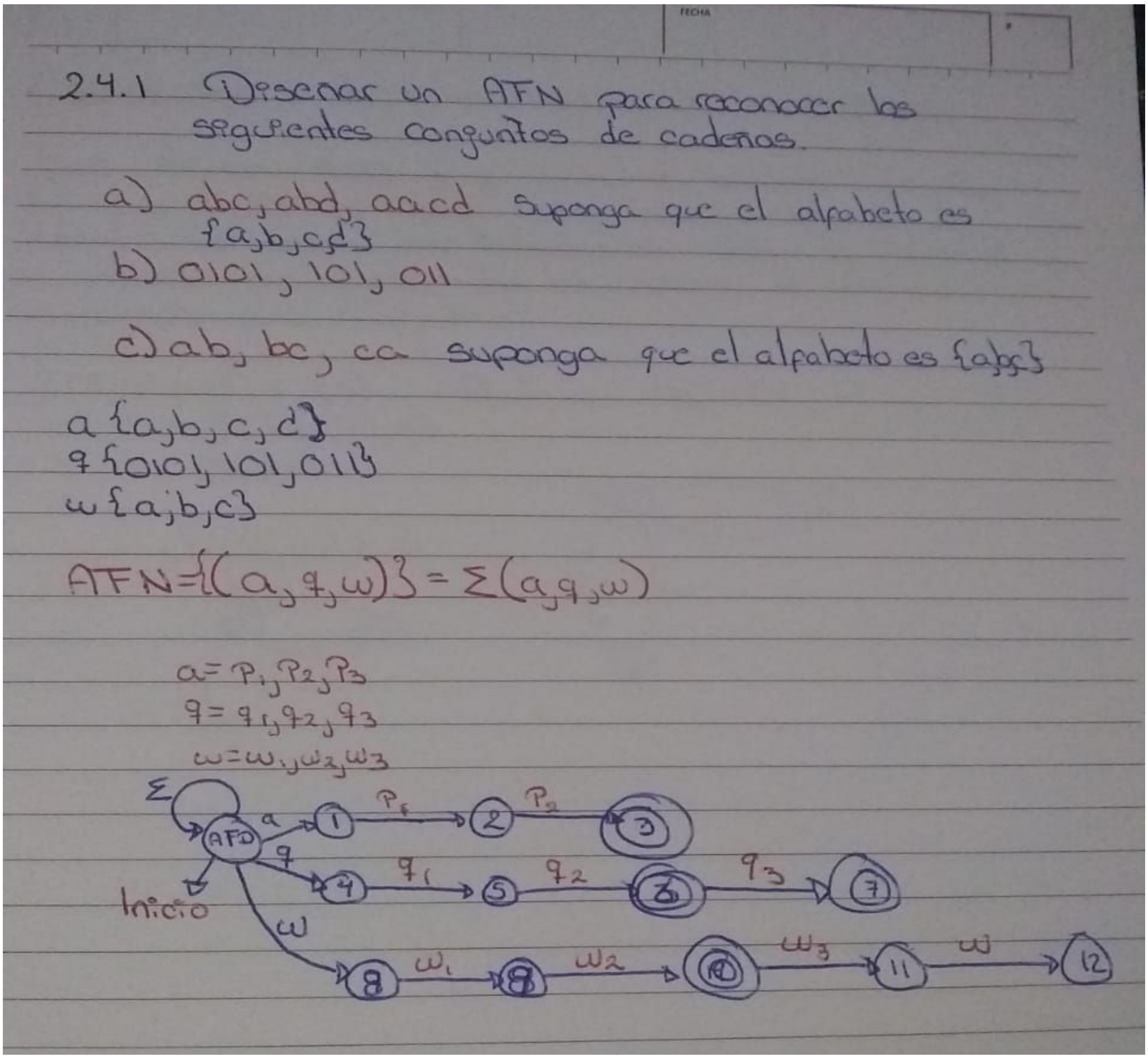




Ejercicio 2.4.1

Ejercicio 2.4.1. Diseñe un AFN para reconocer los siguienets conjuntos de cadenas.
a) abc, abd y aacd. Suponga que el alfabeto es {a,b,c,d}.
b) 0101, 101 y 011.
c) ab, bc y ca. Suponga que el alfabeto es {a,b,c}.





2.4.1 Diseñar un AFN para reconocer los siguientes conjuntos de cadenas.

a) abc, abd, aacd Suponga que el alfabeto es {a,b,c,d}

b) 0101, 101, 011

c) ab, bc, ca suponga que el alfabeto es {a,b,c}

a {a,b,c,d}
q {0101, 101, 011}
ω {a,b,c}

AFN={(a, q, ω)} = Σ(a,q,ω)

a= P1, P2, P3
q = q1, q2, q3
ω=ω1, ω2, ω3

Ejercicio 2.5.3

Ejercicio 2.5.3. Diseñe un AFN-ε para cada uno de los siguientes lenguajes. Intente emplear transiciones-ε para simplificar su diseño.

a) El conjunto de cadenas formado por cero o más letras a seguidas de cero o más letras b, seguida de cero o más letras c.

! b) El conjunto de cadenas que constan de la subcadena 01 repetida una o más veces o de la subcadena 010 repetida una o más veces.

! c) El conjunto de cadenas formadas por ceros y unos tales que al menos una de las diez posiciones es un 1.

1.

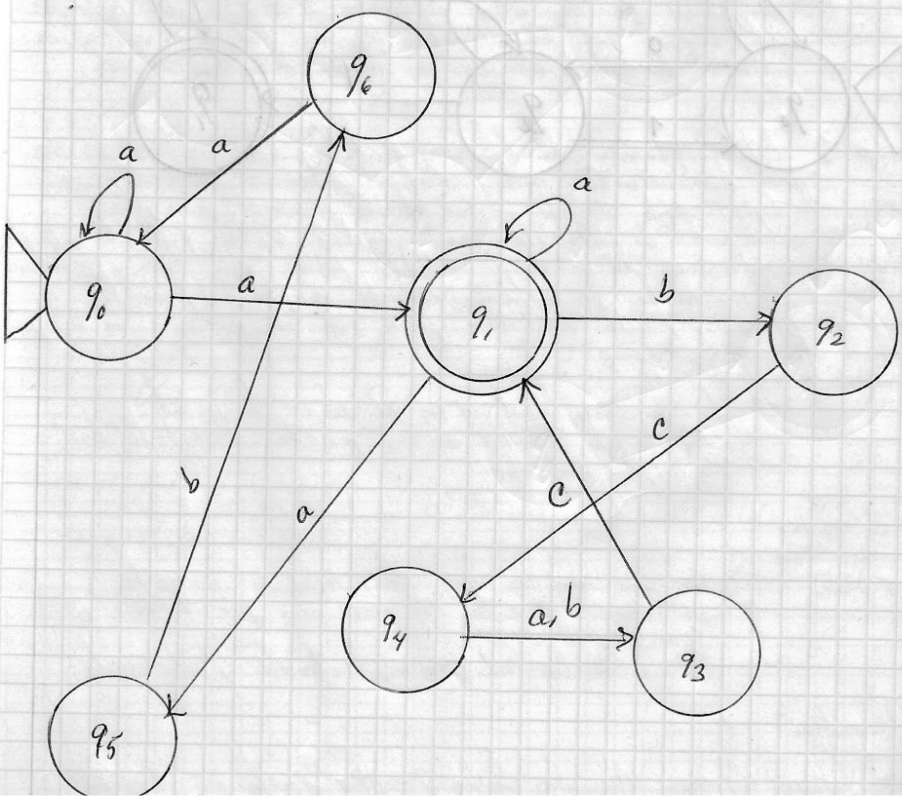
Ejercicio 2,5,3.

