Contenido

- Introducción
- Historia e importancia
- Modelo básico de la MT
- Ejemplo
- Lenguajes tipo 0
- Variaciones de la MT
- Conclusiones

Introducción

Jerarquía de Chomsky

– Gramáticas
– "Poder generativo débil"
– Jerarquía implicativa (`tipo i incluye tipo i+1`)

Introducción

Tipo	Gramática	Lenguaje	Máquina
0	Irrestricta $\lambda_1, \lambda_2 \rightarrow \beta_1, \beta_2$	Recursivamente enumerables	Máquina de Turing
1	Dependiente del Contexto $x\lambda Z \rightarrow x\beta Z$	Dependiente del contexto	Autómatas linealmente acotados
2	Independiente del Contexto $A \rightarrow \beta$	Independiente del Contexto	Autómatas de Pila

Historia

De la matemática a la computación:

- **Siglo XIX, D. Hilbert**

  Validación de fórmulas matemáticas
- **1931, K. Gödel**

  Teorema de la incompletitud
- **1936, A. Turing**

Historia

- Máquinas de Turing → Teoría Matemática de la Computación

- Tesis o hipótesis de Church-Turing

  "Cualquier forma general de computación nos permitirá calcular únicamente las *funciones recursivas parciales*"

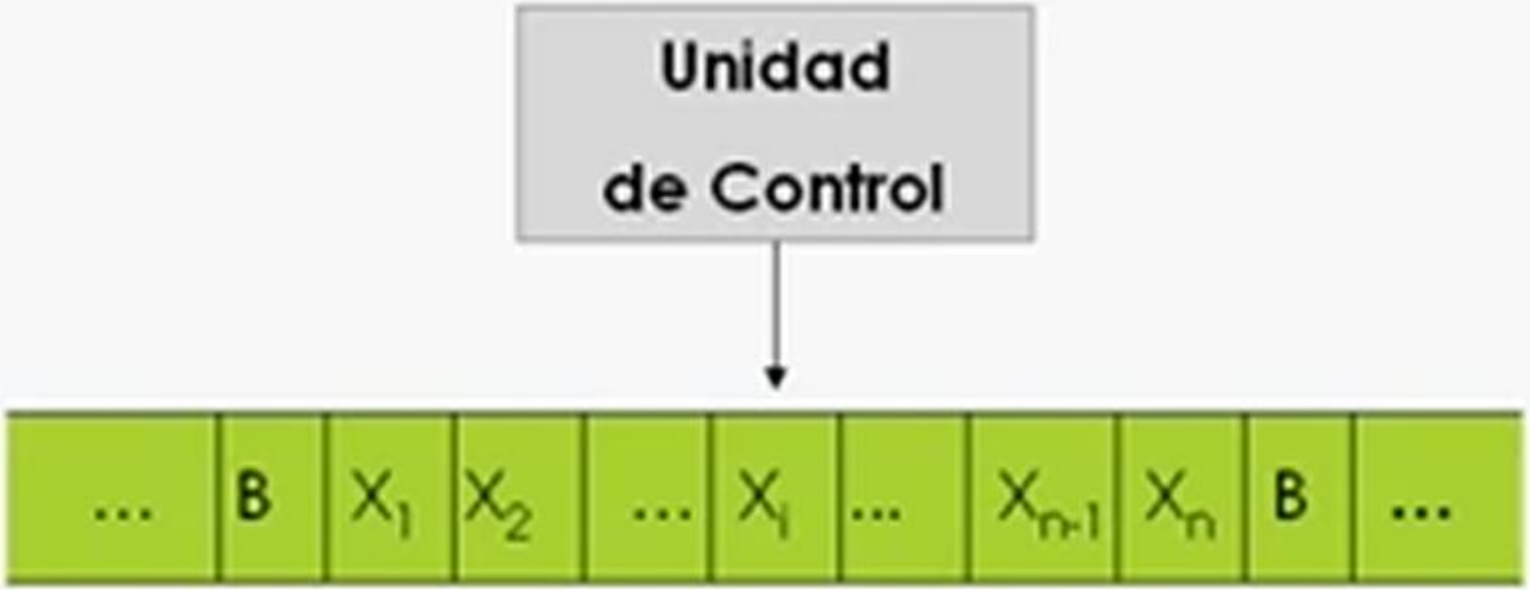
Importancia

- Modelo teórico
- Máquina simple, sencilla y precisa
- Generalidad
- Determinar características de los problemas
  - Decidible (existe un algoritmo)
  - Tratable (existe un algoritmo rápido)

Modelo básico

Componentes:

- Unidad de Control (autómata finito)
- Cinta infinita



Máquinas de Turing
Modelo básico

- Cinta infinita
  - Infinita a ambos extremos
  - Dividida en casillas
  - Un símbolo por casilla, inicializada en B
  - Entrada w ubicada sobre la cinta infinita
  - Cabezal que indica posición actual. Inicia en el primer símbolo de la entrada.
  - Cabezal con movimiento secuencial infinidad de veces de izquierda a derecha

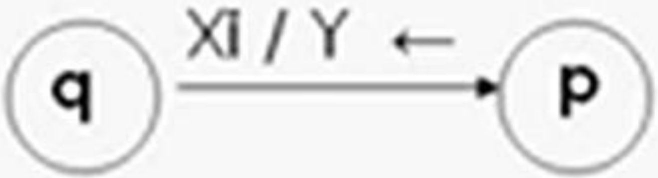
Modelo básico

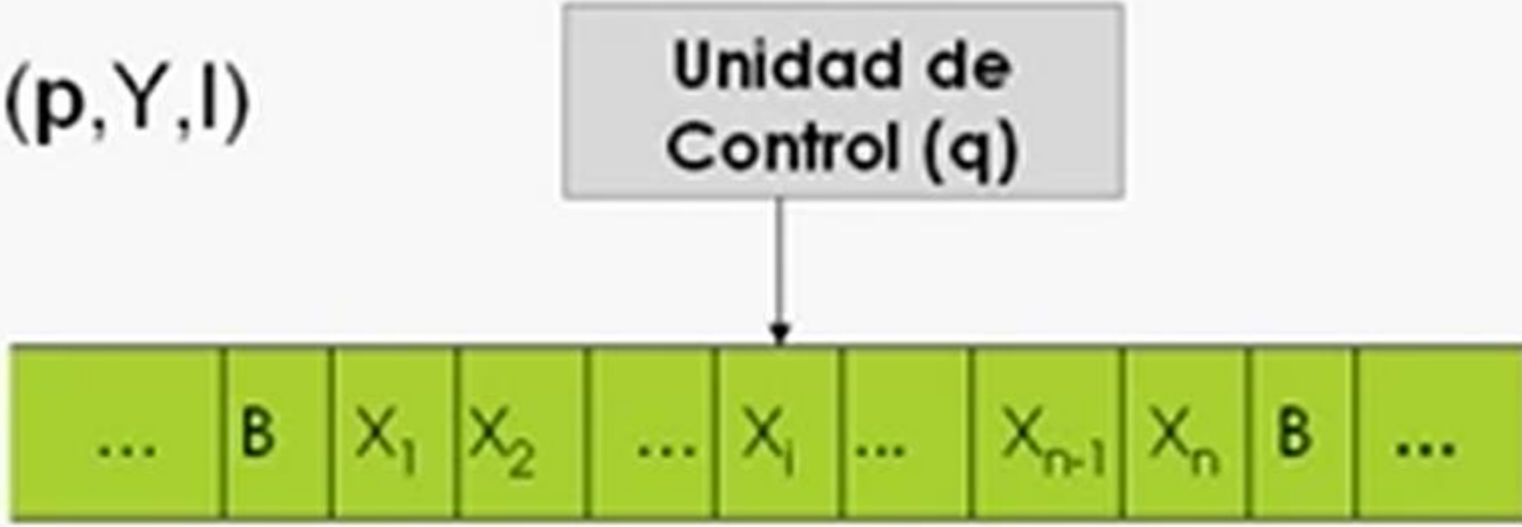
$$\mathbf{M} = (\mathbf{Q}, \Sigma, \Gamma, \delta, \mathbf{q_0}, \mathbf{B}, \mathbf{F})$$

- **Q**: conjunto de estados
- Σ: símbolos de entrada
- Γ: símbolos de la cinta. Γ contiene a Σ U B
- δ: $Q \times \Gamma \rightarrow Q \times \Gamma \times \{D, I\}$
- $\mathbf{q_0}$: estado inicial
- **B**: símbolo espacio en blanco

Modelo básico

- **Transición** $\delta(\mathbf{q},X_i) = (\mathbf{p},Y,I)$





- **Entrada** (Σ^*)

  $X_1\ X_2\ \ldots\ X_{i-1}\ X_i\ X_{i+1}\ \ \ldots\ \ X_n$

  Descripción instantánea: $X_1\ X_2\ \ldots\ X_{i-1}\ \mathbf{q}\ X_i\ X_{i+1}\ \ \ldots\ \ X_n$

- **Movimiento** $\vdash$

  $X_1\ \ldots\ X_{i-1}\ \mathbf{q}\ X_i\ X_{i+1}\ \ \ldots\ \ X_n \vdash\ X_1\ \ldots\ X_{i-2}\ \mathbf{p}\ X_{i-1}\ Y\ X_{i+1}\ \ldots\ X_n$

Ejemplo

- Diseñar una máquina de Turing que acepte el lenguaje:

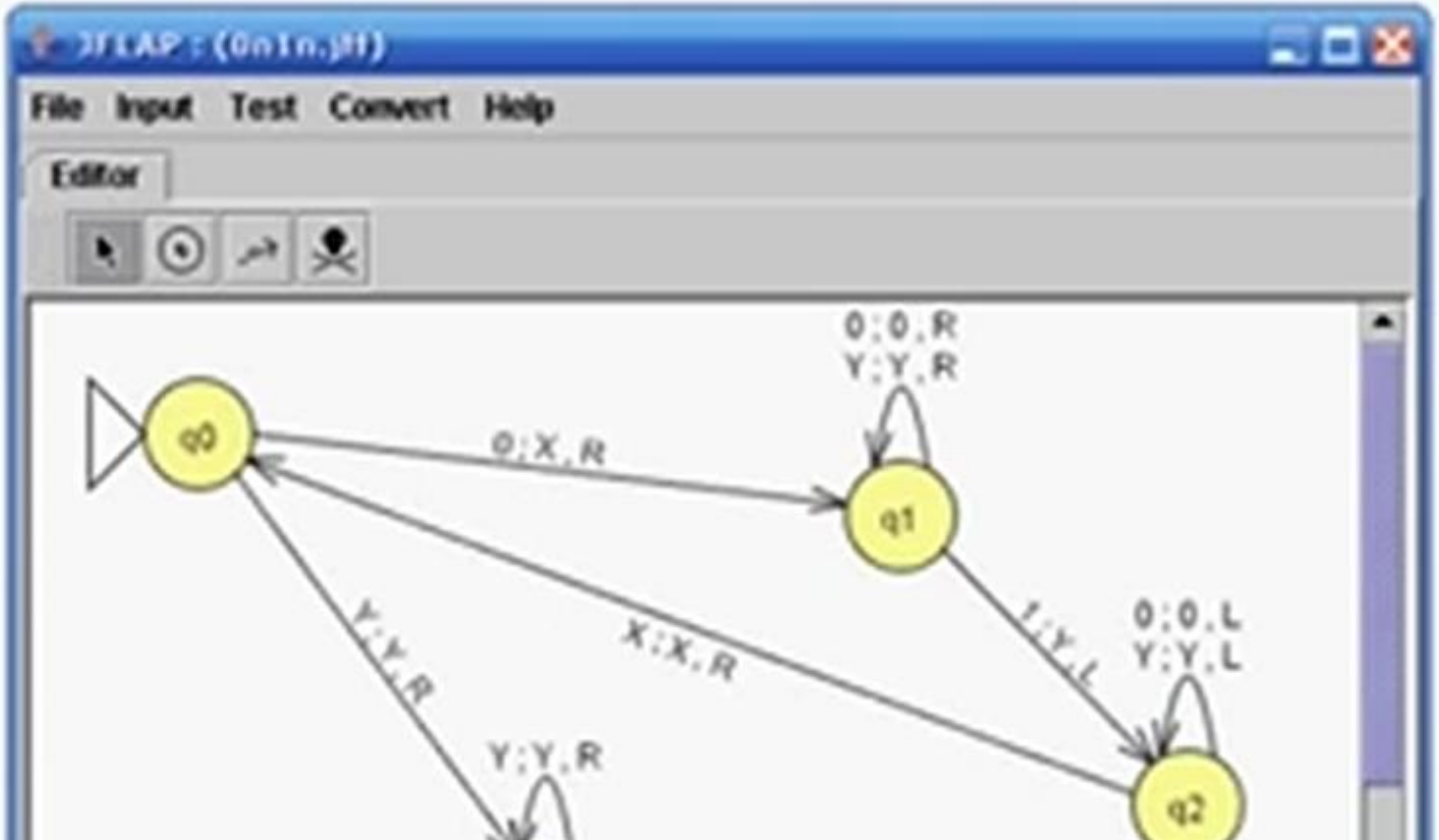
  $L = \{ 0^n 1^n \mid n \geq 1\}$

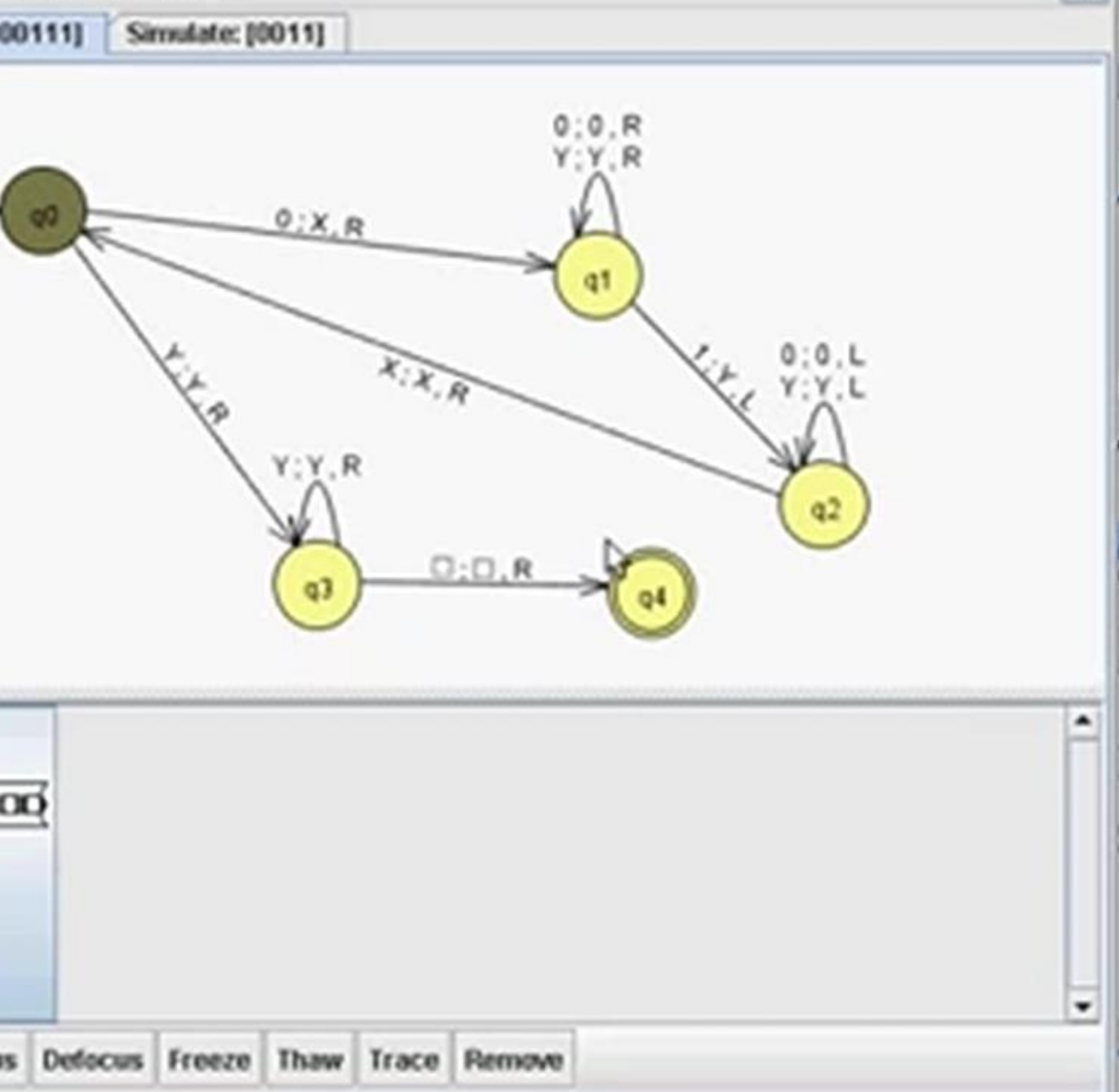
Idea:

- Cambiar **0** por **X** y **1** por **Y** en una pasada
- Repetir hasta el final de la cadena de entrada
- Si hay **$X^n Y^n$** la máquina acepta
- $\Sigma = \{0,1\}$

Ejemplo

- La máquina de Turing que acepta el lenguaje: **$\{ 0^n1^n \mid n \geq 1\}$** (archivo 0n1n.jff)