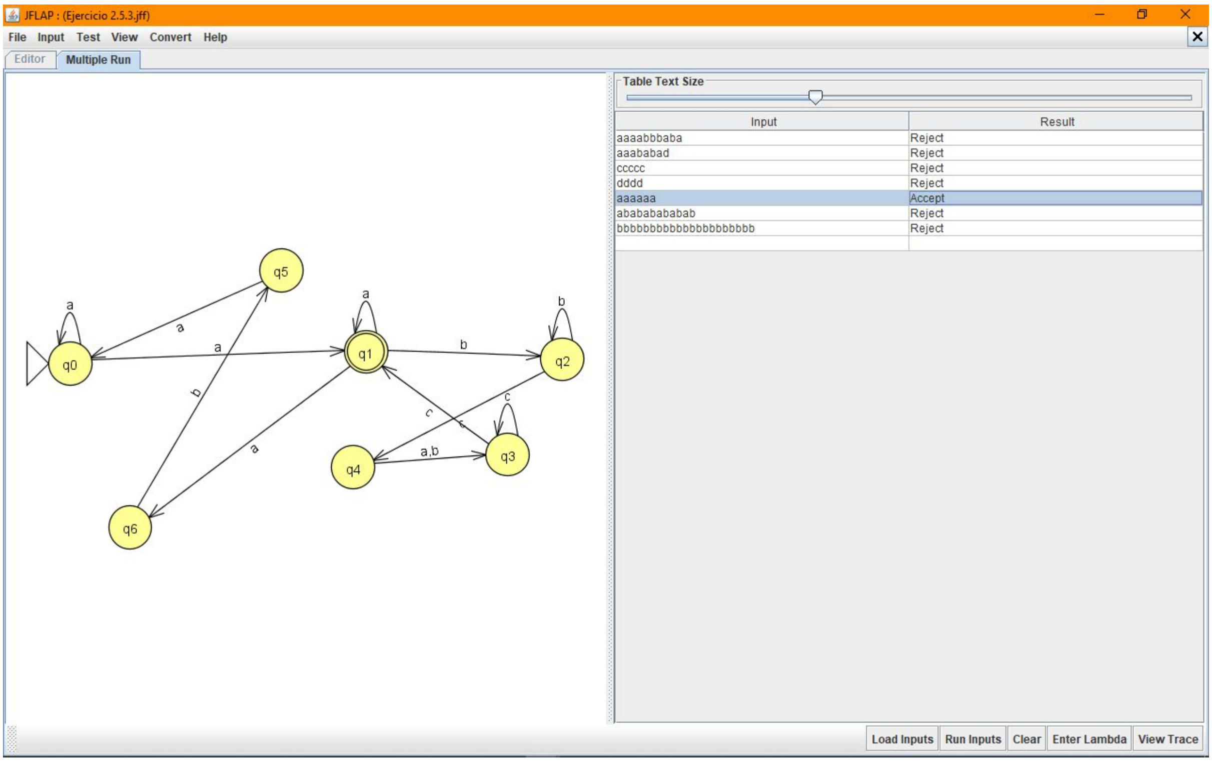
Diseñe un AFN para cada uno de los lenguajes. Intente emplear transicciones para simplificar su diseño.

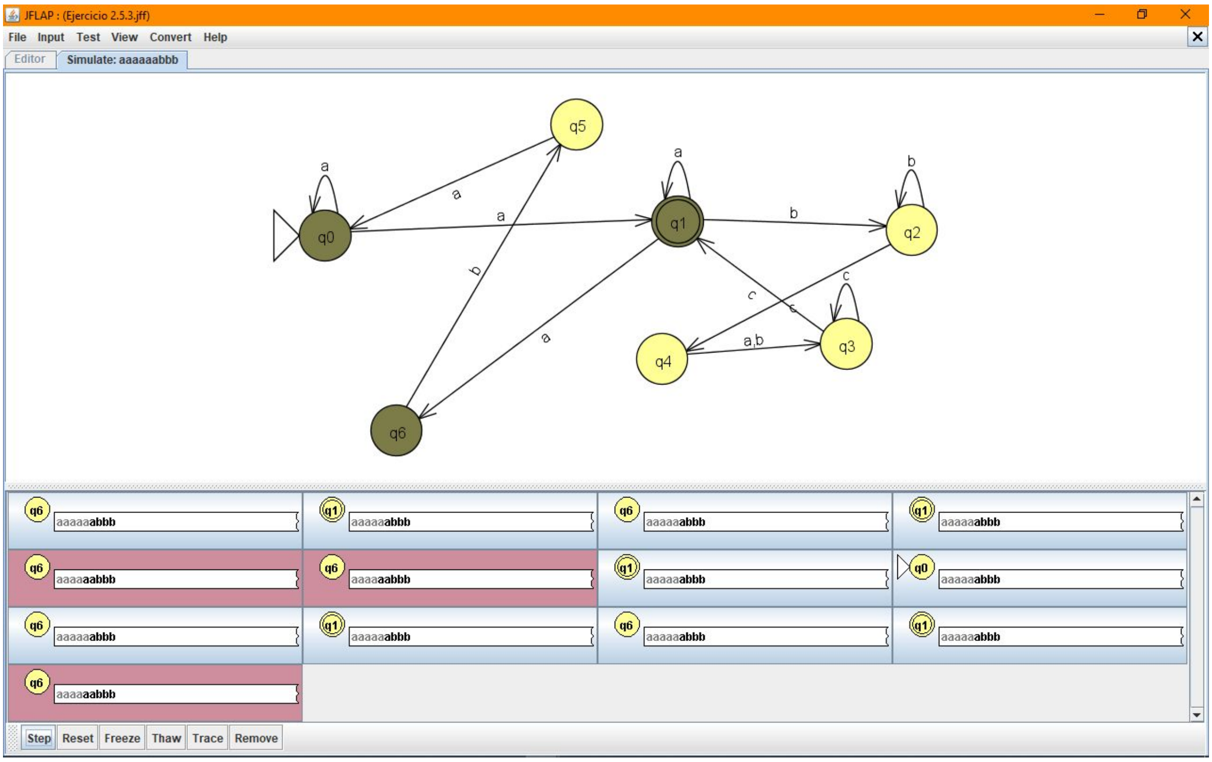
a) El conjunto de cadenas formadas por cero ó mas letras a seguidas de cero ó más letras b, seguida de cero, ó más letras c.

b) El conjunto de cadenas que consten de la subcadena 01 repetida una ó más veces ó de la cadena 010 repetida una ó más veces.

c) El conjunto de cadenas formadas por ceros y unos tales que al menos una de las 10 posiciones es un 1.




JFLAP : (Ejercicio 2.5.3.jff)
File Input Test View Convert Help
Editor
Multiple Run
Table Text Size
Input
Result
aaaabbbaba Reject
aaababad Reject
ccccc Reject
dddd Reject
aaaaaa Accept
abababababab Reject
bbbbbbbbbbbbbbbbbbbb Reject
q0 q1 q2 q3 q4 q5 q6
Load Inputs
Run Inputs
Clear
Enter Lambda
View Trace


JFLAP : (Ejercicio 2.5.3.jff)
File Input Test View Convert Help
Editor
Simulate: aaaaaabbb
q0 q1 q2 q3 q4 q5 q6
Step
Reset
Freeze
Thaw
Trace
Remove

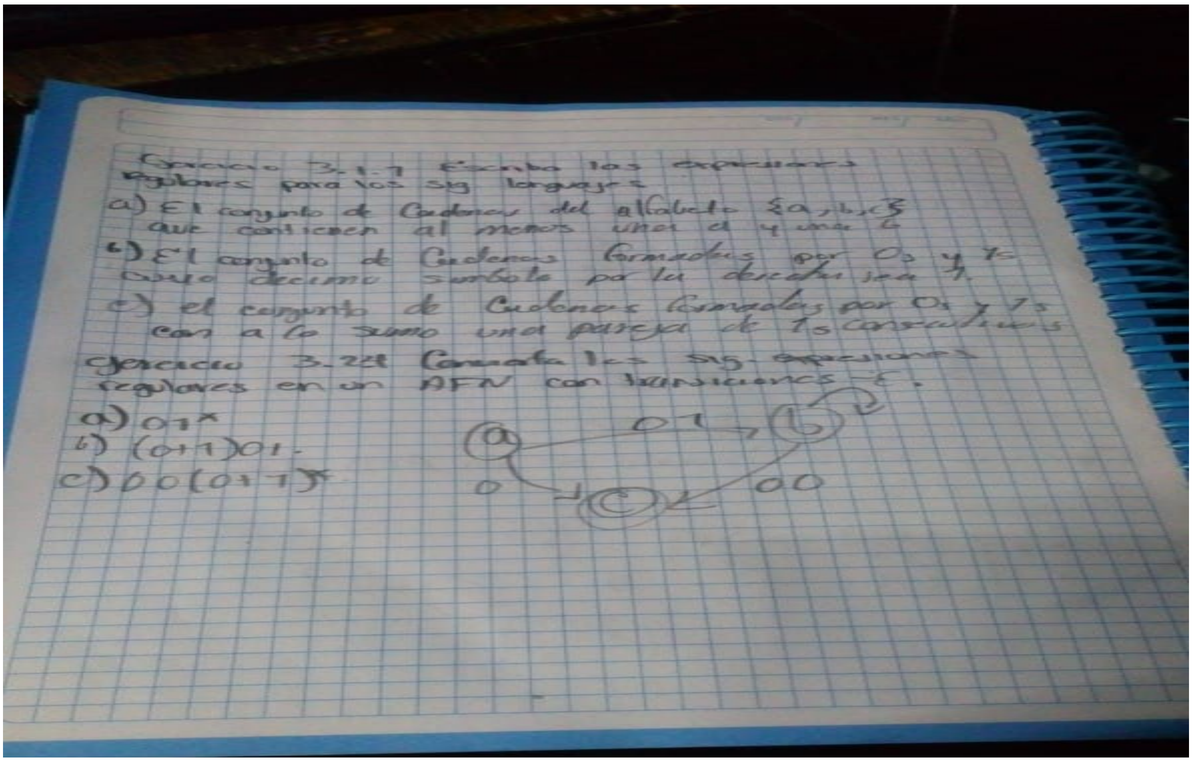
Ejercicio 3.1.1

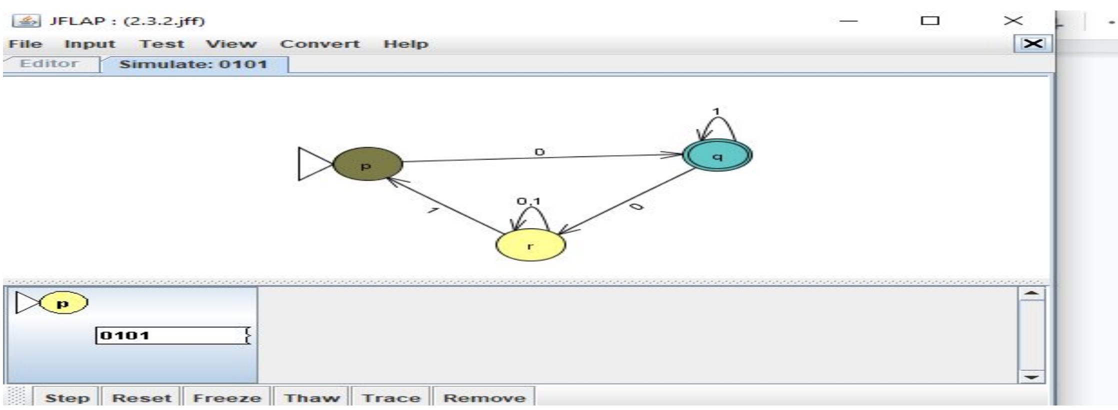
Ejercicio 3.1.1. Escriba expresiones regulares para los siguientes lenguajes:

a) El conjunto de cadenas del alfabeto {a,b,c} que contienen al menos una a y al menos una b.

b) El conjunto de cadenas formadas por 0s y 1s cuyo décimo símbolo por la derecha sea 1.

c) El conjunto de cadenas formadas por 0s y 1s con a lo sumo una pareja de 1s consecutivos.



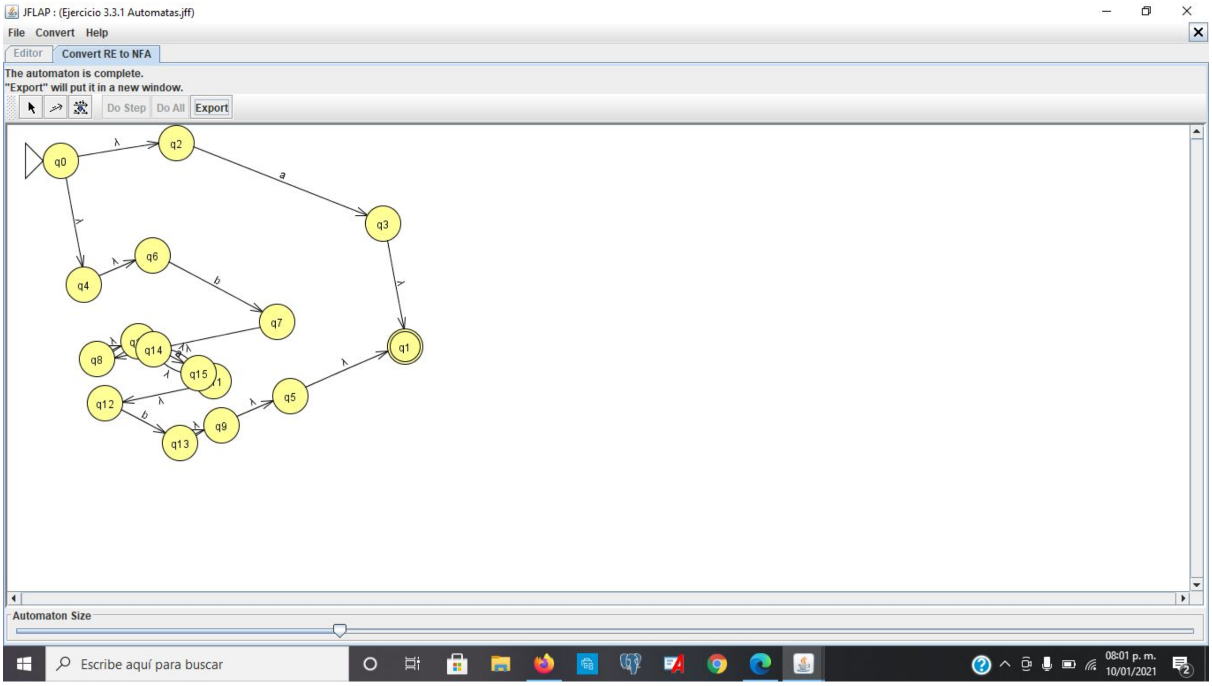


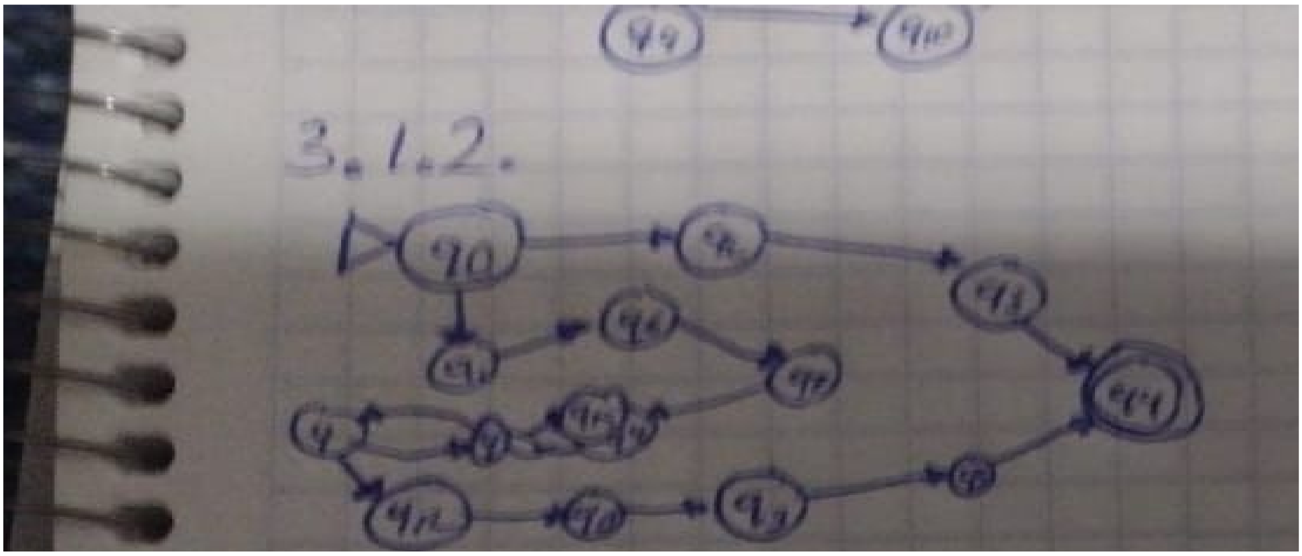
Ejercicio 3.1.2

Escriba expresiones regulares para los siguientes lenguajes: *

a) El conjunto de todas las cadenas formadas por ceros y unos tales que cada pareja de 0s adyacentes aparece antes que cualquier pareja de 1s adyacentes.

b) El conjunto de cadenas formadas por ceros y unos cuyo número de ceros es divisible por cinco.