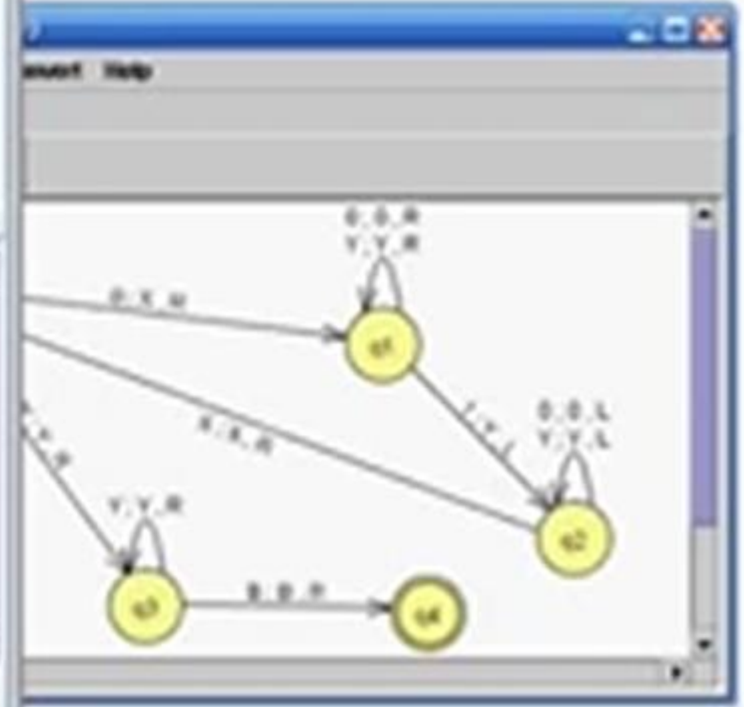


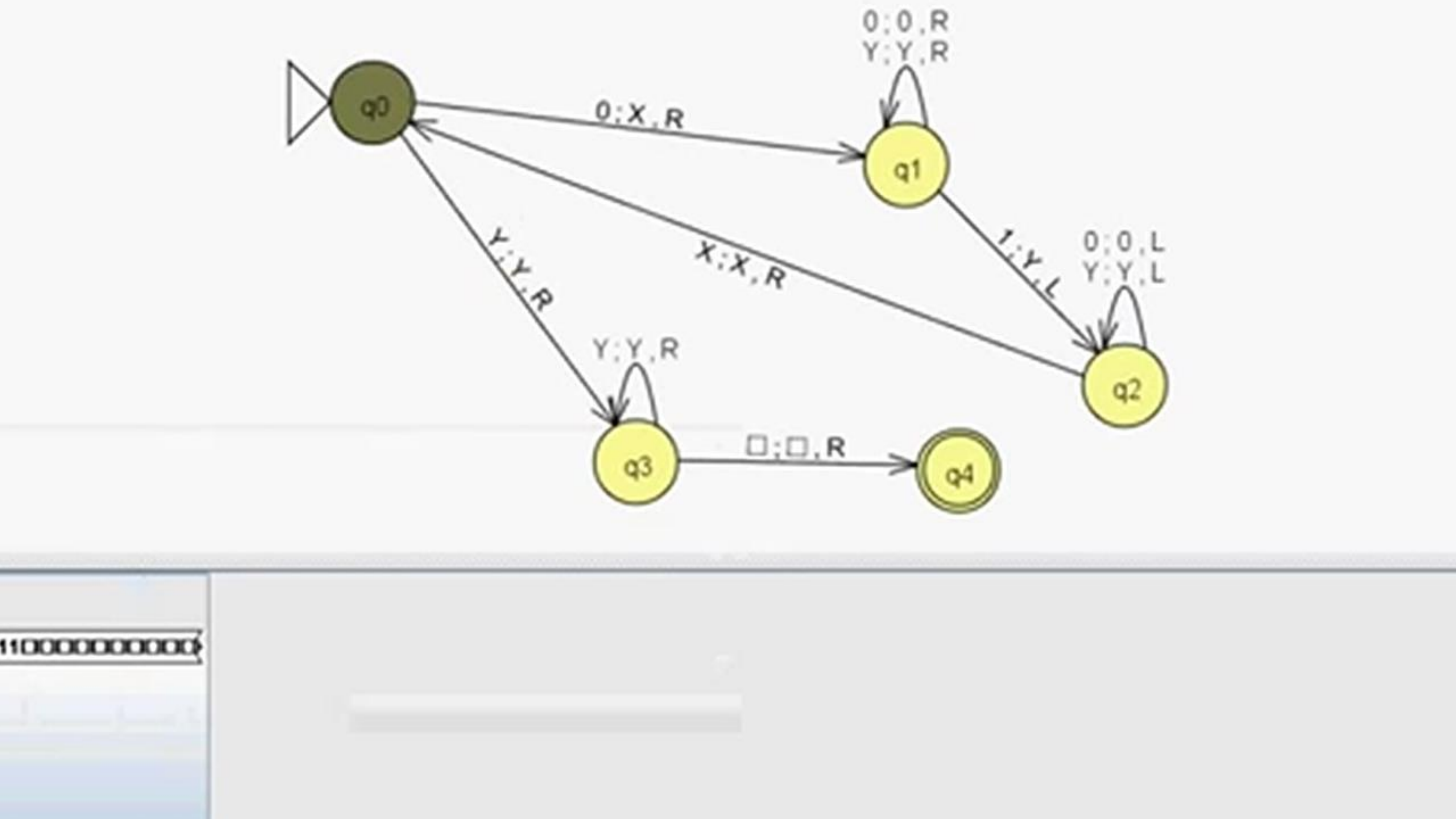
uinas de Turing

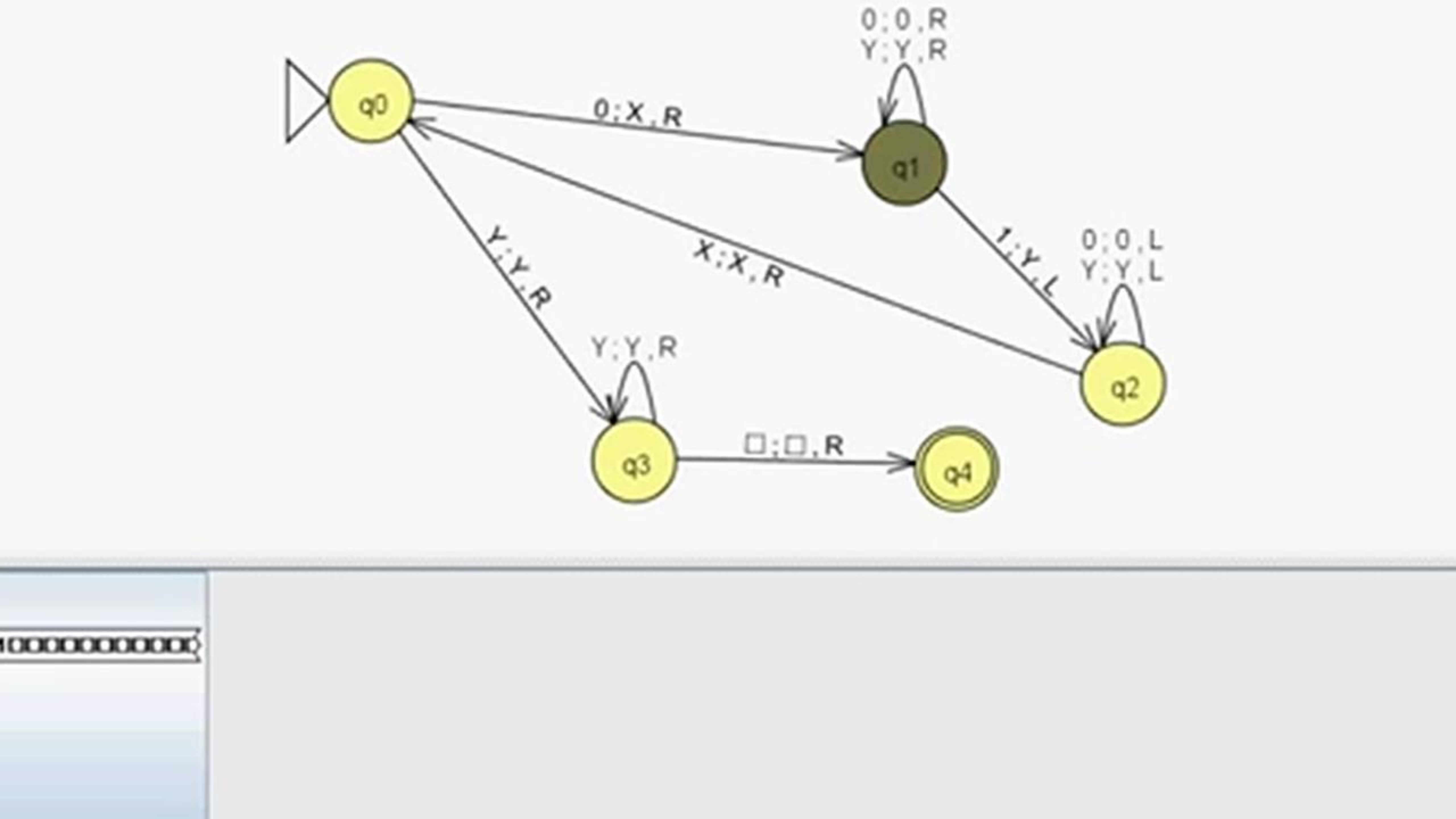
Ejemplo

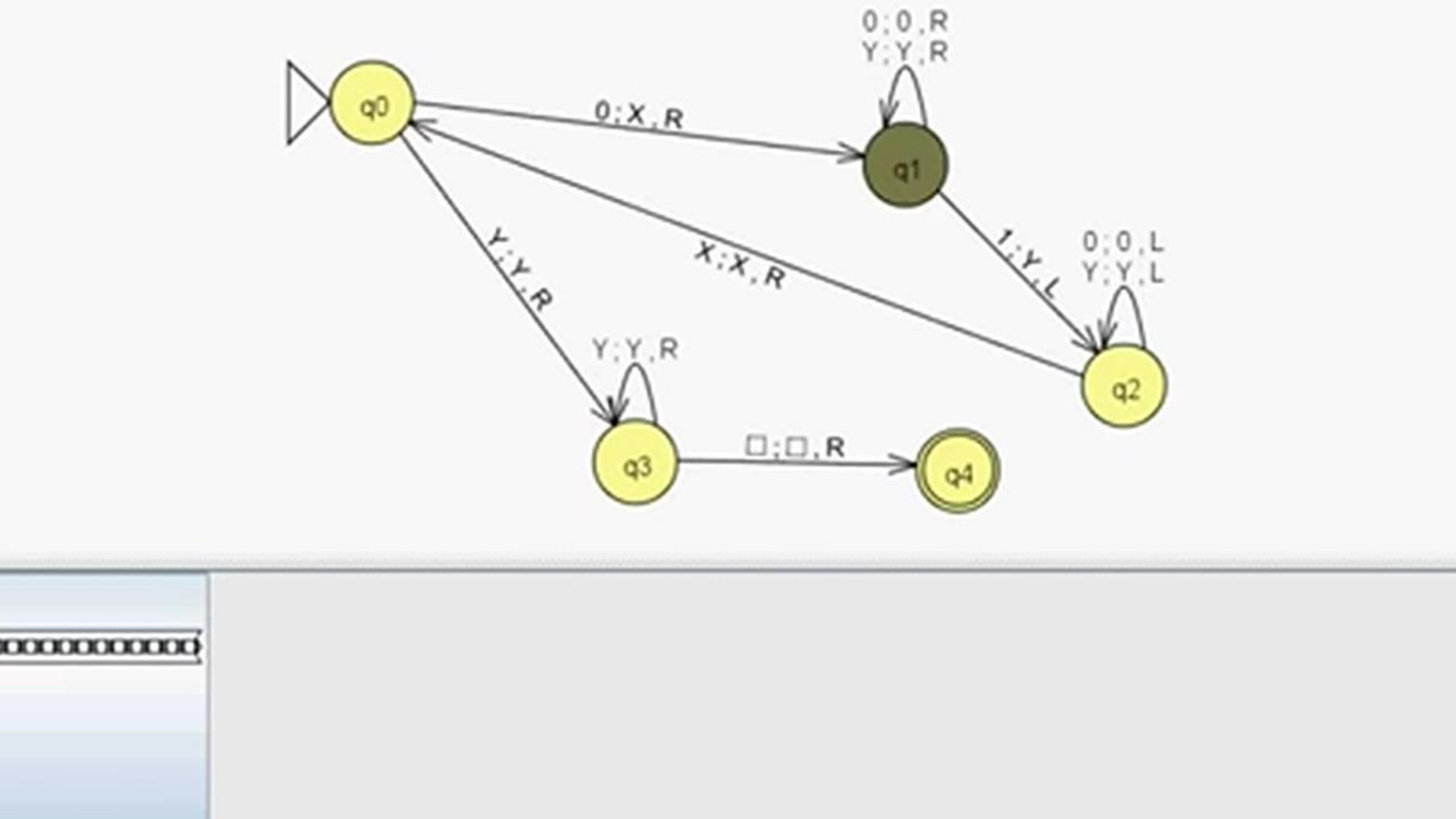
e Turing que acepta el

1^n | n ≥ 1} (archivo 0n1n.jff)