Ejercicio 3.1.4

Ejercicio 3.1.4. Proporcione las descripciones informales de los lenguajes correspondientes a las siguientes expresiones regulares: *

a) (1+ε)(00∗1)∗0∗.

b) (0∗1∗)∗000(0+1)∗. c) (0+10)∗1∗.

3.1.4 Proporcione las descripciones informales de los lenguajes correspondientes a los siguientes expresiones regulares.

a) (1+E)(00* 1)*0*.

b) (0*1*)*000(0+1)*.

c) (0+10)*1*

a)((1)+(00)(*1)(*0*))E

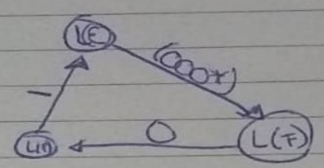
b) (0*)(1*) (000)* (0)+(1)*

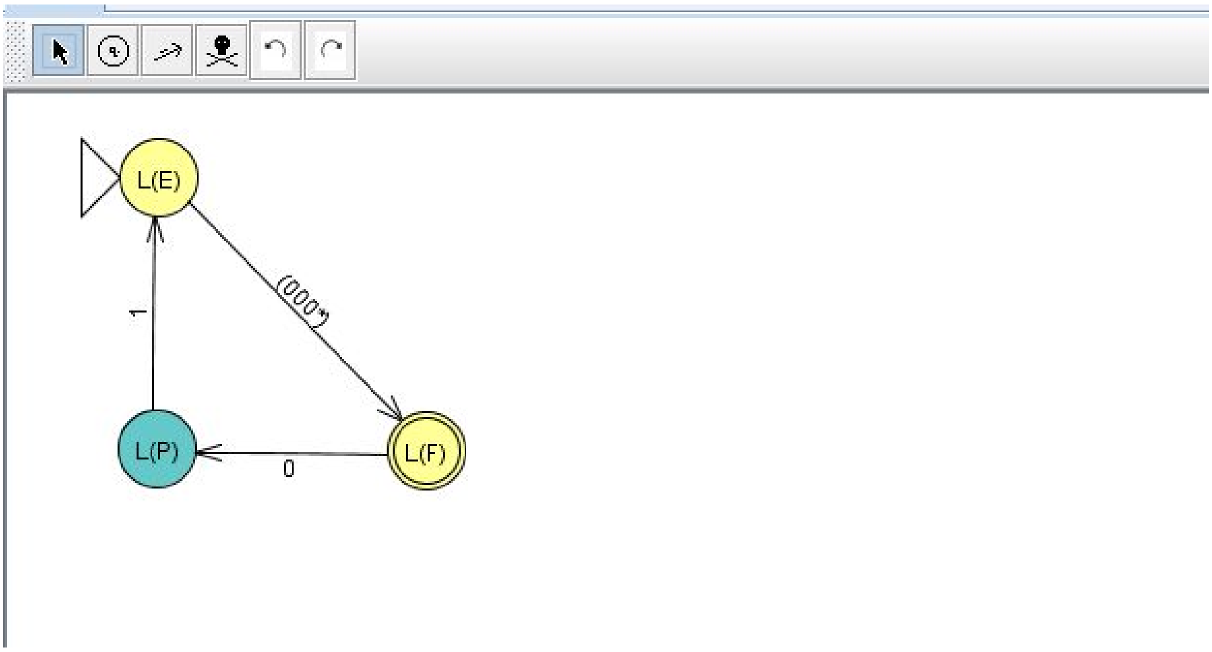
c) (0)+(10*)(1*)

a) ~~#E~~ L(E)(1)(00*)(1*)(0*)

b) L(F)=(0*)(1*)(00)(0*)(0)+(1*)

c) L(P)=(0*)(00)(0*)+(1)(1*)

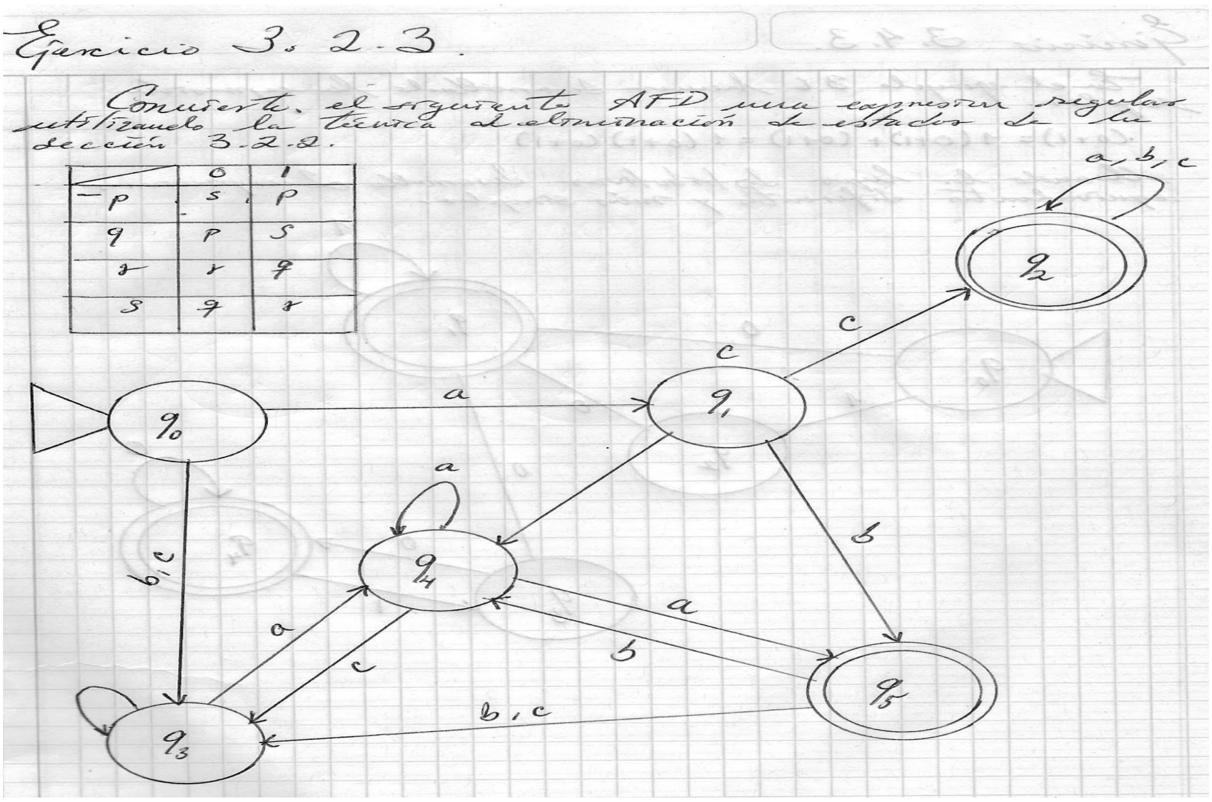




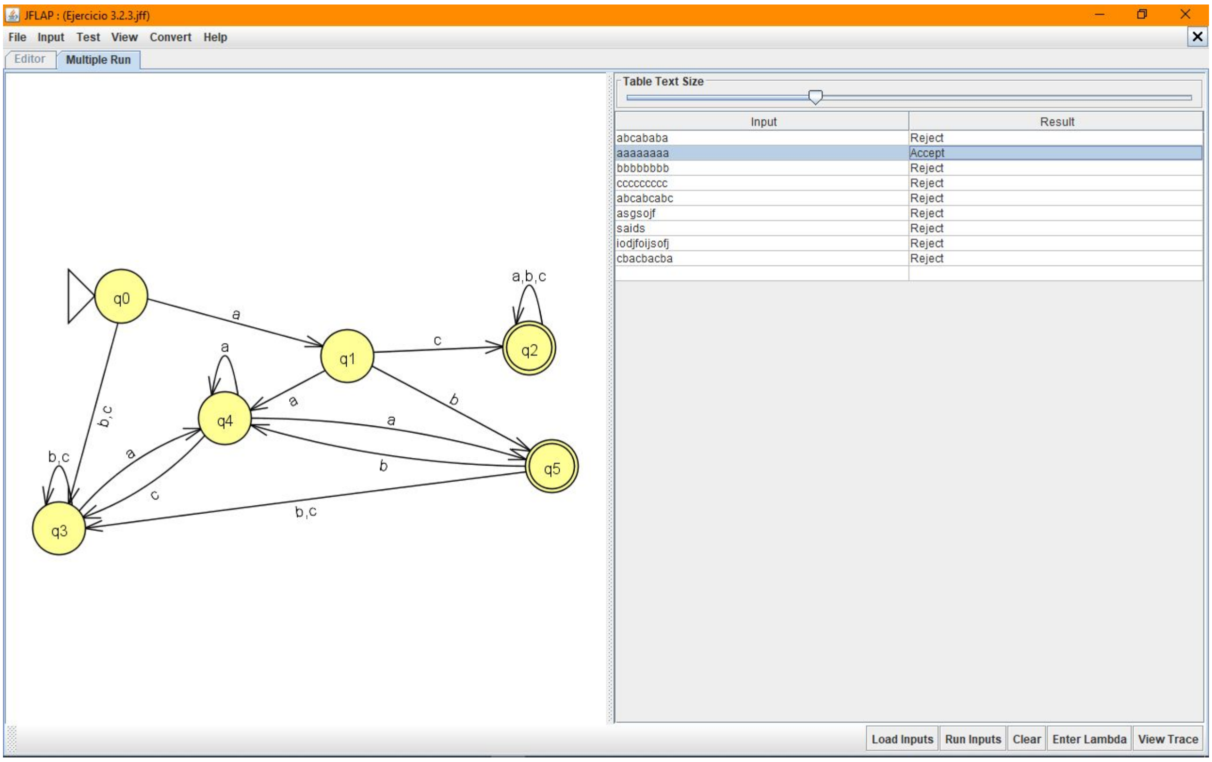
Ejercicio 3.2.3

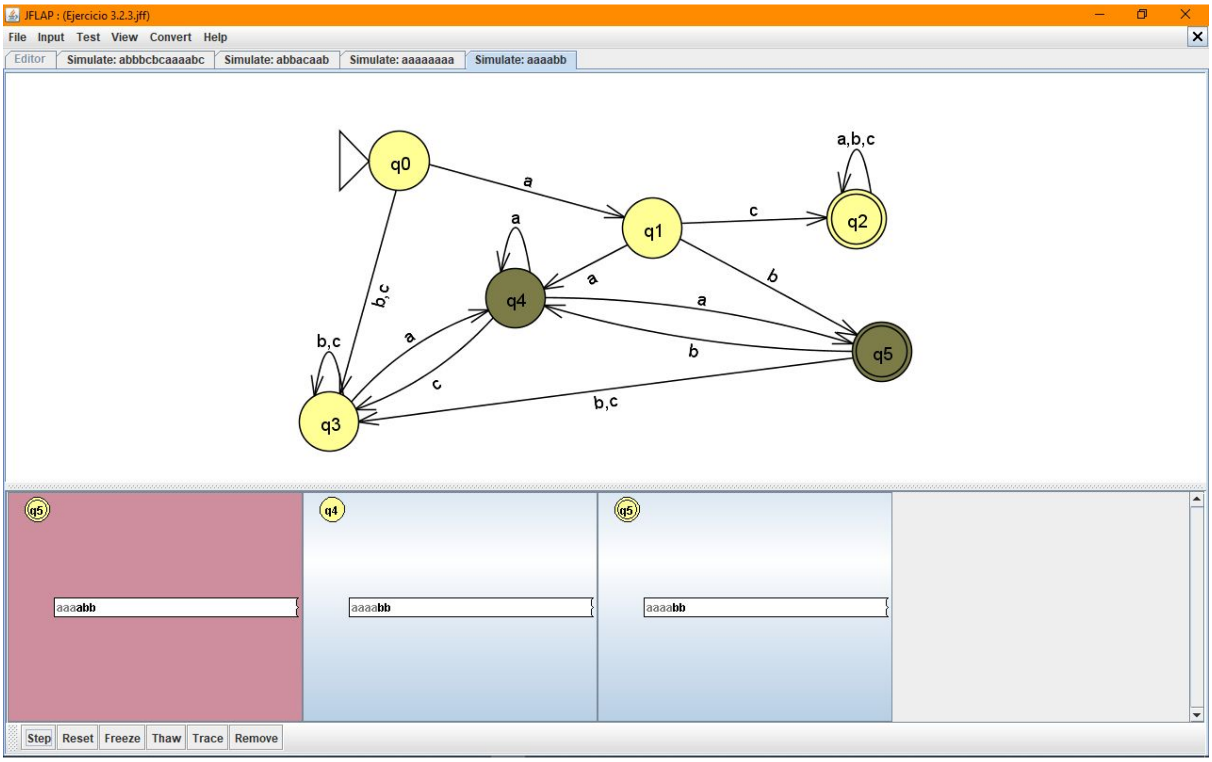
Ejercicio 3.2.3. Convierta el siguiente AFD en una expresión regular utilizando la técnica de eliminación de estados de la Sección 3.2.2.

	0	1
→*p	s	p
q	p	s
r	r	q



s q r


JFLAP : (Ejercicio 3.2.3.jff)
File Input Test View Convert Help
Editor
Multiple Run
Table Text Size
Input
Result
abcababa Reject
aaaaaaaa Accept
bbbbbbbb Reject
ccccccccc Reject
abcabcabc Reject
asgsojf Reject
saids Reject
iodjfoijsofj Reject
cbacbacba Reject
Load Inputs
Run Inputs
Clear
Enter Lambda
View Trace
q0
q1
q2
q3
q4
q5
a,b,c
b,c


JFLAP : (Ejercicio 3.2.3.jff)
File Input Test View Convert Help
Editor
Simulate: abbbcbcaaaabc
Simulate: abbacaab
Simulate: aaaaaaaa
Simulate: aaaabb
q0
q1
q2
q3
q4
q5
a,b,c
b,c
aaaabb
aaaabb
aaaabb
Step
Reset
Freeze
Thaw
Trace
Remove