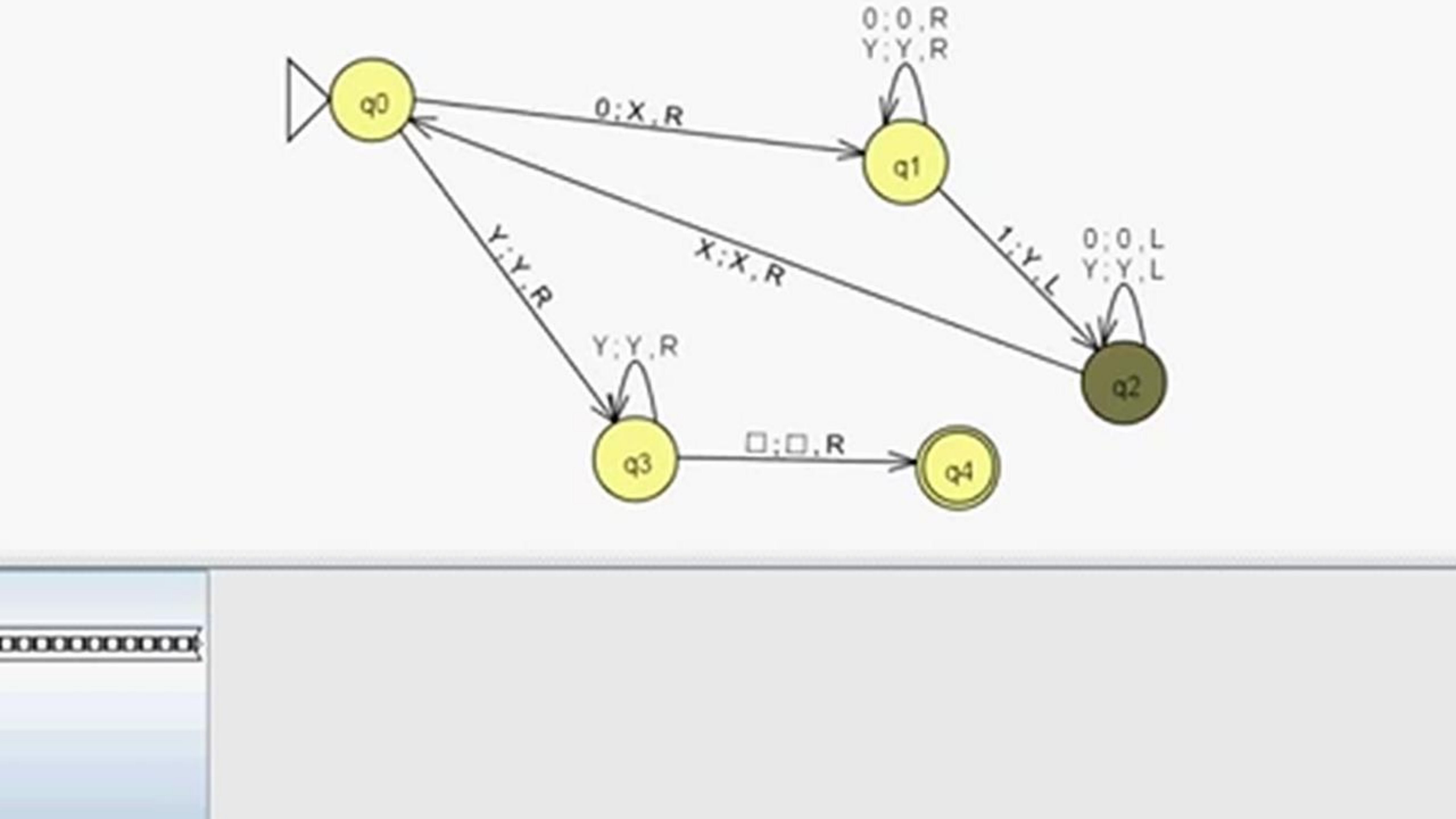



q0
q1
q2
q3
q4
0 ; X , R
0 ; 0 , R
Y ; Y , R
1 ; Y , L
0 ; 0 , L
Y ; Y , L
X ; X , R
Y ; Y , R
Y ; Y , R
□ ; □ , R

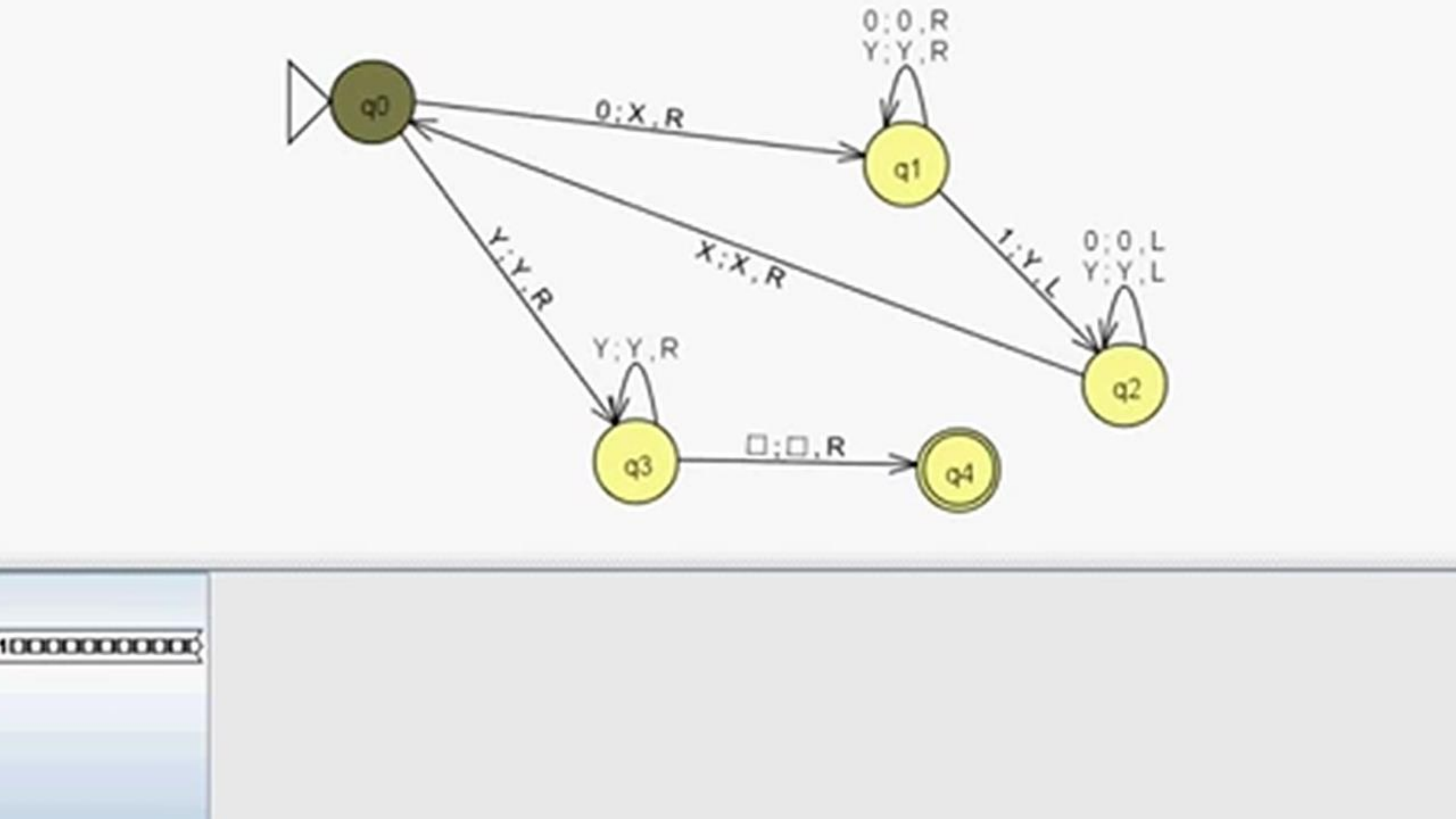

0;0,R
Y;Y,R
q0
0;X,R
q1
Y;Y,R
X;X,R
1;Y,L
0;0,L
Y;Y,L
Y;Y,R
q2
q3
□;□,R
q4

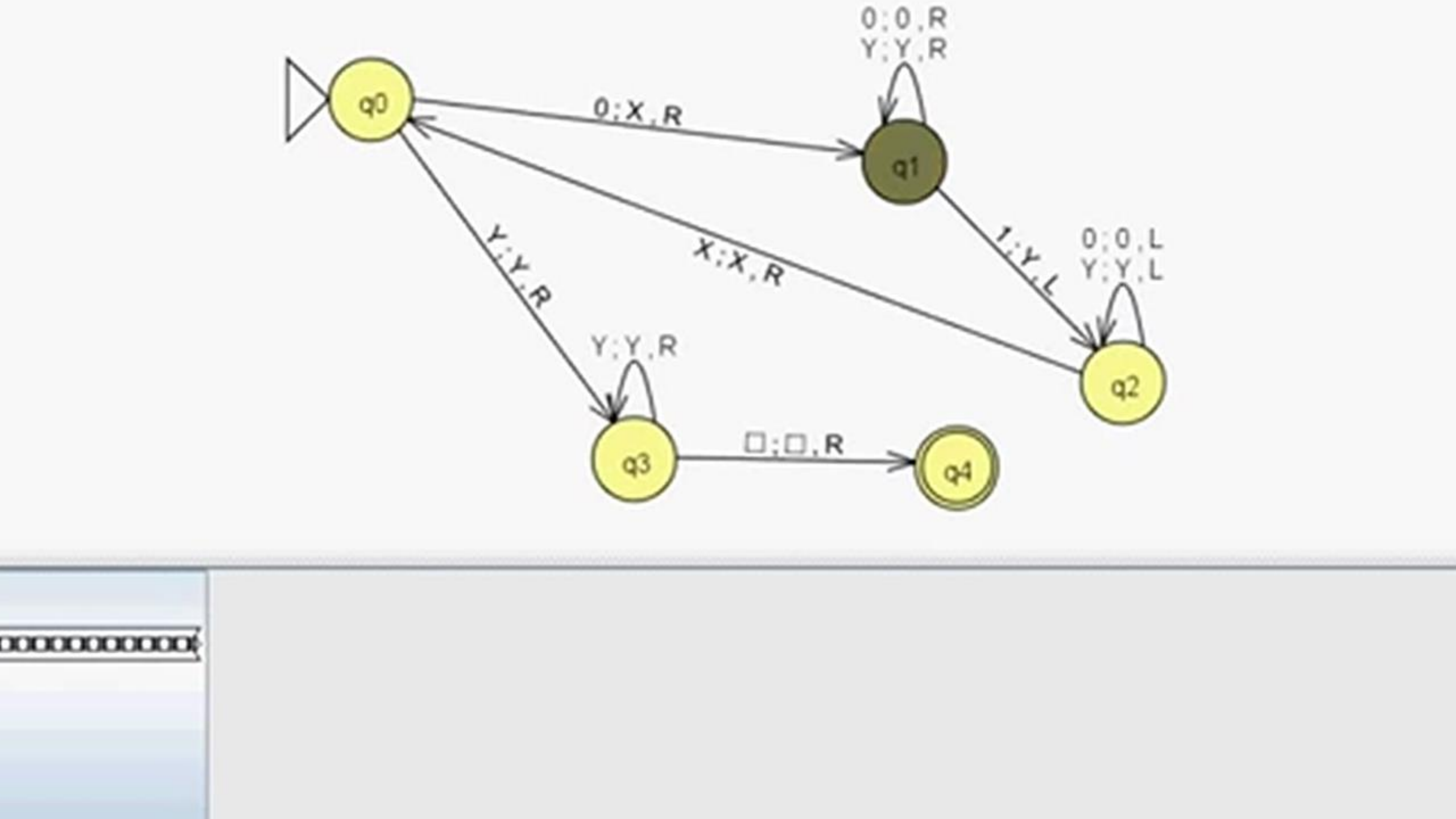


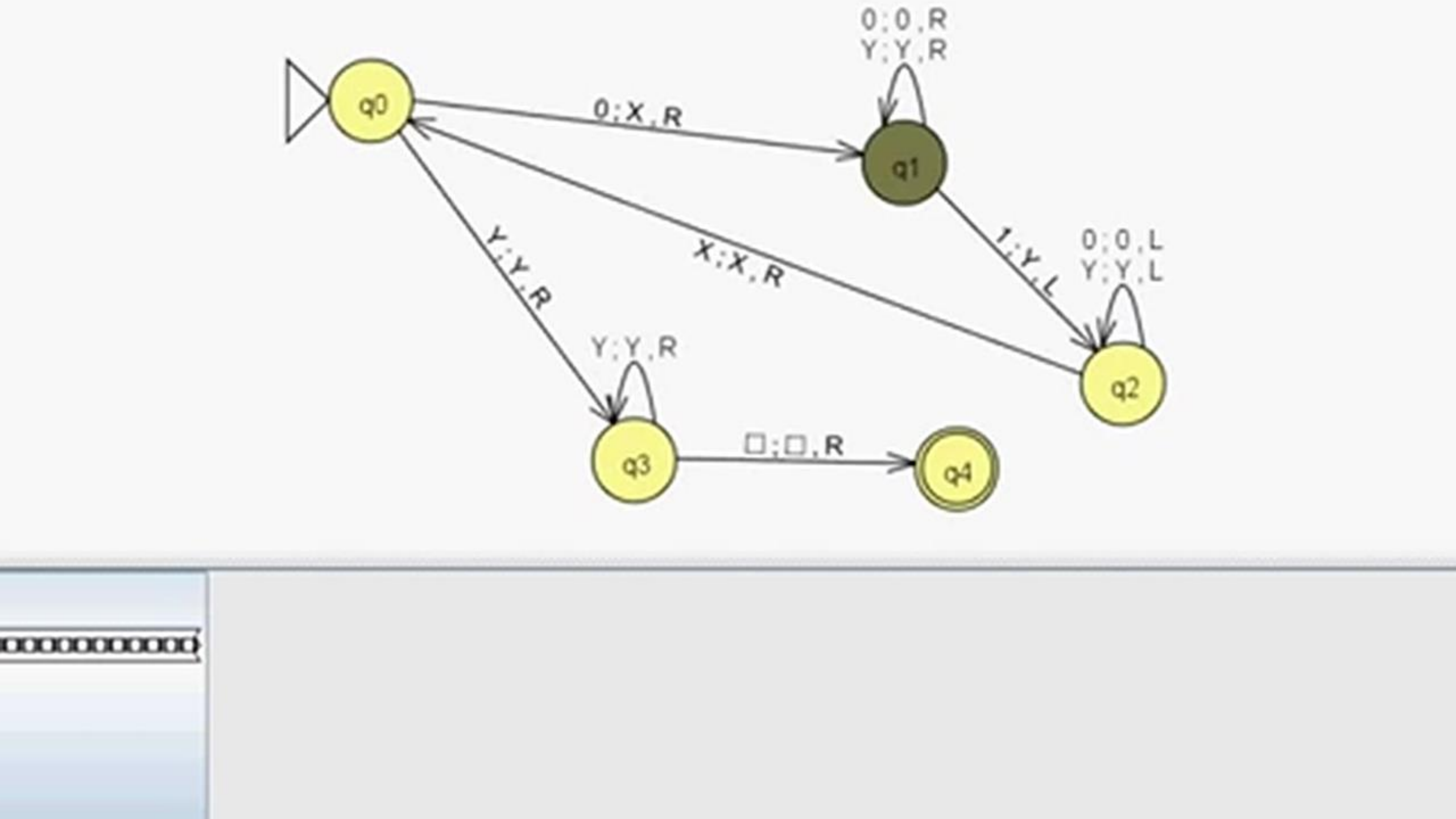
0;0,R
Y;Y,R
q0
0;X,R
q1
1;Y,L
0;0,L
Y;Y,L
X;X,R
Y;Y,R
q2
Y;Y,R
q3
□;□,R
q4

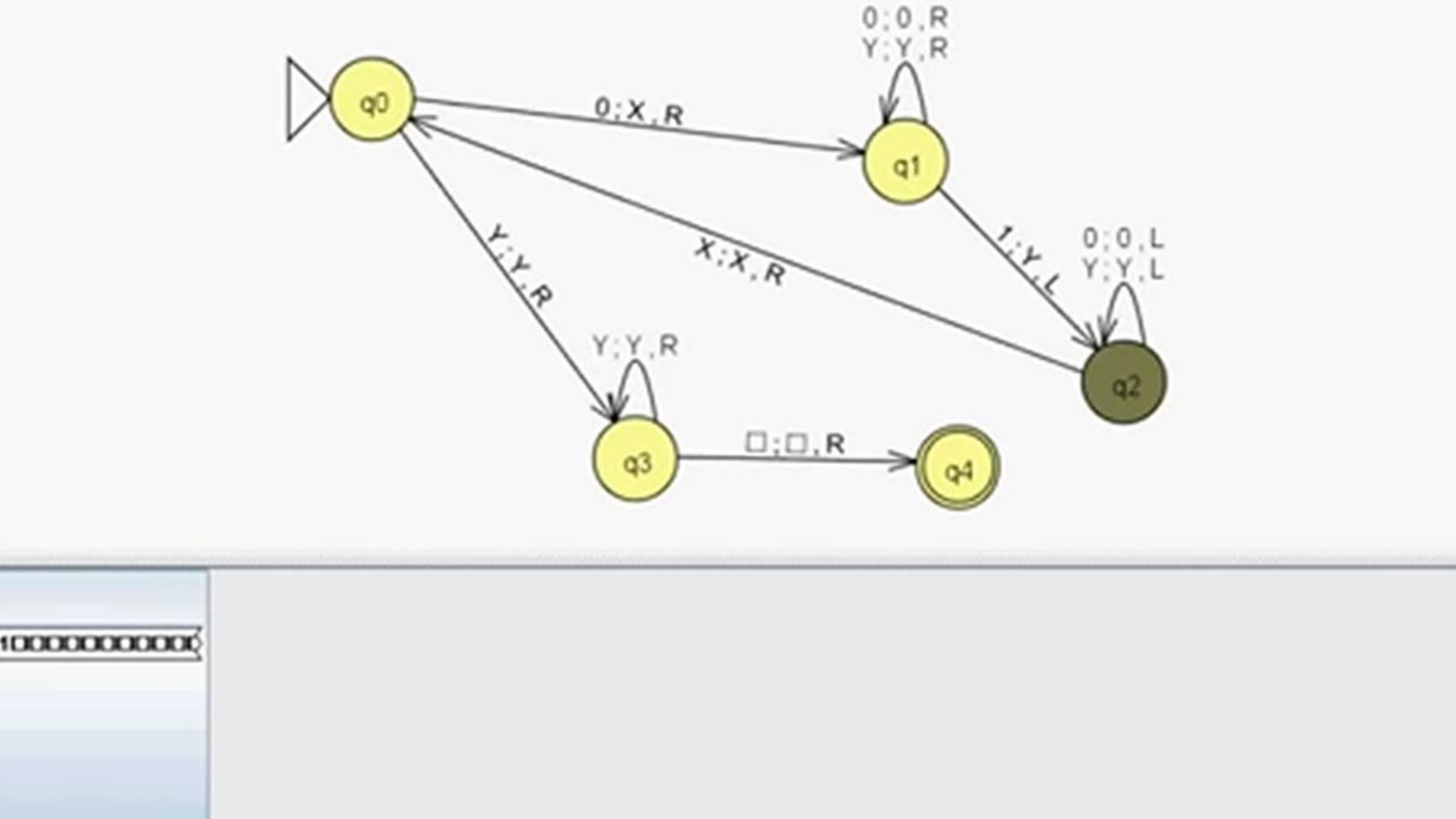


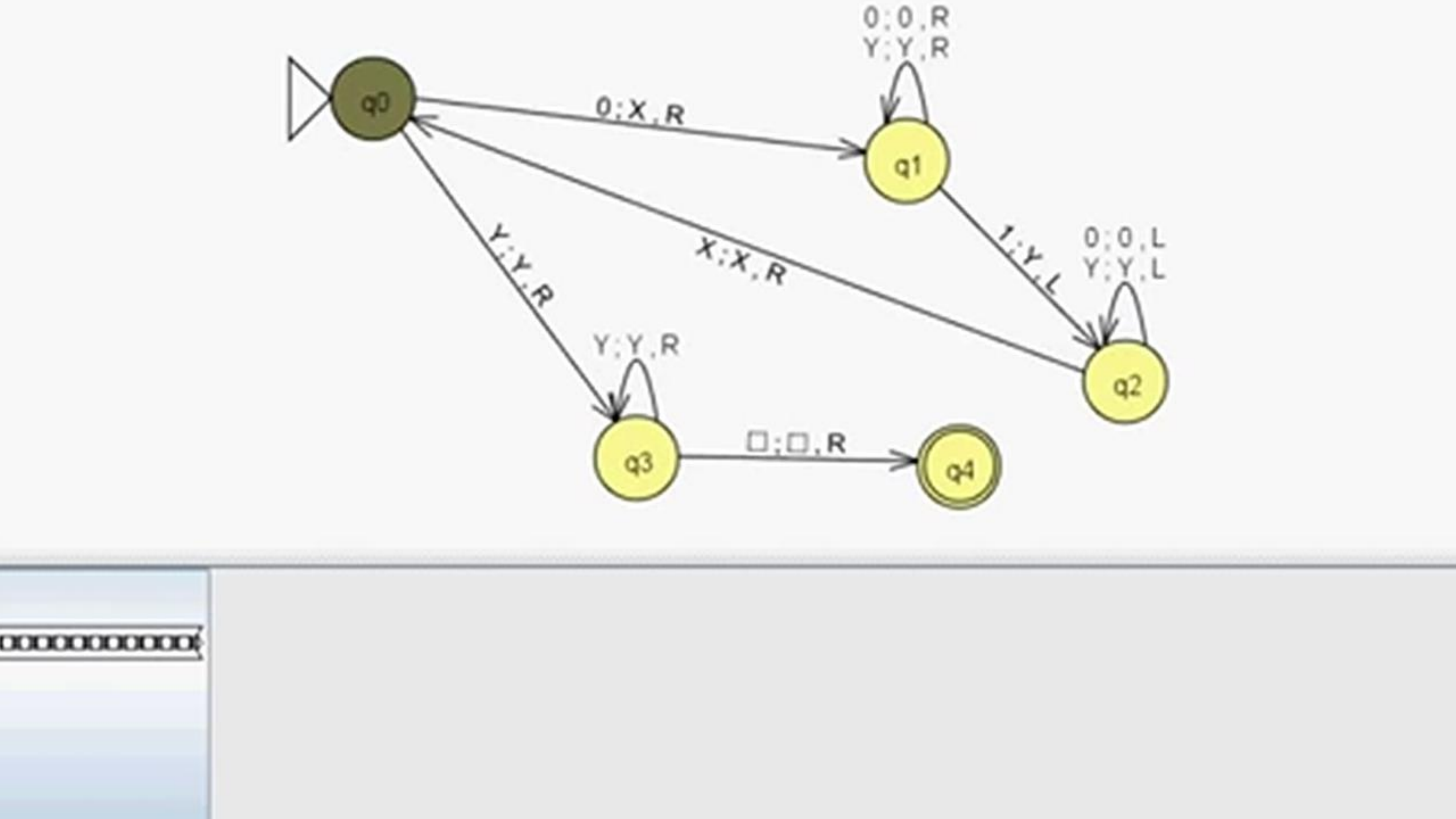
0;0,R
Y;Y,R
q0
0;X,R
q1
1;Y,L
X;X,R
Y;Y,R
0;0,L
Y;Y,L
q2
Y;Y,R
q3
□;□,R
q4