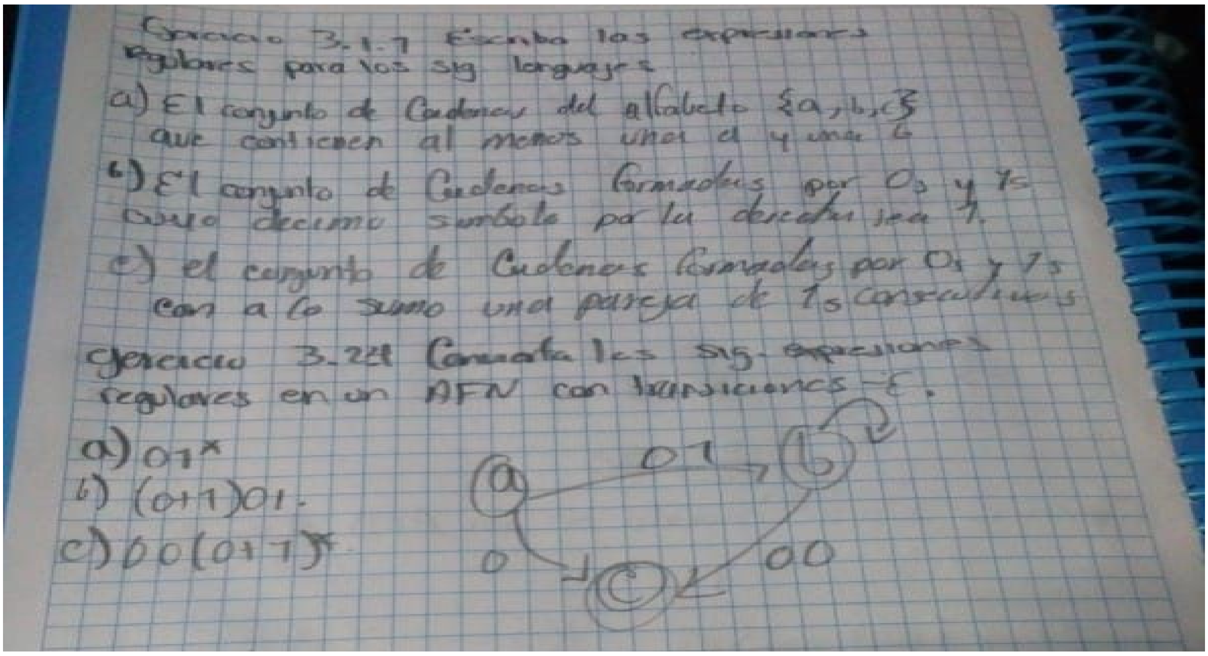
Ejercicio 3.2.4

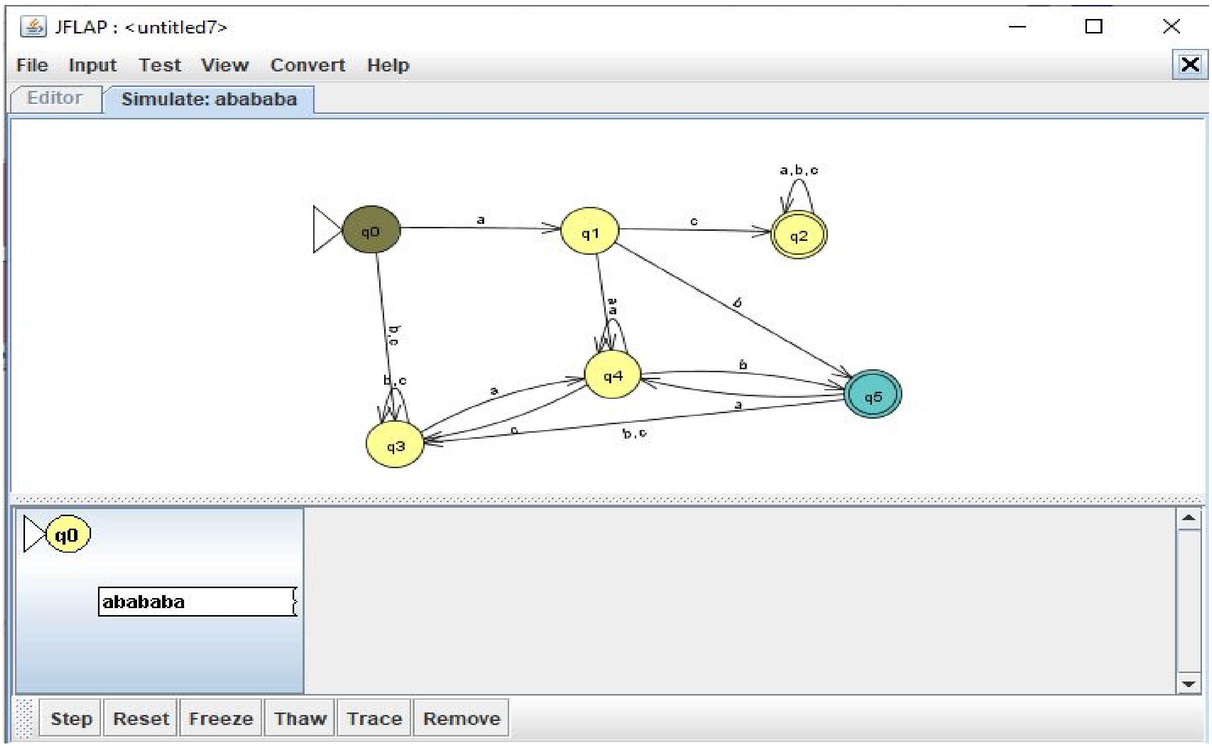
Ejercicio 3.2.4. Convierta las siguientes expresiones regulares en un AFN con transiciones-ε. *

a) 01∗.

b) (0+1)01.

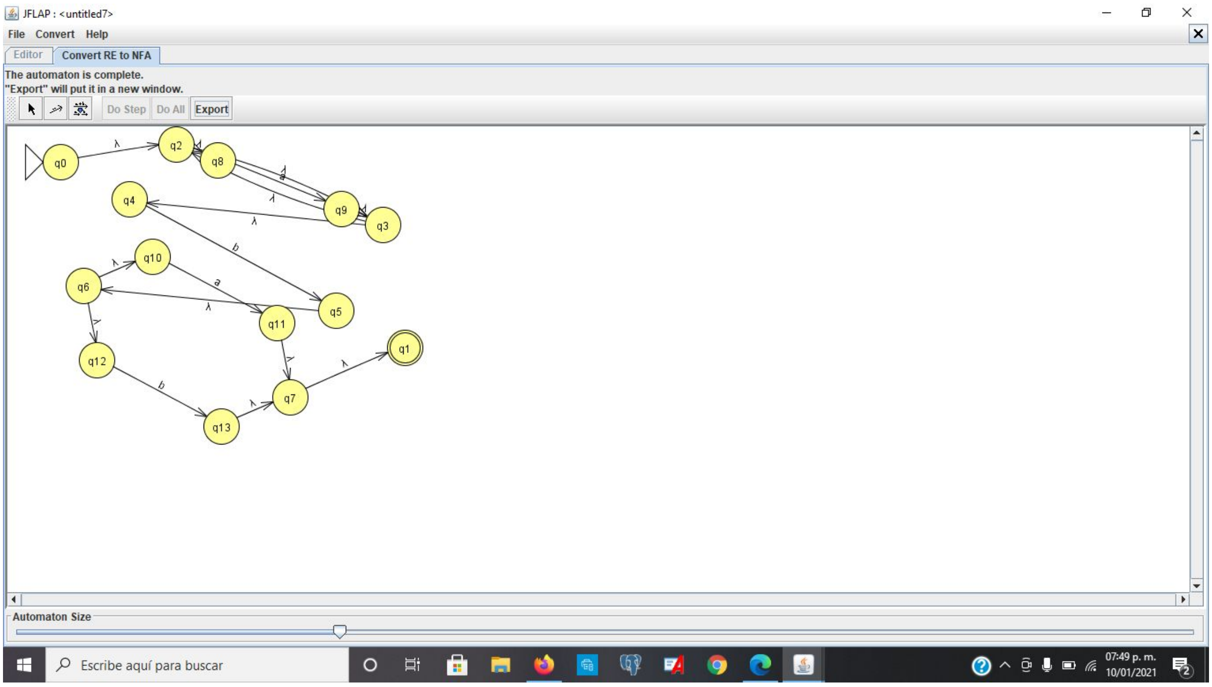
c) 00(0+1)∗.

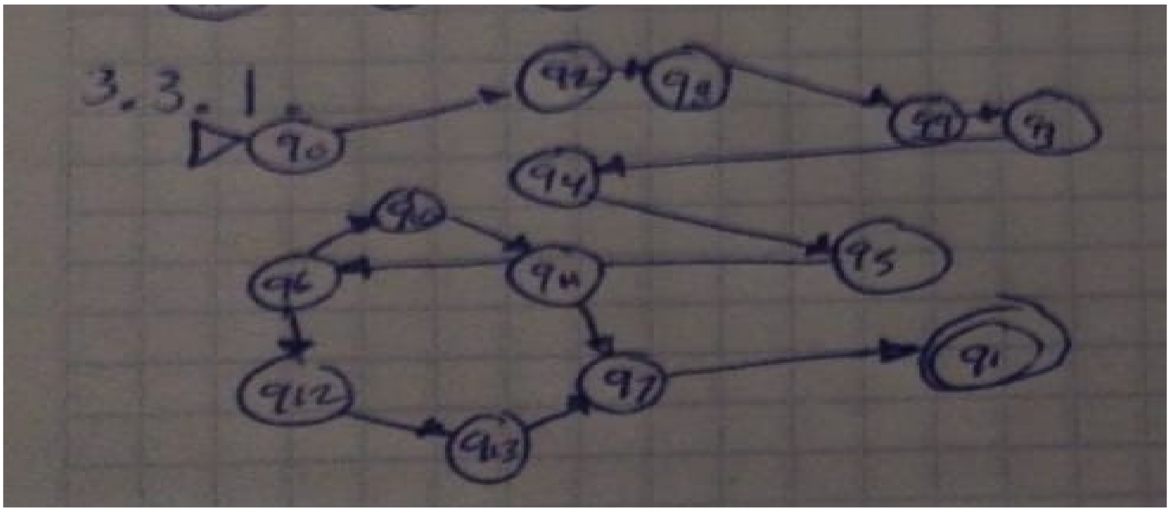




Ejercicio 3.3.1

Obtenga una expresión regular que describa números de télefono en todas las variantes que pueda imaginar. Tenga en cuenta los números internacionales así como el hecho de que los distintos países emplean una cantidad diferente de dígitos para los códigos de área de los números de teléfono locales.