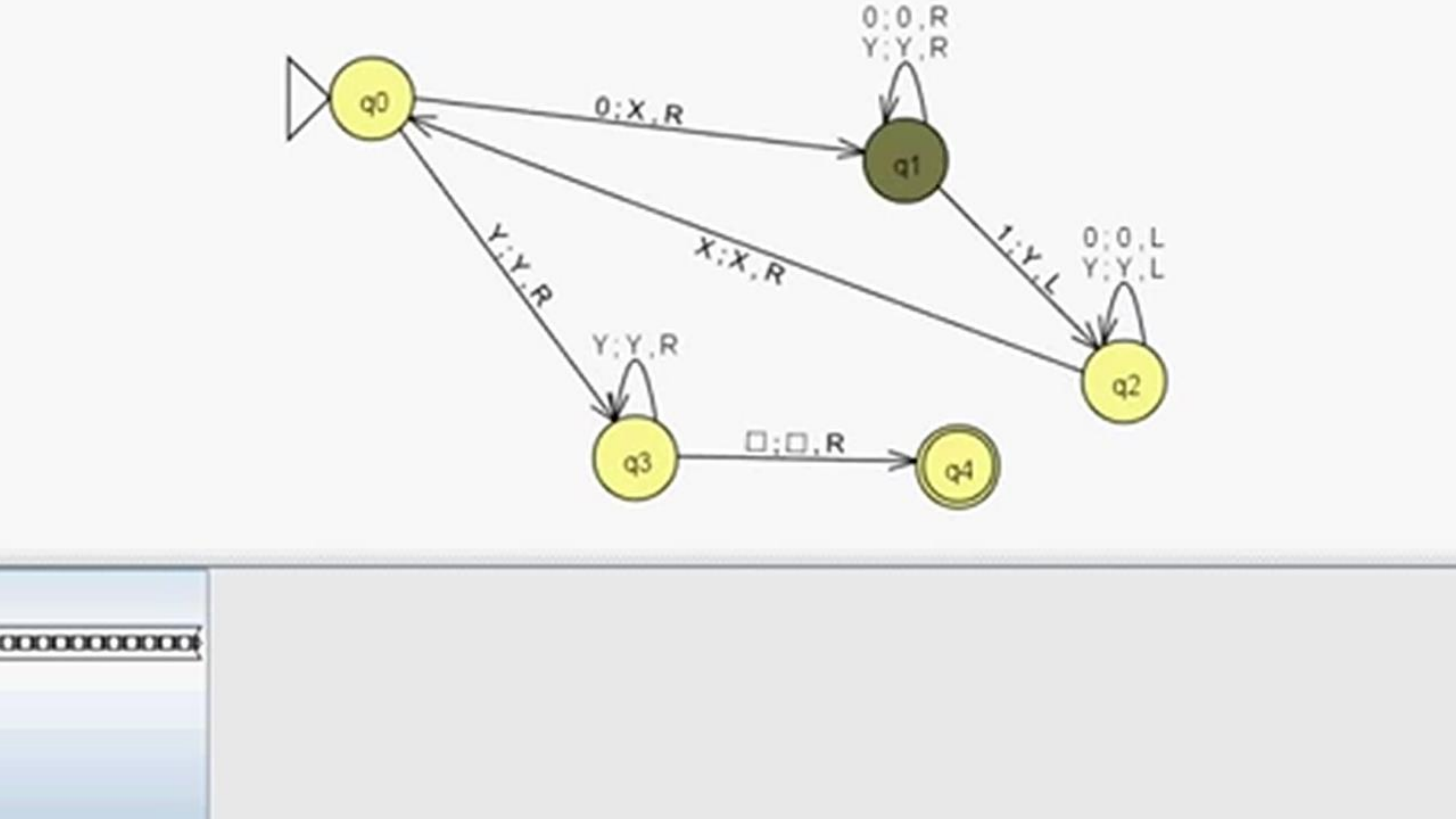

0;0,R
Y;Y,R
q0
0;X,R
q1
1;Y,L
0;0,L
Y;Y,L
q2
X;X,R
Y;Y,R
q3
Y;Y,R
□;□,R
q4

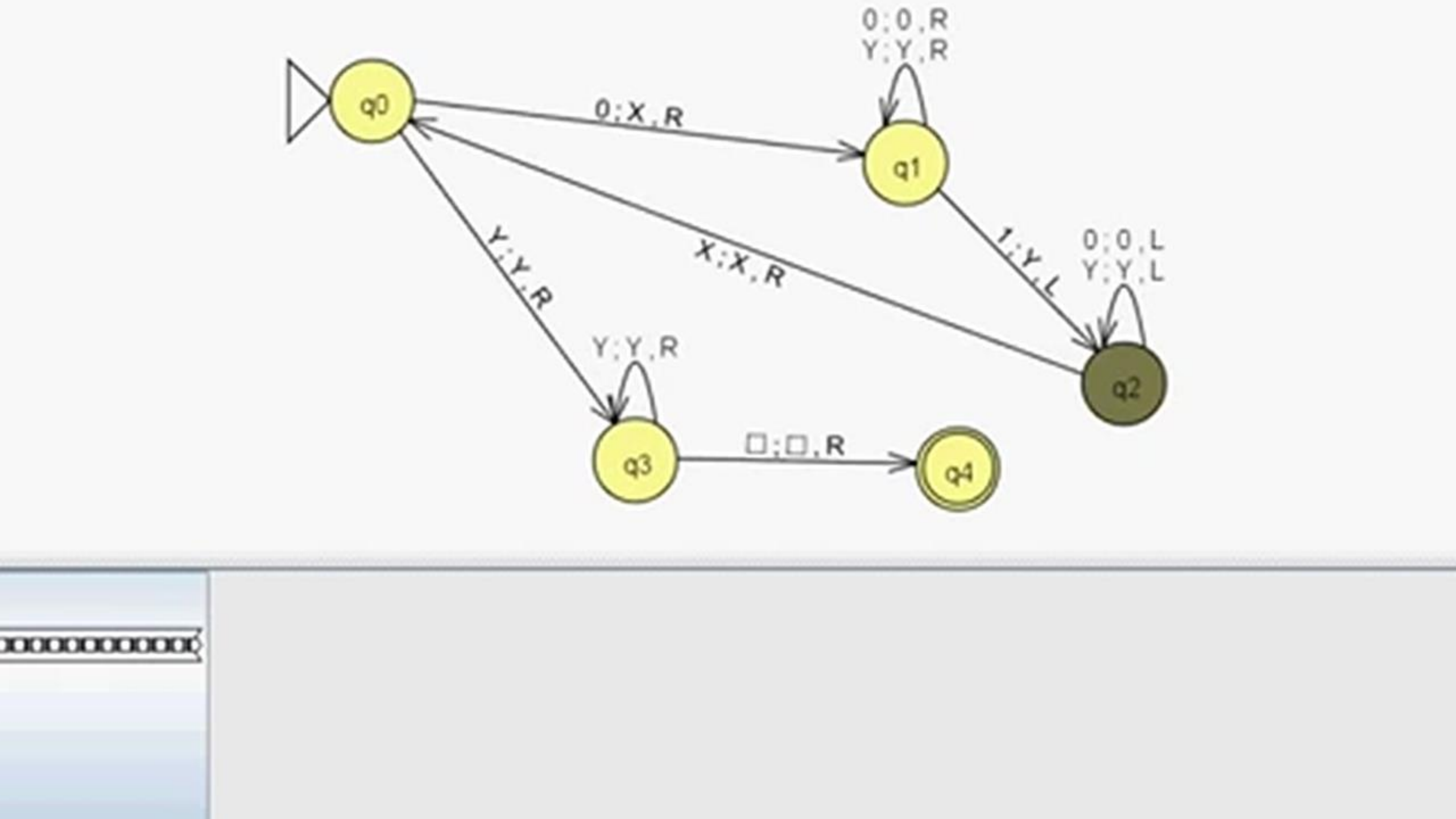

0;0,R
Y;Y,R
q0
0;X,R
q1
Y;Y,R
X;X,R
1;Y,L
0;0,L
Y;Y,L
Y;Y,R
q2
q3
□;□,R
q4


0;0,R
Y;Y,R
q0
0;X,R
q1
Y;Y,R
X;X,R
1;Y,L
0;0,L
Y;Y,L
Y;Y,R
q2
q3
□;□,R
q4

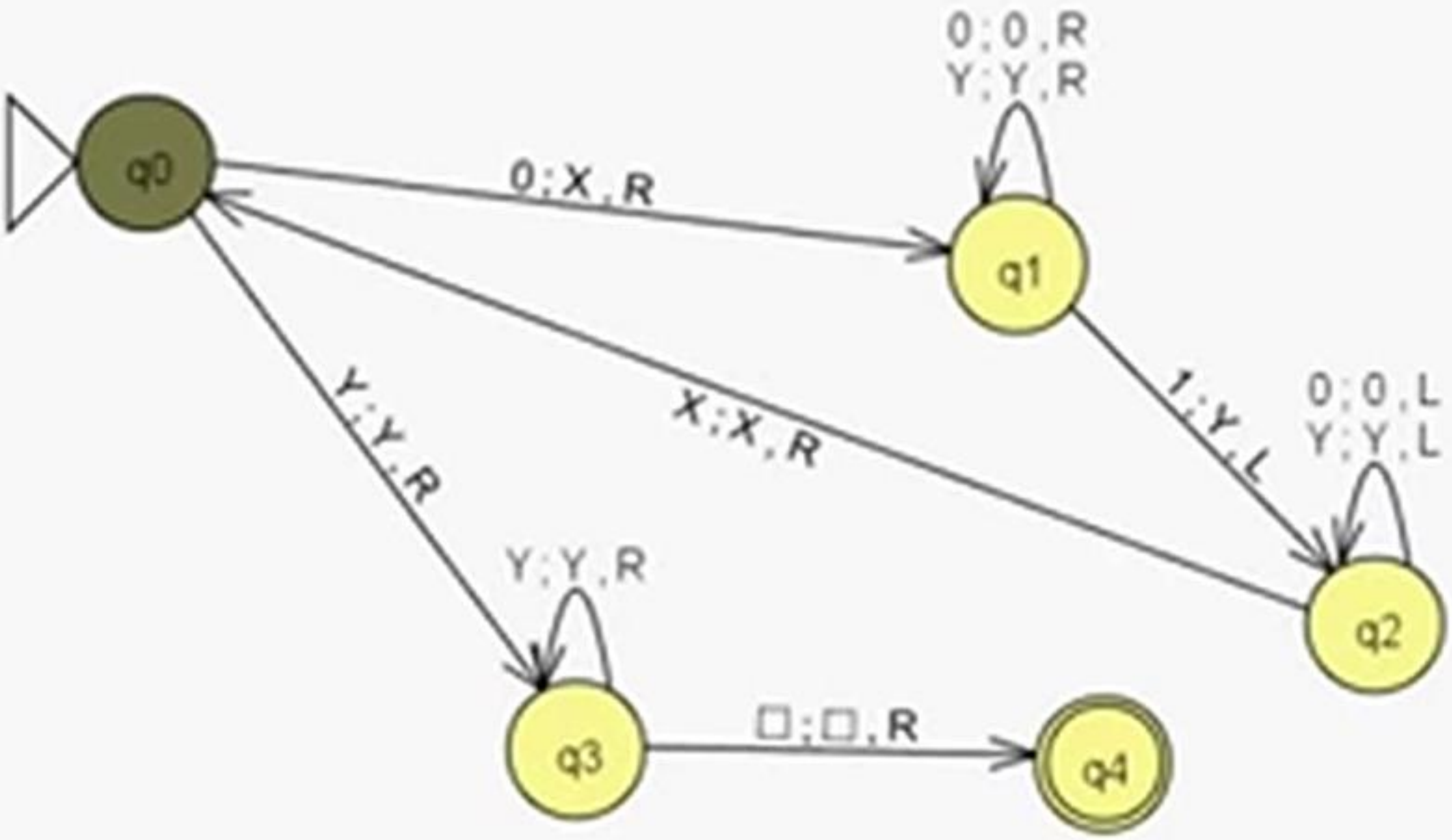

q0
q1
q2
q3
q4
0;X,R
0;0,R
Y;Y,R
1;Y,L
0;0,L
Y;Y,L
X;X,R
Y;Y,R
Y;Y,R
□;□,R

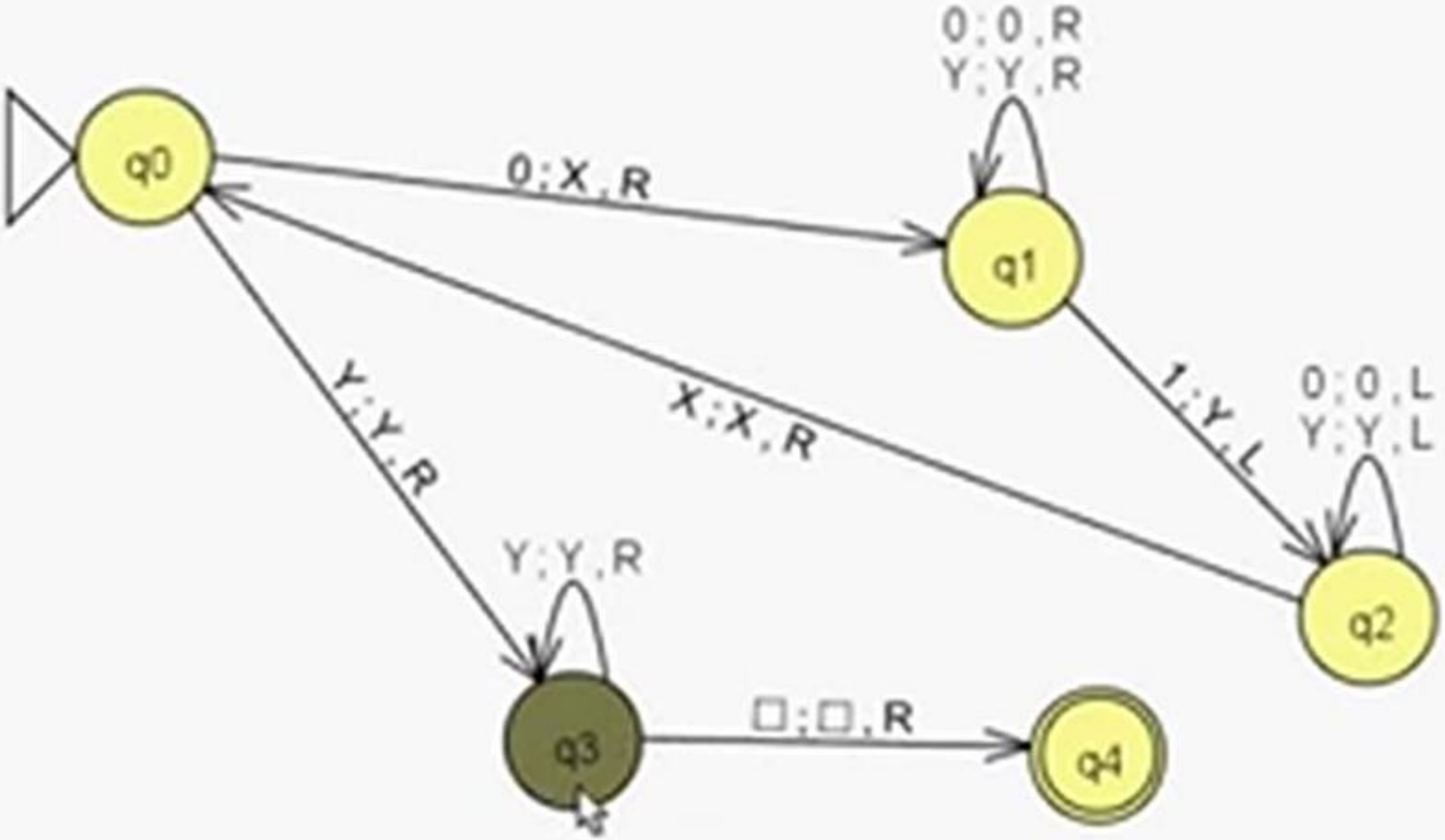

q0
q1
q2
q3
q4
0;X,R
0;0,R
Y;Y,R
1;Y,L
0;0,L
Y;Y,L
X;X,R
Y;Y,R
Y;Y,R
□;□,R


q0
q1
q2
q3
q4
0;0,R
Y;Y,R
0;X,R
1;Y,L
0;0,L
Y;Y,L
X;X,R
Y;Y,R
Y;Y,R
□;□,R

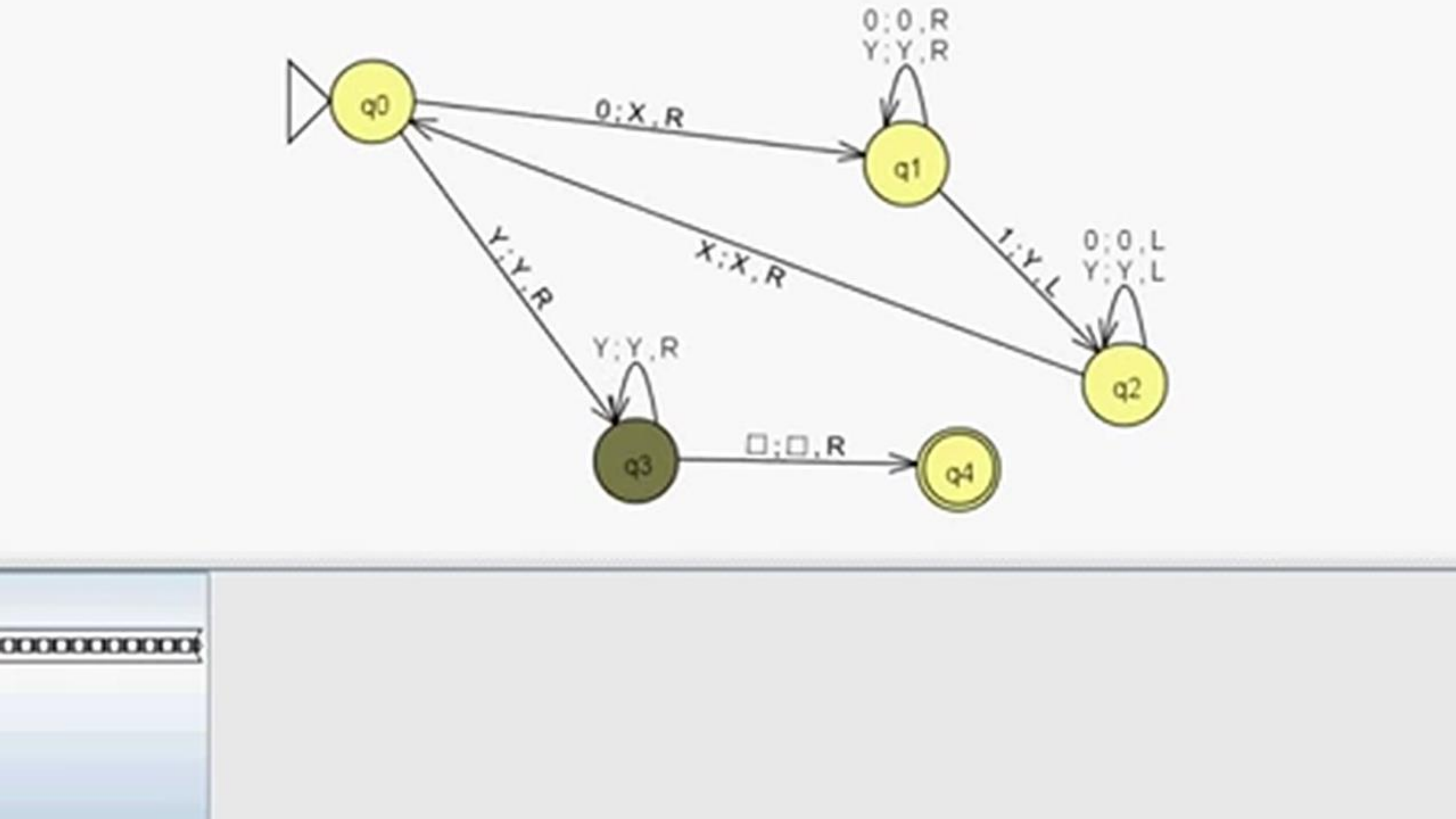


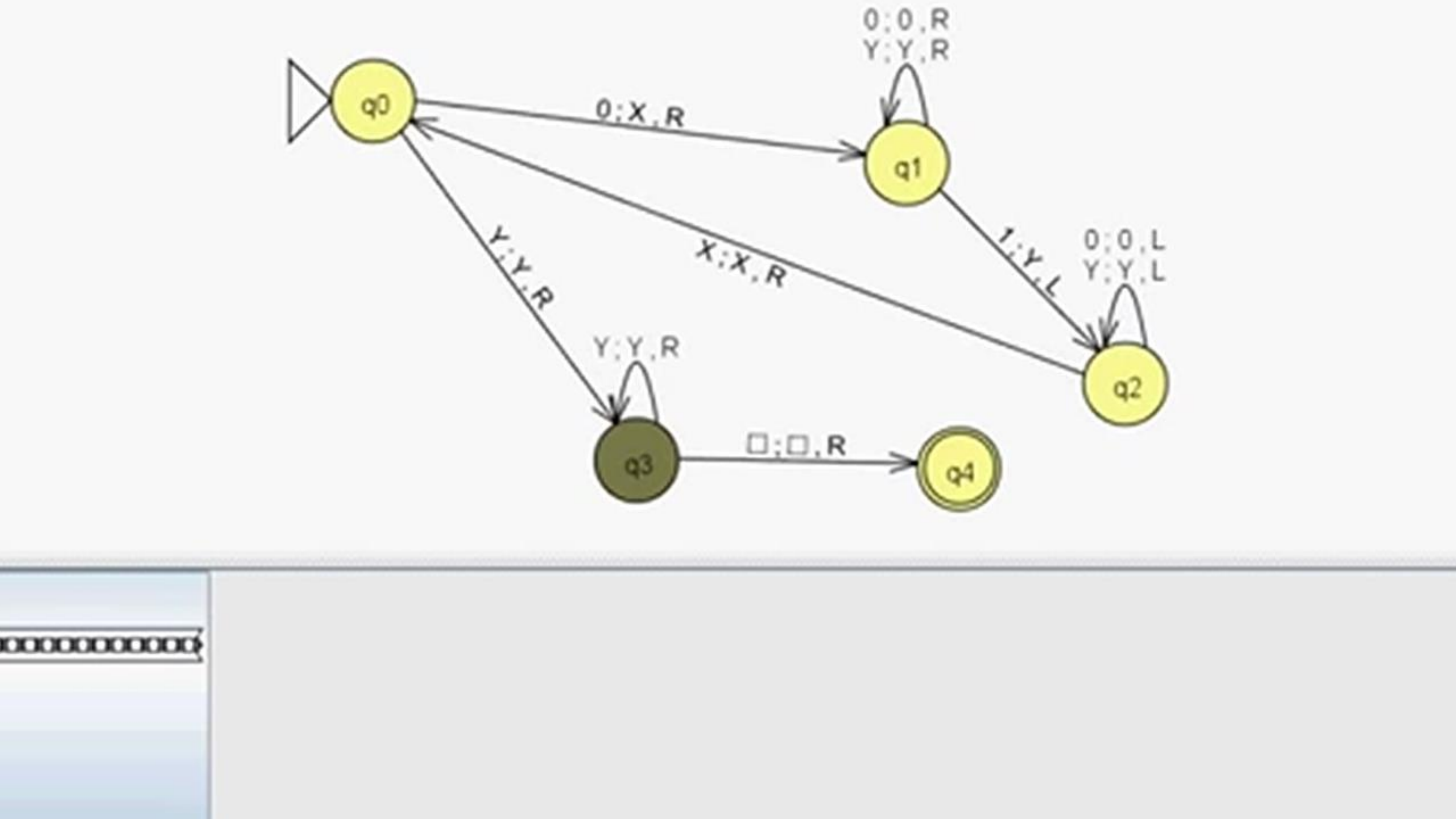
0 ; 0 , R
Y ; Y , R
q0
0 ; X , R
q1
1 ; Y , L
0 ; 0 , L
Y ; Y , L
X ; X , R
Y ; Y , R
q2
Y ; Y , R
q3
□ ; □ , R
q4


q0
q1
q2
q3
q4
0;X,R
0;0,R
Y;Y,R
1;Y,L
0;0,L
Y;Y,L
X;X,R
Y;Y,R
Y;Y,R
□;□,R

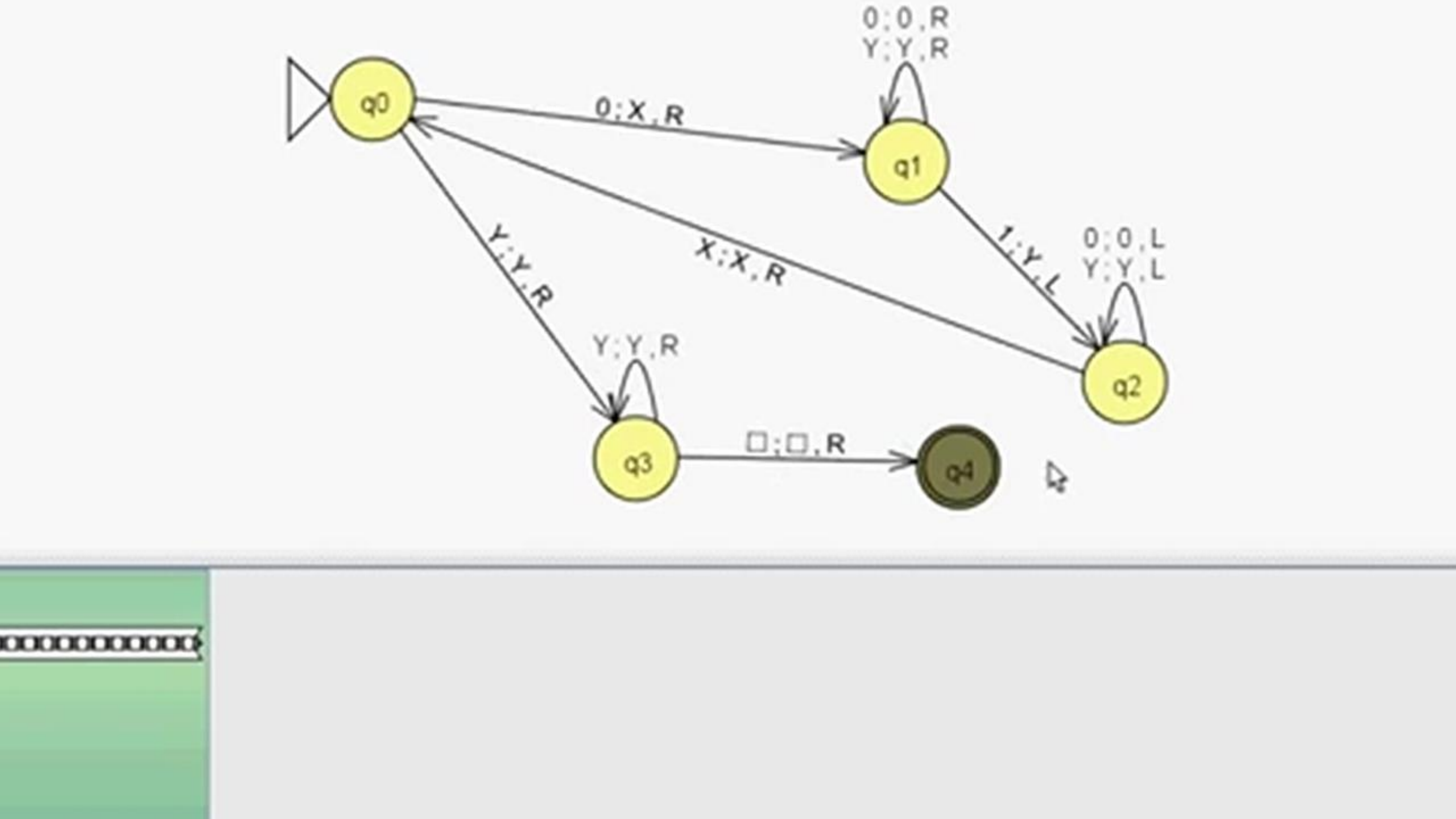


0;0,R
Y;Y,R
q0
0;X,R
q1
Y;Y,R
X;X,R
1;Y,L
0;0,L
Y;Y,L
Y;Y,R
q2
q3
□;□,R
q4


0;0,R
Y;Y,R
q0
0;X,R
q1
1;Y,L
0;0,L
Y;Y,L
q2
X;X,R
Y;Y,R
Y;Y,R
q3
□;□,R
q4



0 ; 0 , R
Y ; Y , R
q0
0 ; X , R
q1
1 ; Y , L
X ; X , R
Y ; Y , R
0 ; 0 , L
Y ; Y , L
q2
Y ; Y , R
q3
□ ; □ , R
q4


0 ; 0 , R
Y ; Y , R
q0
0 ; X , R
q1
Y ; Y , R
X ; X , R
1 ; Y , L
0 ; 0 , L
Y ; Y , L
Y ; Y , R
q2
q3
□ ; □ , R
q4

Ejemplo

	0	1	X	Y	B
$\rightarrow q_0$	(q_1,X,D)			(q_3,Y,D)	
q_1	$(q_1,0,D)$	(q_2,Y,I)		(q_1,Y,D)	
q_2	$(q_2,0,I)$		(q_0,X,D)	(q_2,Y,I)	
q_3				(q_3,Y,D)	(q_4,B,D)
$*q_4$					

para $w = 01$ se tiene $q_001 \vdash Xq_11 \vdash q_2XY \vdash Xq_0Y \vdash XYq_3B \vdash XYBq_4B$

Lenguaje tipo 0

Sea **M** = (**Q**, Σ, Γ, δ, $\mathbf{q_0}$, **B**, **F**) una Máquina de Turing

Lenguaje aceptado se define:

$$L(M) = \{ w \mid w \text{ en } \Sigma^* , q_0 w \vdash^* \alpha_1 q \alpha_2, \alpha_1 \text{ y } \alpha_2 \text{ en } \Gamma^* \text{ y } q \text{ en } \mathbf{F} \}$$

L(M) es un lenguaje recursivamente

Lenguaje tipo 0

- Lenguaje **Recursivamente Enumerable**
  Existe una MT
- Lenguaje **Recursivo** (decidible = algoritmo)
  Existe una MT que siempre para

Utilidad

- **Aceptador** Lenguajes
  ¿w en L?
- **Generador** de Lenguajes
  01B0011B000111B…
- **Computador** de funciones matemáticas sobre enteros
  0^m10^n generar $0^{m\times n}$

Conclusiones

- MT es la máquina abstracta más **poderosa**
- MT consta de un **autómata finito** y una **cinta infinita**
- MT reconoce los lenguajes tipo 0 **recursivamente enumerables**
- Si MT se detiene para toda entrada, el **lenguaje es recursivo**

Teoría de la Computación y Lenguajes Formales

sesión 12

Unidad III: Máquinas de Turing y Computabilidad

Extensiones de la MT y Lenguajes recursivamente enumerables

Prof. Hilda Y. Contreras
Departamento de Computación
hyelitza@ula.ve
http://webdelprofesor.ula.ve/ingenieria/hyelitza
http://moodle2.ula.ve/course/view.php?id=359

Jerarquía de Chomsky

Tipo	Lenguaje	Máquina	Gramática G = (V,T,P,S)
0	Recursivamente enumerable	Máquina de Turing	Gramática sin restricciones $\alpha \rightarrow \beta$ (α, β en $(V \cup T)^*$, α contiene una variables)
1	Dependiente del Contexto	Autómata linealmente acotado	*Gramática sensible al contexto* $\alpha \rightarrow \beta$ (α, β en $(V \cup T)^*$, α contiene una variables, $\lvert\beta\rvert \geq \lvert\alpha\rvert$)
2	Independiente del Contexto	Autómata de Pila	*Gramática libre de contexto* $A \rightarrow \alpha$ (A en V y α en $(V \cup T)^*$)
3	Lenguaje	Autómata finito	*Gramática Regular* $A \rightarrow aB$