Ejercicio 3.4.2

Ejercicio 3.4.2. Demuestre si cada una de las siguientes proposiciones acerca de expresiones regulares es verdadera o falsa.

a) $(R+S)* = R* + S*$.

b) $(RS+R)*R = R(SR+R)*$. *

c) $(RS+R)*RS = (RR*S)*$.
d) $(R+S)*S = (R*S)*$.
e) $S(RS+S)*R = RR*S(RR*S)*$.

3.4.2 Demuestre si cada una de las siguientes proposiciones acerca de expresiones regulares es verdadera o falsa

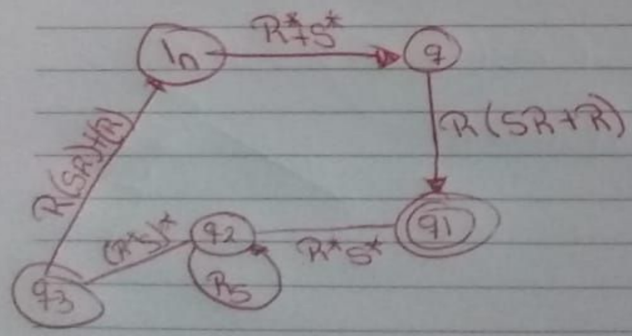
a) $(R+S)^* = R^* + S^*$
b) $(RS+R)^*R = R(SR+R)^*$
c) $(RS+R)^*RS = (RR^*S)^*$
d) $(R+S)^*S = (R^*S)^*$
e) ~~SS~~ $S(RS+S)^*R = RR^*S(RR^*S)^*$