Contenido

- Técnicas de construcción de la MT:
  - Memoria en la unidad de control
  - Multipistas
  - Multicintas
- MT generadora de lenguajes y MT calculadora de funciones
- Lenguajes recursivamente enumerables
- Lenguajes tipo 1
- Gramáticas sin restricciones

Modelo básico de MT

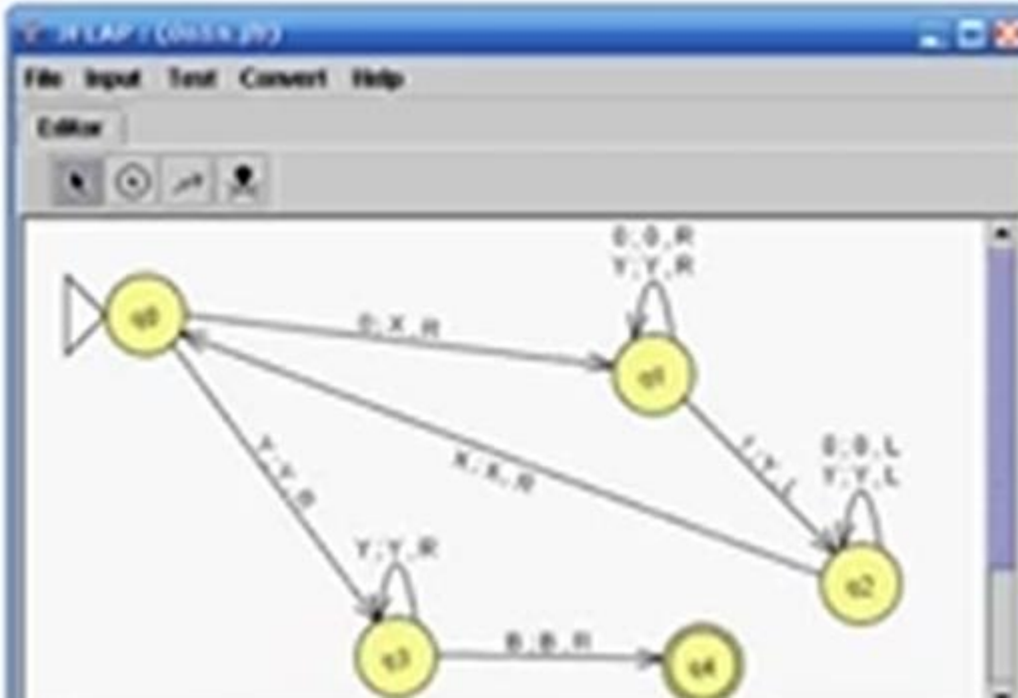
$M = (\mathbf{Q}, \Sigma, \Gamma, \delta, \mathbf{q_0}, \mathbf{B}, \mathbf{F})$

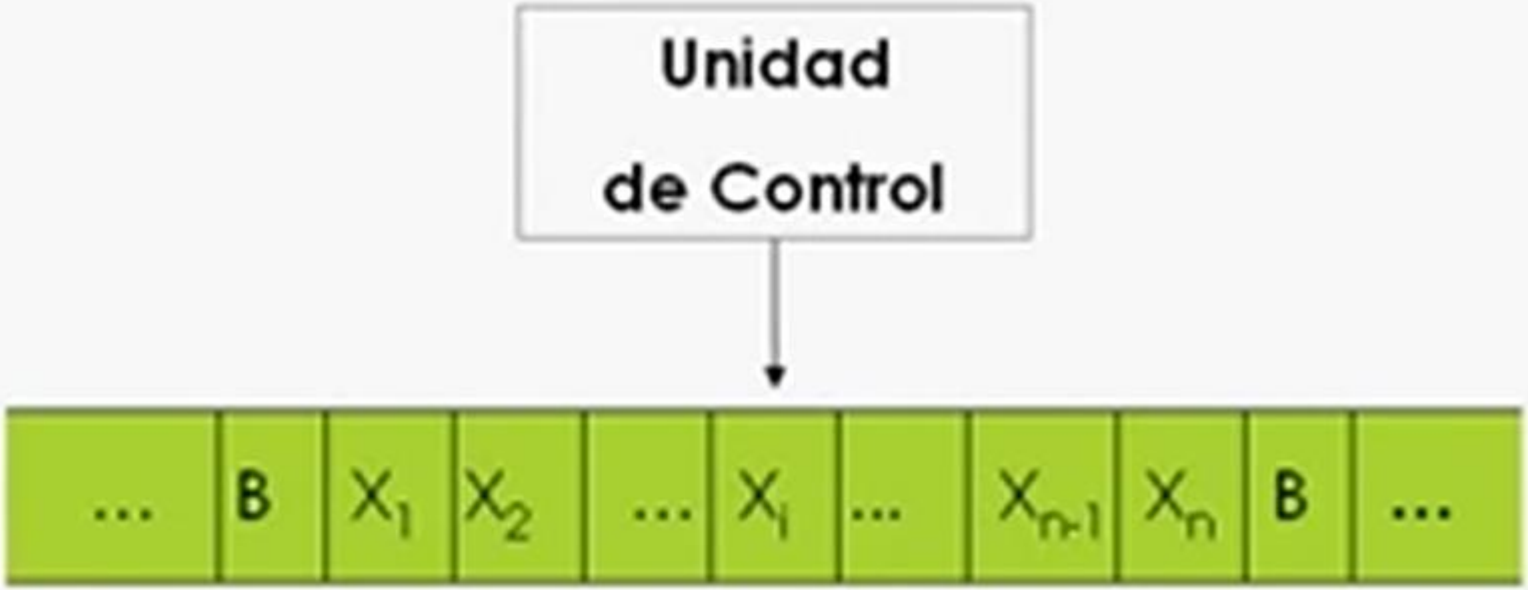
$\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{D, I\}$

$\delta(q, X_i) = (p, Y_i, D)$

$\mathbf{L = \{ 0^n 1^n \mid n \geq 1 \}}$

JFLAP: 0n1n iff





- Extensiones del modelo básico (una cinta, una pista, sin memoria en el control)
- Ninguna técnica aumenta el poder de la MT
- Técnicas facilitan la programación, disminuye el número de transiciones, mejora la comprensión del algoritmo, etc.

Unidad de Control
0

$$M = (Q', \Sigma, \Gamma, \delta, q_0, B, F)$$

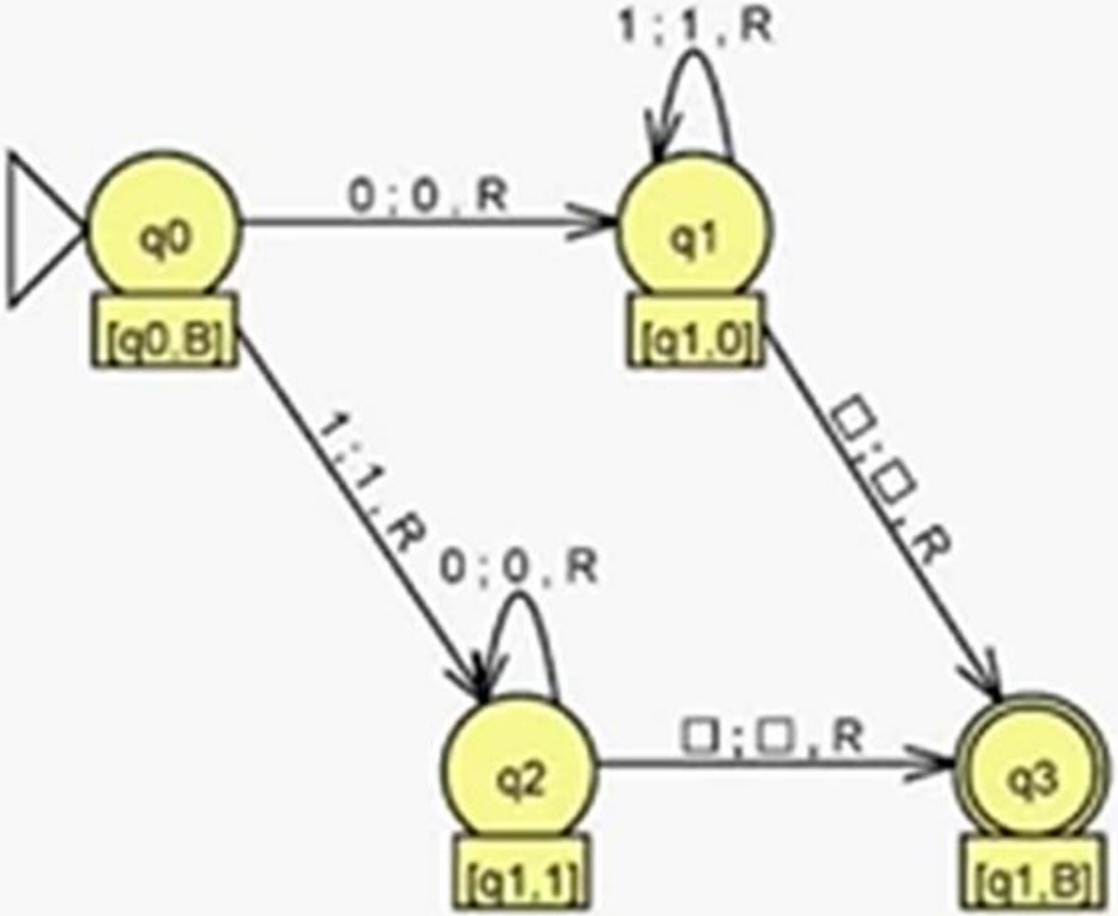
- Utiliza el estado de control para almacenar una cantidad **finita** de información.
- Conjunto Q de estados cambia

  $Q = \{q_0, q_1\}$ y $\Delta = \{0,1\}$ (Δ información finita)

  $Q' = Q_x\Delta = \{[q_0,0], [q_0,1], [q_1,0], [q_1,1]\}$

Por ejemplo: L = { 01* + 10* }

- Almacenar el primer símbolo leído
- $\Delta = \{0,1,B\}$
- JFLAP: 01a10a.jff



$\delta([q_0,B],0) = ([q_1,0],0,D)$

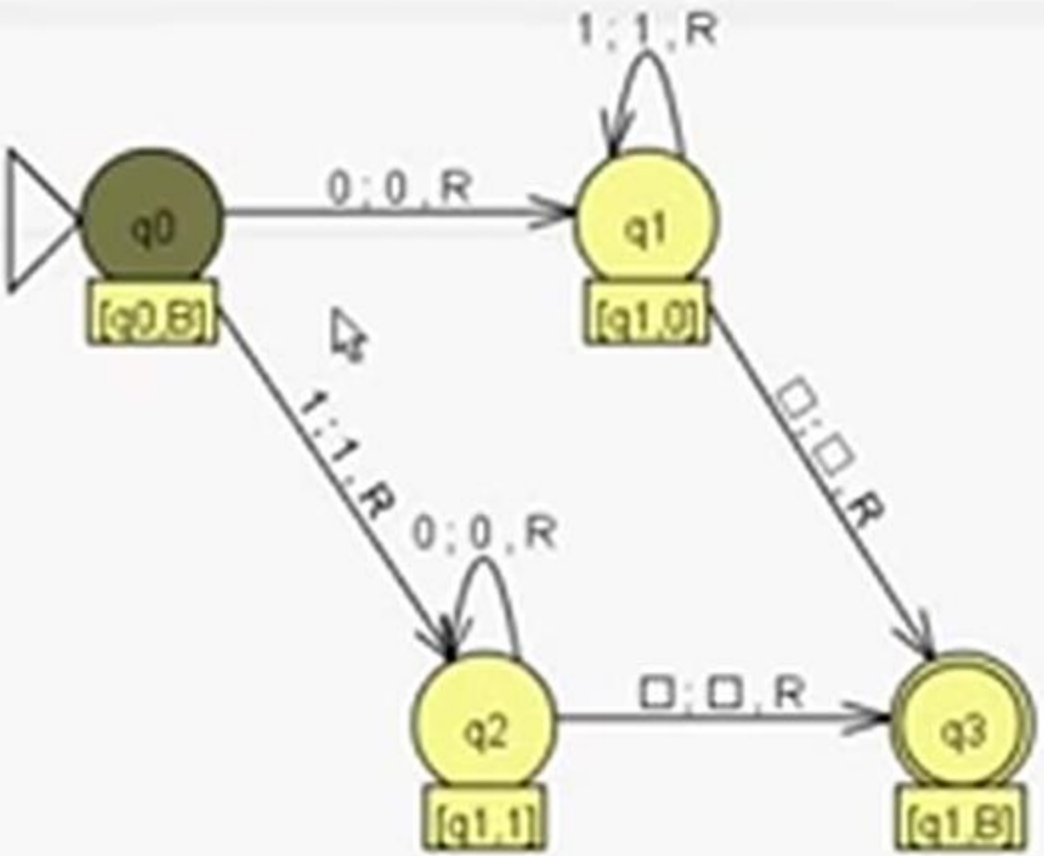
$\delta([q_1,0],1) = ([q_1,0],1,D)$

$\delta([q_1,0],B) = ([q_1,B],B,D)$

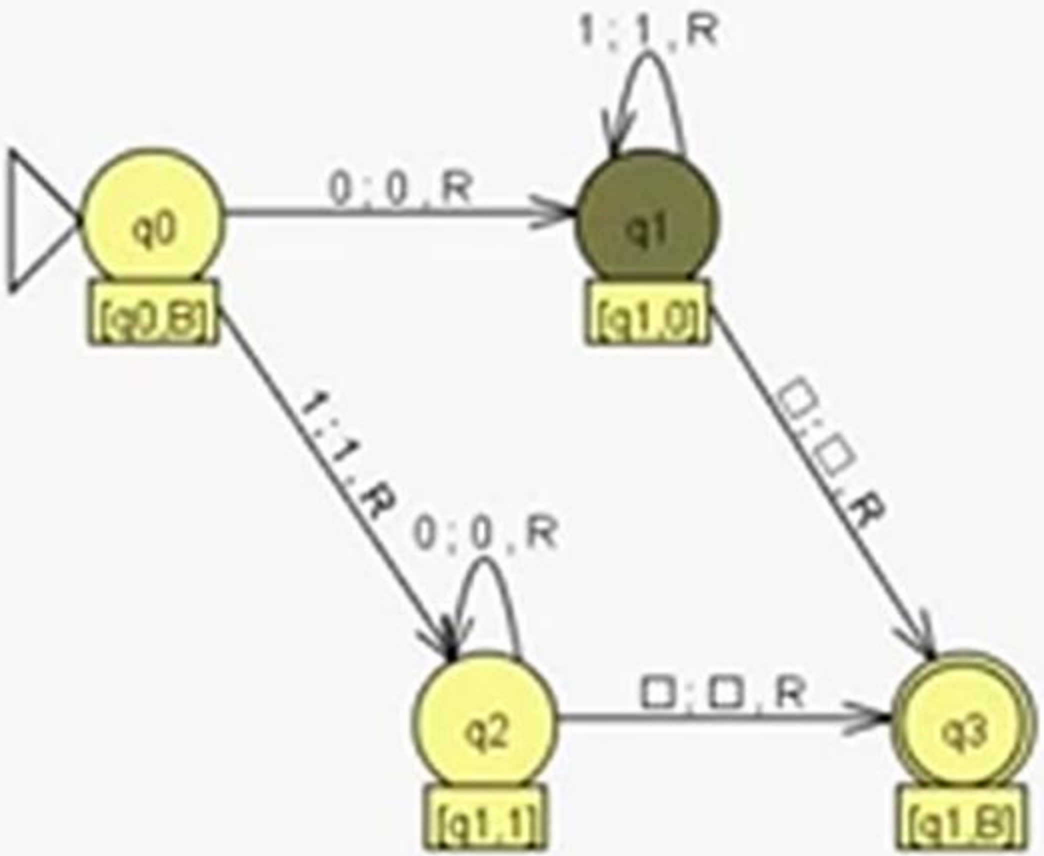
$\delta([q_0,B],1) = ([q_1,1],1,D)$

$\delta([q_1,1],0) = ([q_1,1],0,D)$

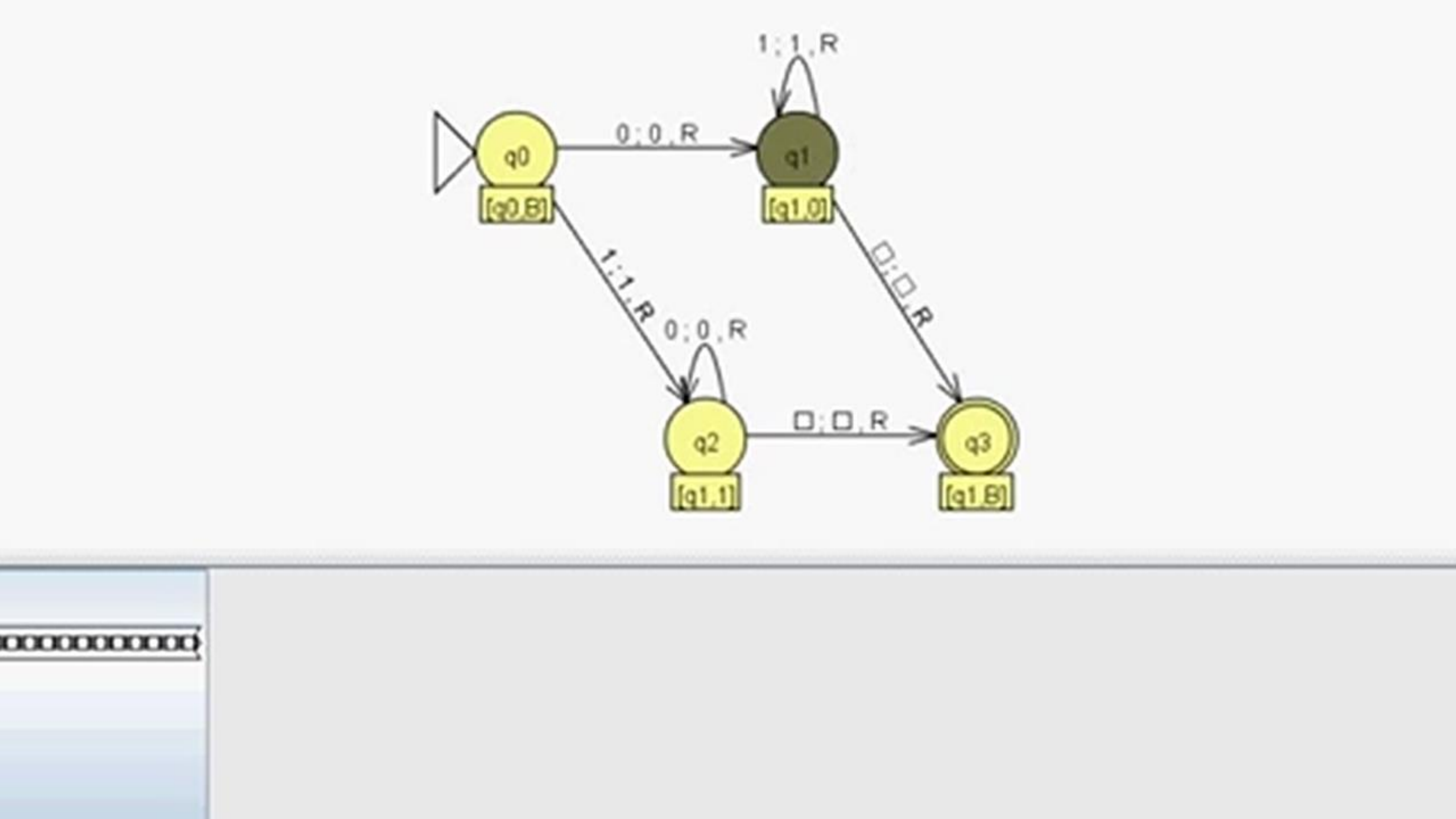
$\delta([q_1,1],B) = ([q_1,B],B,D)$

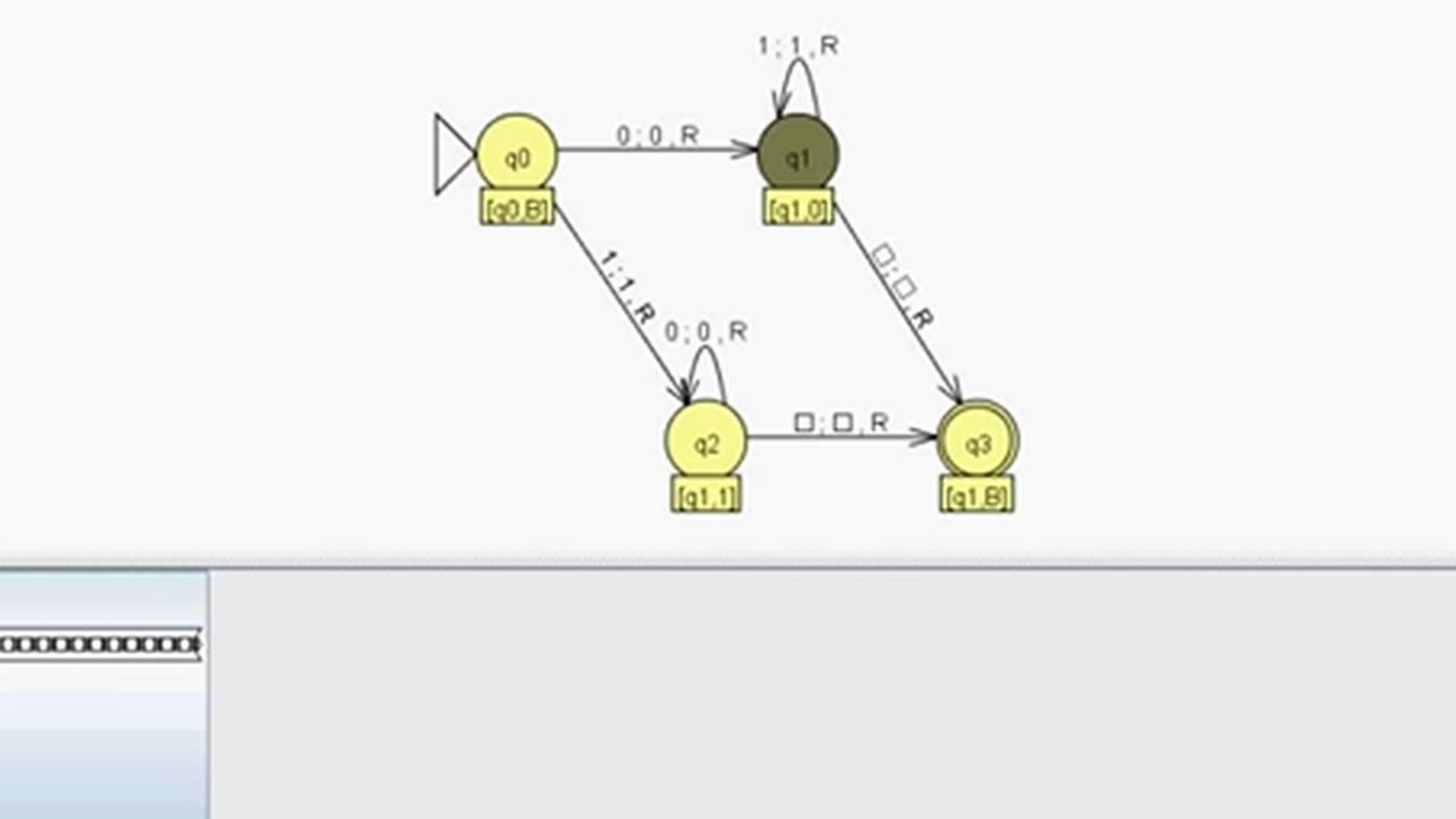


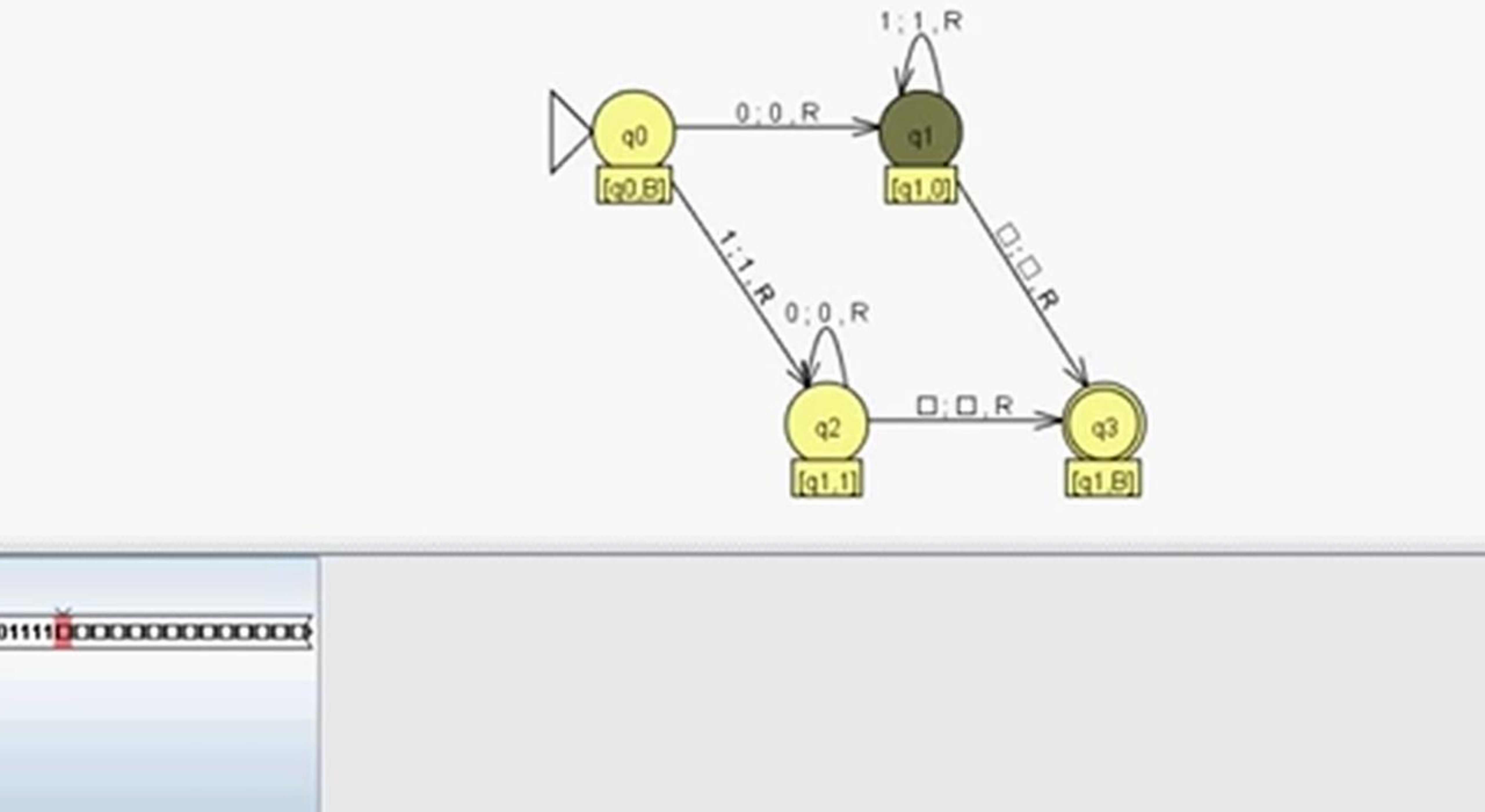
1 ; 1 , R
0 ; 0 , R
q0
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
□ ; □ , R
q3
[q1,1]
[q1,B]

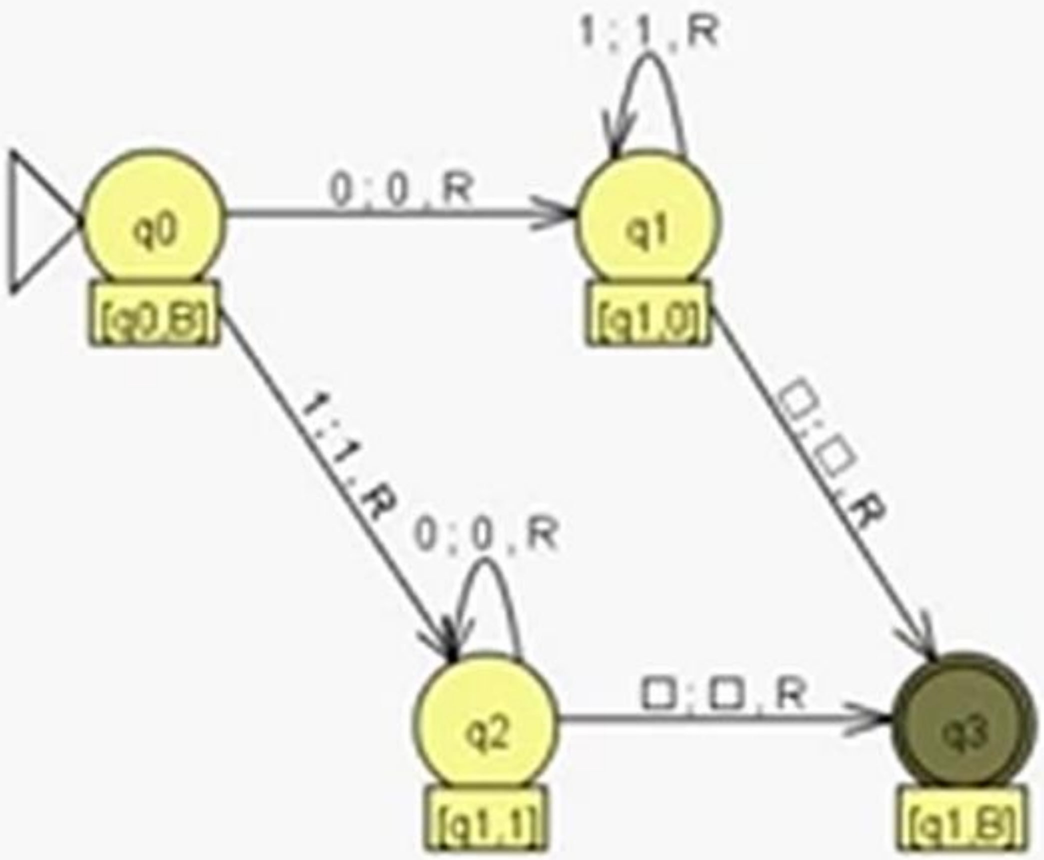


1;1,R
0;0,R
q0
q1
[q0,B]
[q1,0]
1;1,R
□;□,R
0;0,R
q2
□;□,R
q3
[q1,1]
[q1,B]

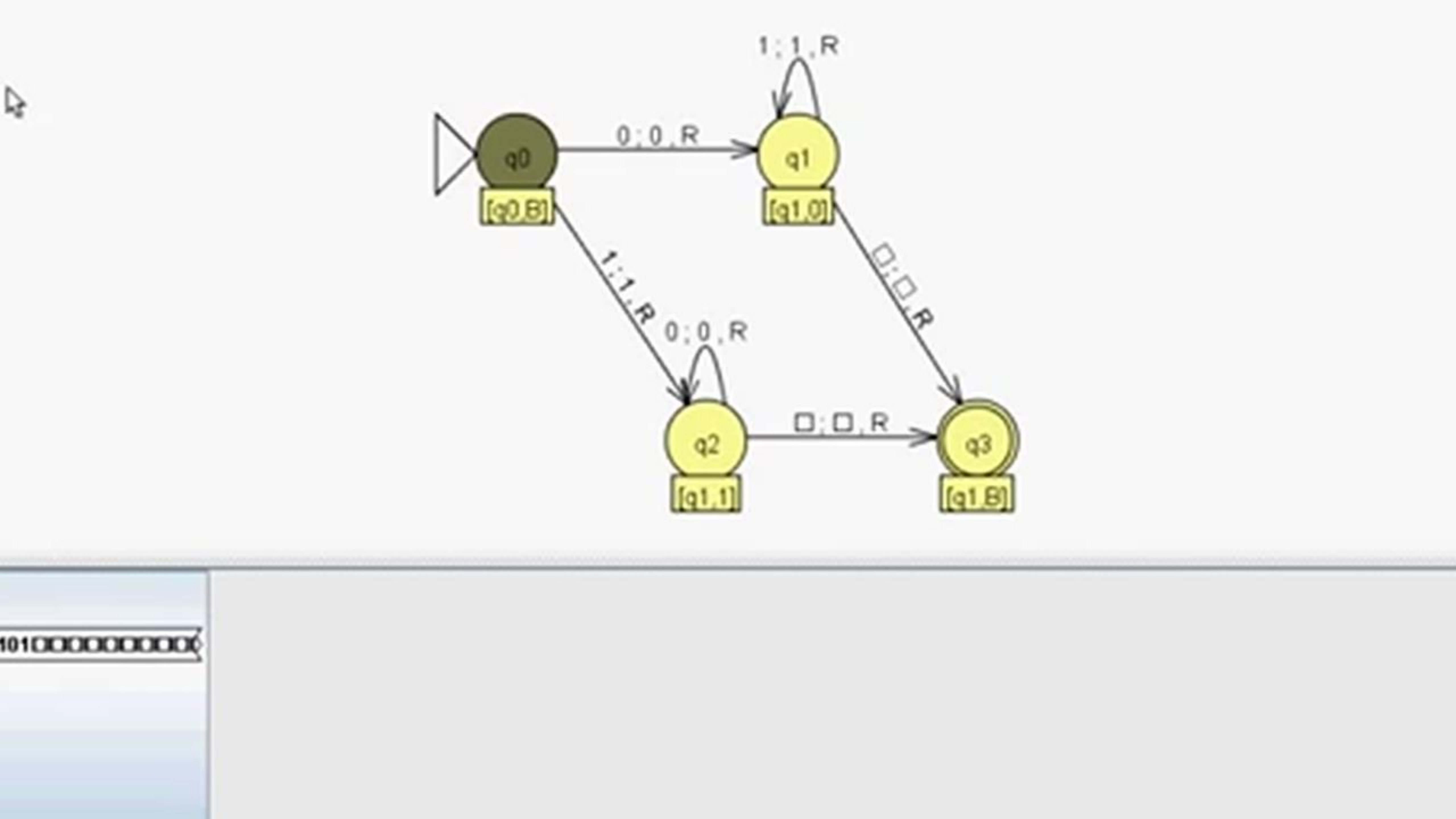

1;1,R
0;0,R
q0
[q0,B]
q1
[q1,0]
1;1,R
□;□,R
0;0,R
q2
[q1,1]
□;□,R
q3
[q1,B]