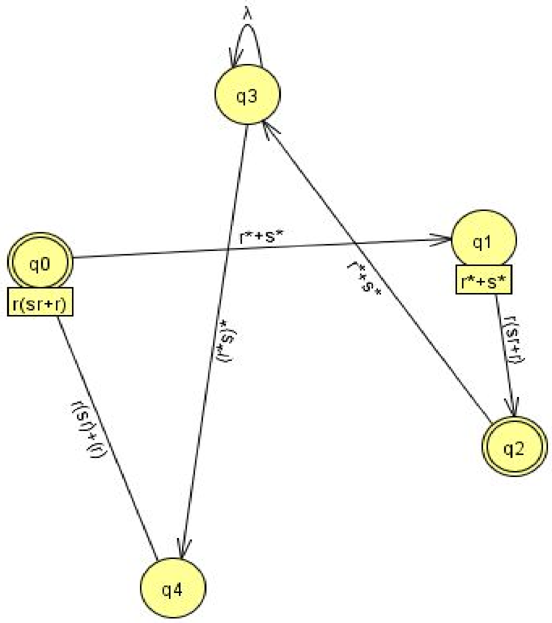
a) $R^* + S^*$

b) $R(SR)+(R))$

c) $(RR^*S)^*$

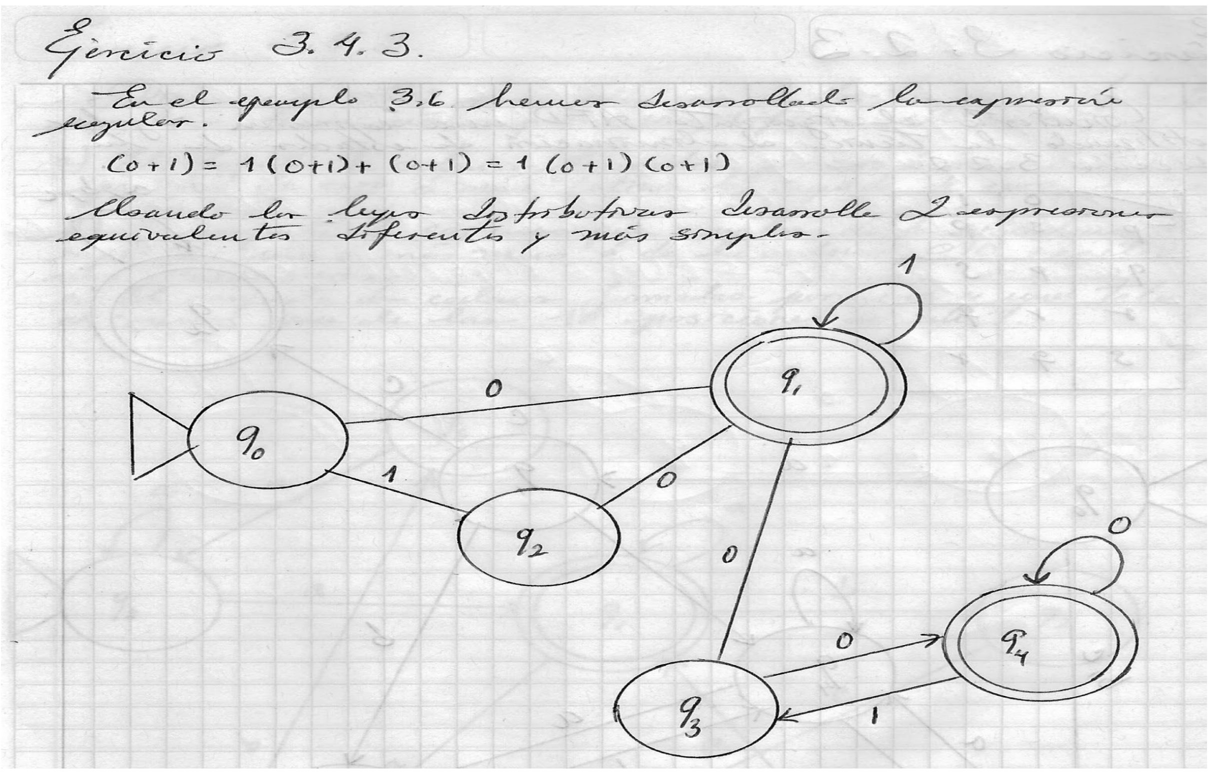
d) $R^* S^*$

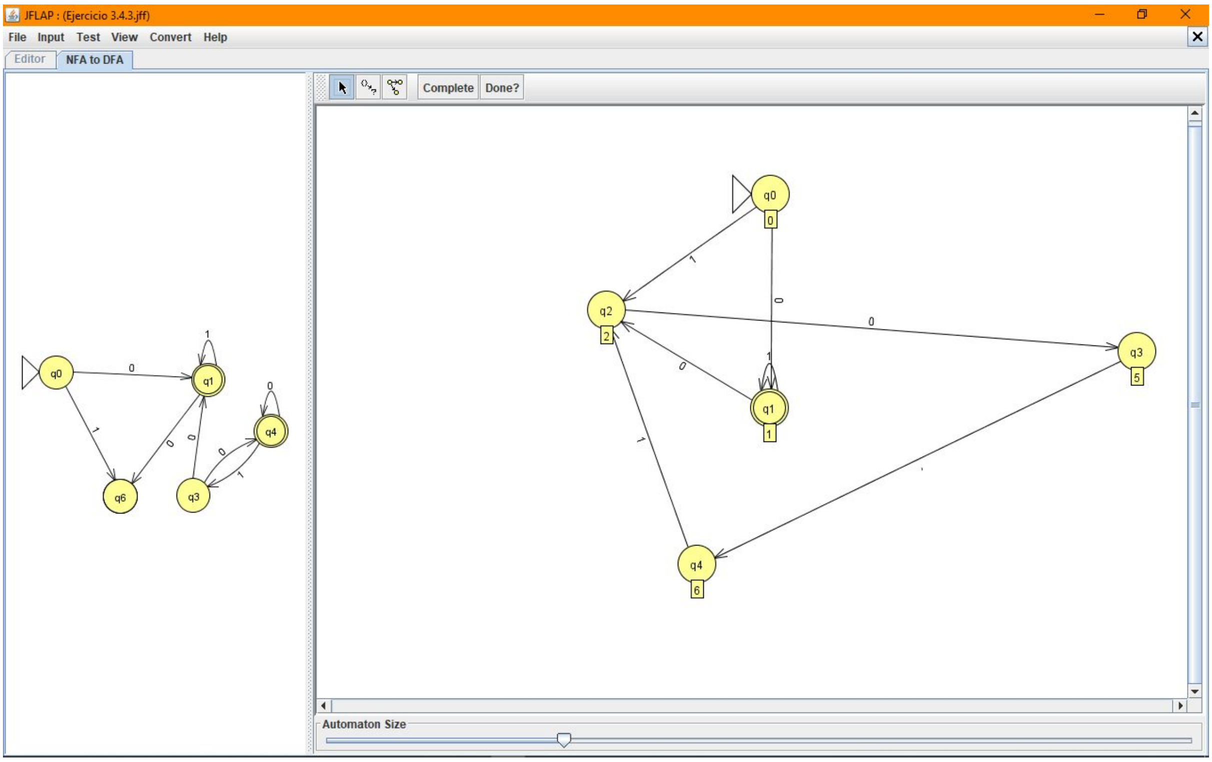


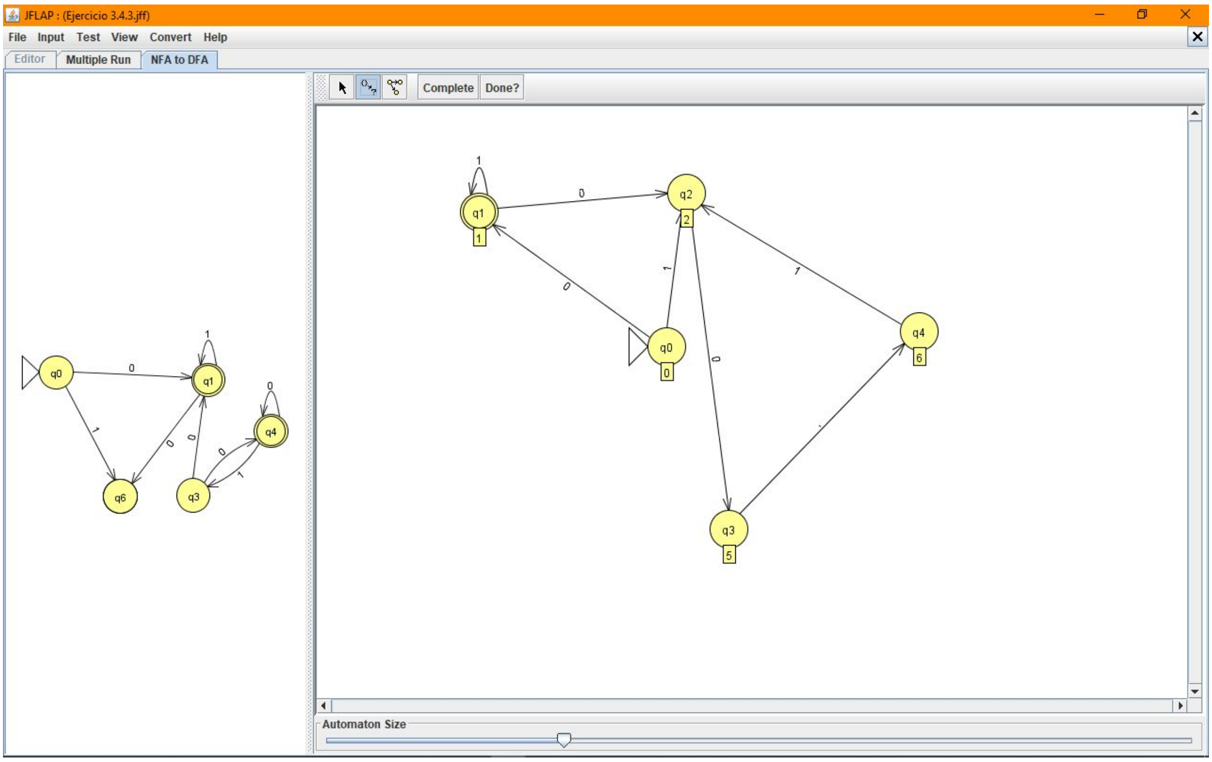


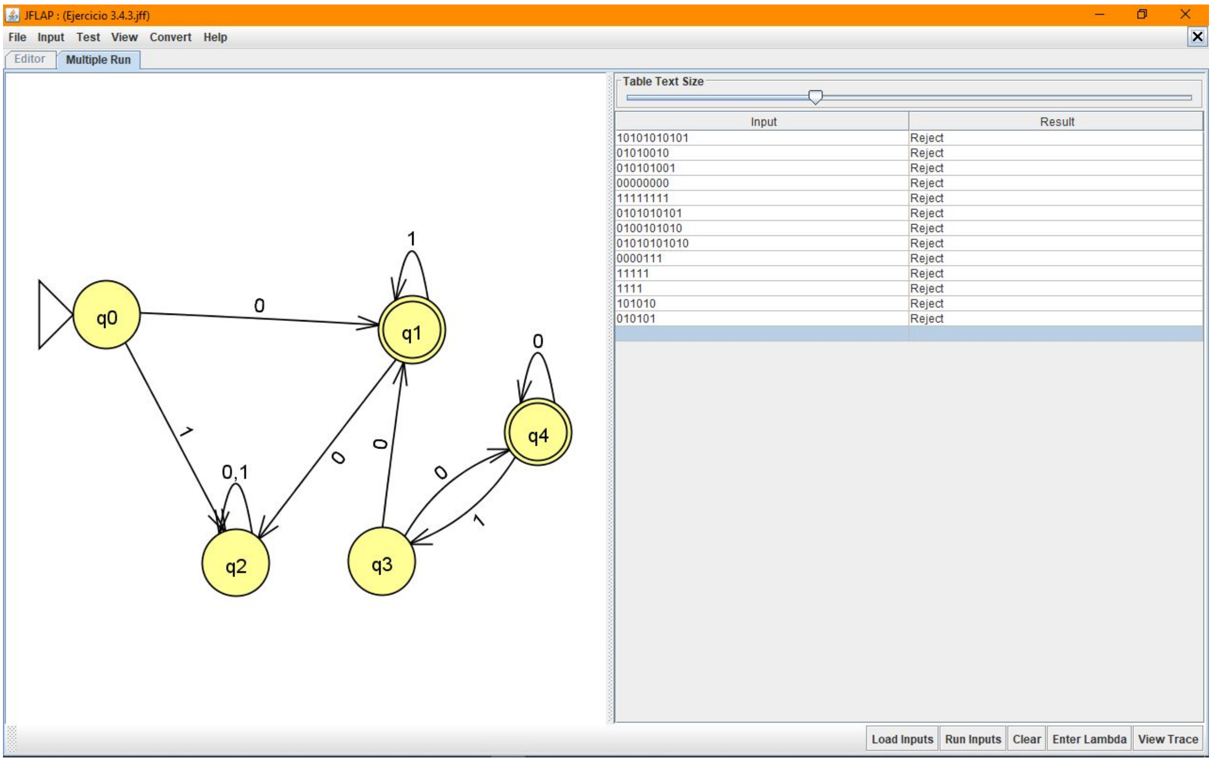
Ejercicio 3.4.3

Ejercicio 3.4.3. En el Ejemplo 3.6 hemos desarrollado la expresión regular
(0+1)∗1(0+1)+(0+1)∗1(0+1)(0+1)
Utilizando las leyes distributivas desarrolle dos expresiones equivalentes diferentes y más simples.




JFLAP : (Ejercicio 3.4.3.jff)
File Input Test View Convert Help
Editor
NFA to DFA
Complete
Done?
Automaton Size


JFLAP : (Ejercicio 3.4.3.jff)
File Input Test View Convert Help
Editor
Multiple Run
NFA to DFA
Complete
Done?
Automaton Size


JFLAP : (Ejercicio 3.4.3.jff)
File Input Test View Convert Help
Editor
Multiple Run
q0
q1
q2
q3
q4
0
1
0,1
Table Text Size
Input
Result
10101010101 Reject
01010010 Reject
010101001 Reject
00000000 Reject
11111111 Reject
0101010101 Reject
0100101010 Reject
01010101010 Reject
0000111 Reject
11111 Reject
1111 Reject
101010 Reject
010101 Reject
Load Inputs
Run Inputs
Clear
Enter Lambda
View Trace