1 ; 1 , R
0 ; 0 , R
q0
[q0,B]
q1
[q1,0]
1 ; 1 , R
0 ; 0 , R
□ ; □ , R
q2
[q1,1]
□ ; □ , R
q3
[q1,B]

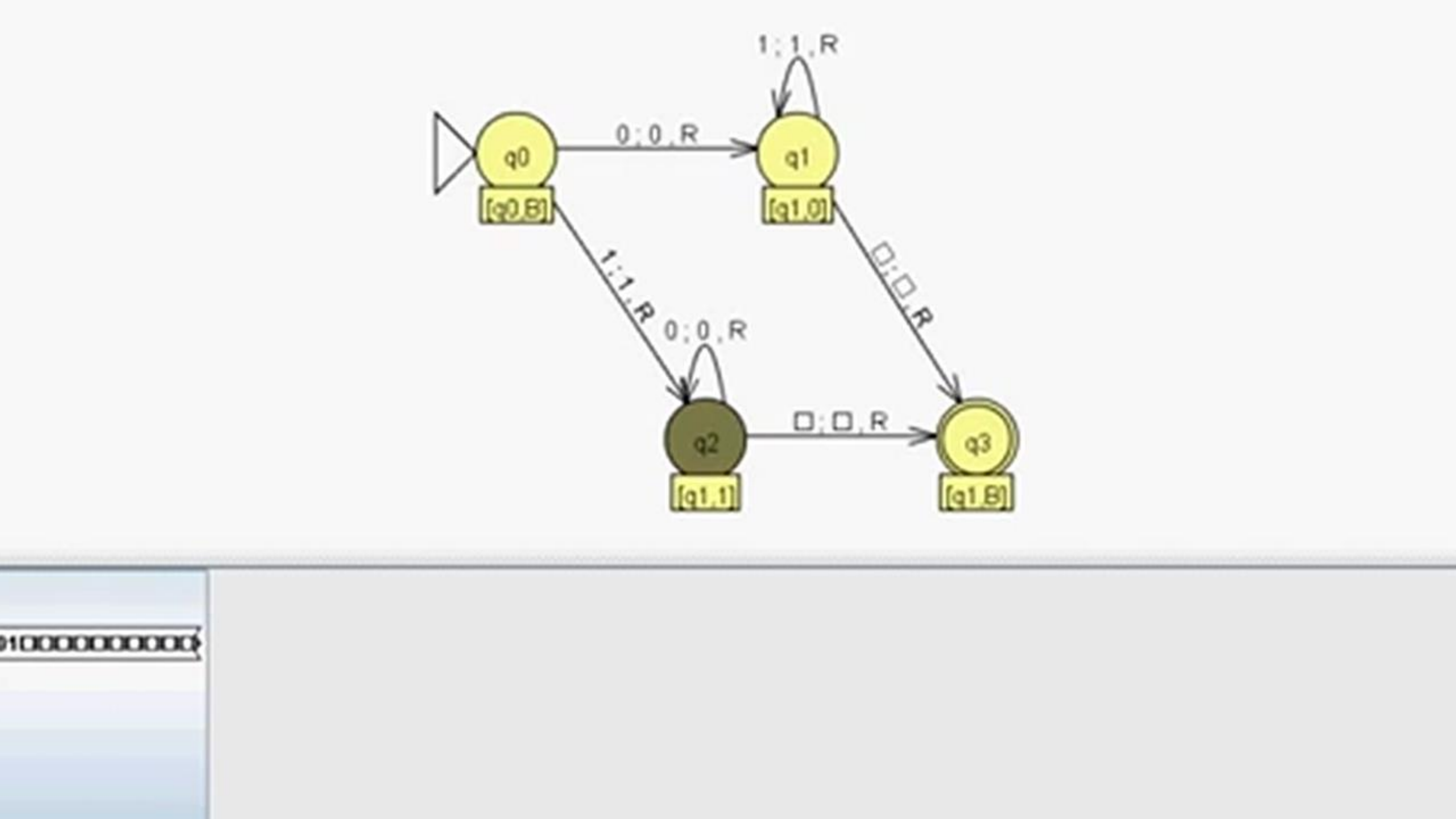

1 ; 1 , R
0 ; 0 , R
1 ; 1 , R
0 ; 0 , R
□ ; □ , R
□ ; □ , R
q0
[q0,B]
q1
[q1,0]
q2
[q1,1]
q3
[q1,B]
01111



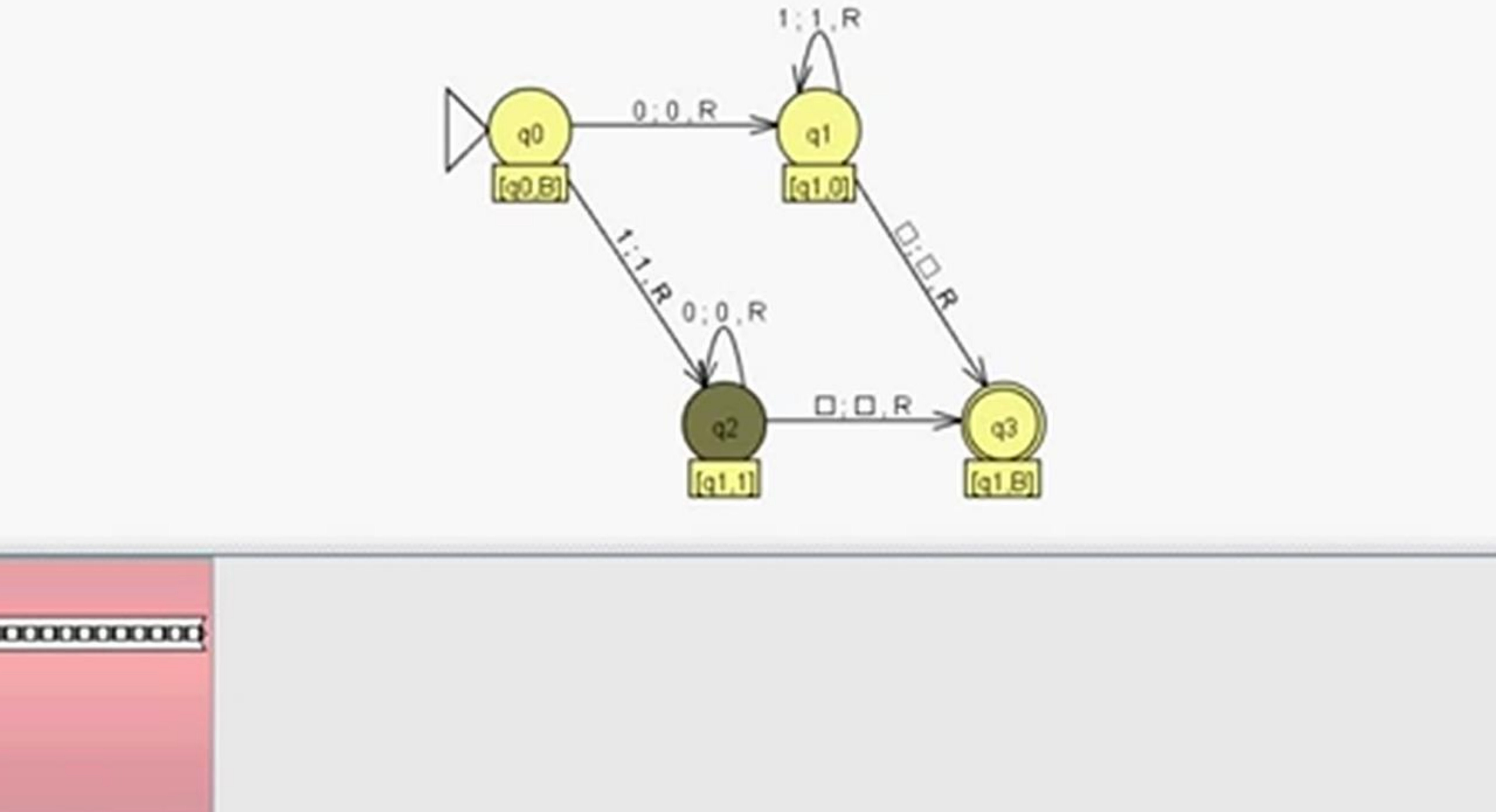
1 ; 1 , R
0 ; 0 , R
q0
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
q3
□ ; □ , R
[q1,1]
[q1,B]

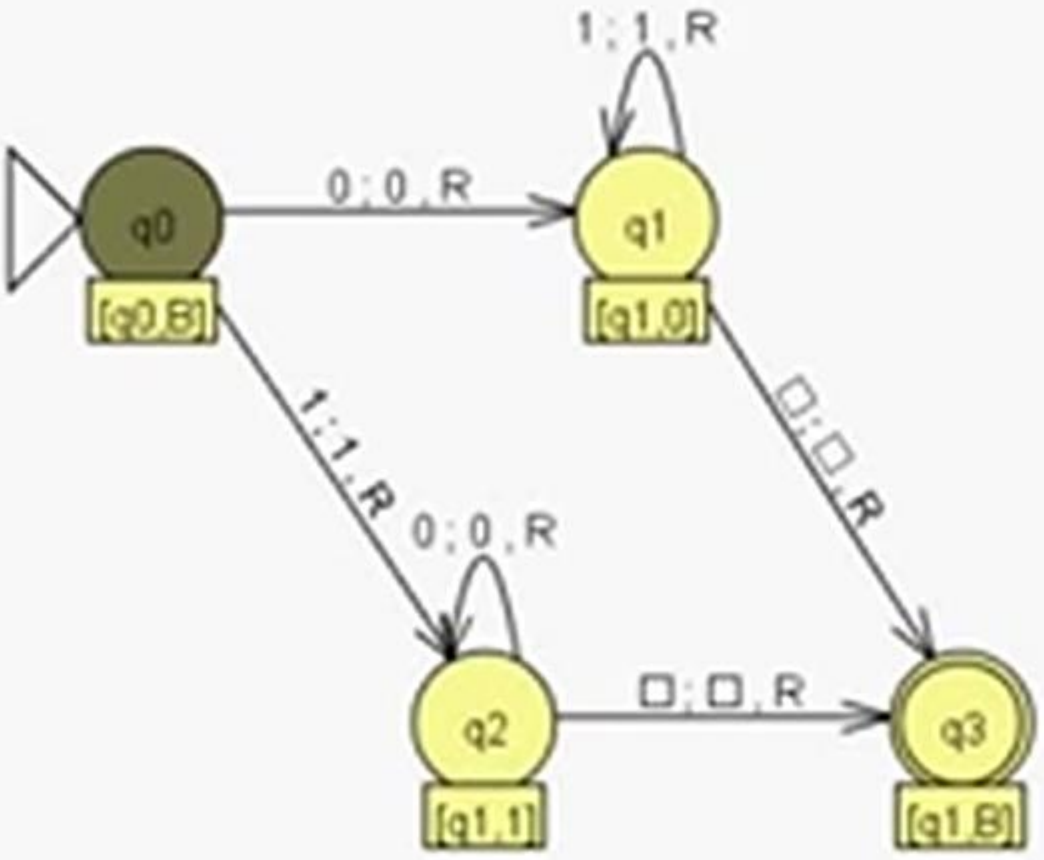


q0
[q0,B]
0;0,R
q1
[q1,0]
1;1,R
1;1,R
q2
[q1,1]
0;0,R
☐;☐,R
☐;☐,R
q3
[q1,B]
101☐☐☐☐☐☐☐☐☐☐☐

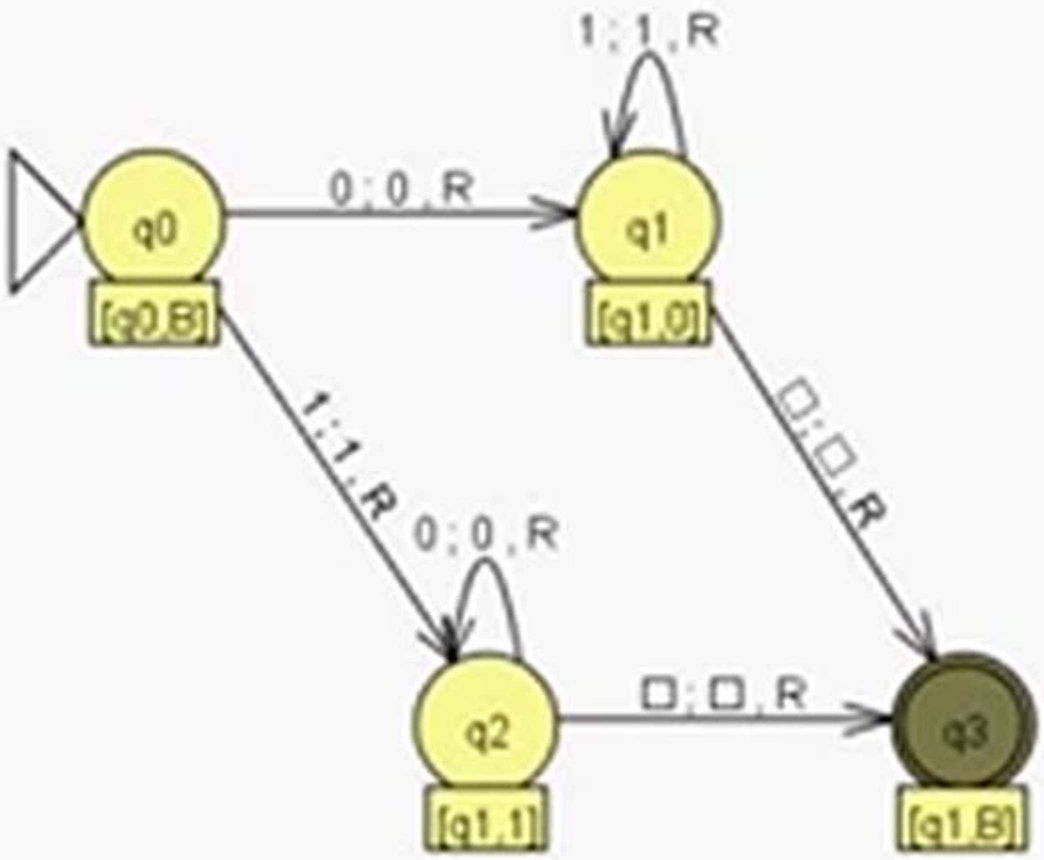


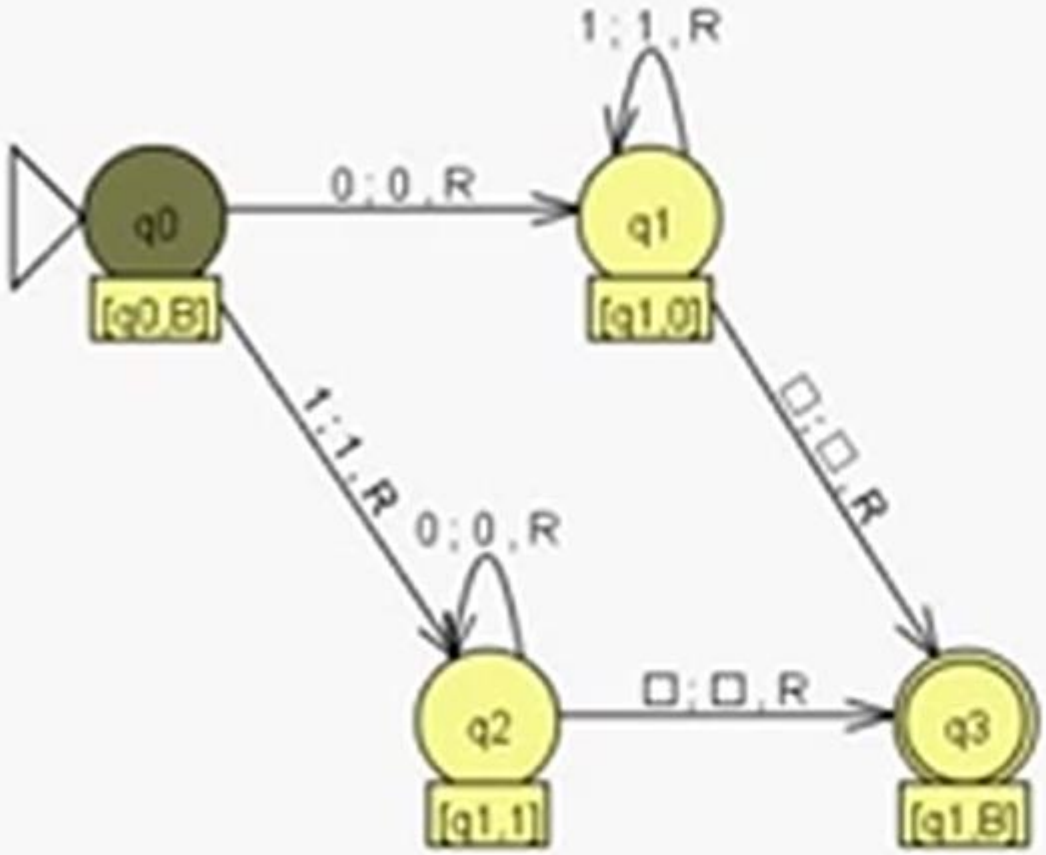
1;1,R
0;0,R
q0
q1
[q0,B]
[q1,0]
1;1,R
□;□,R
0;0,R
q2
□;□,R
q3
[q1,1]
[q1,B]

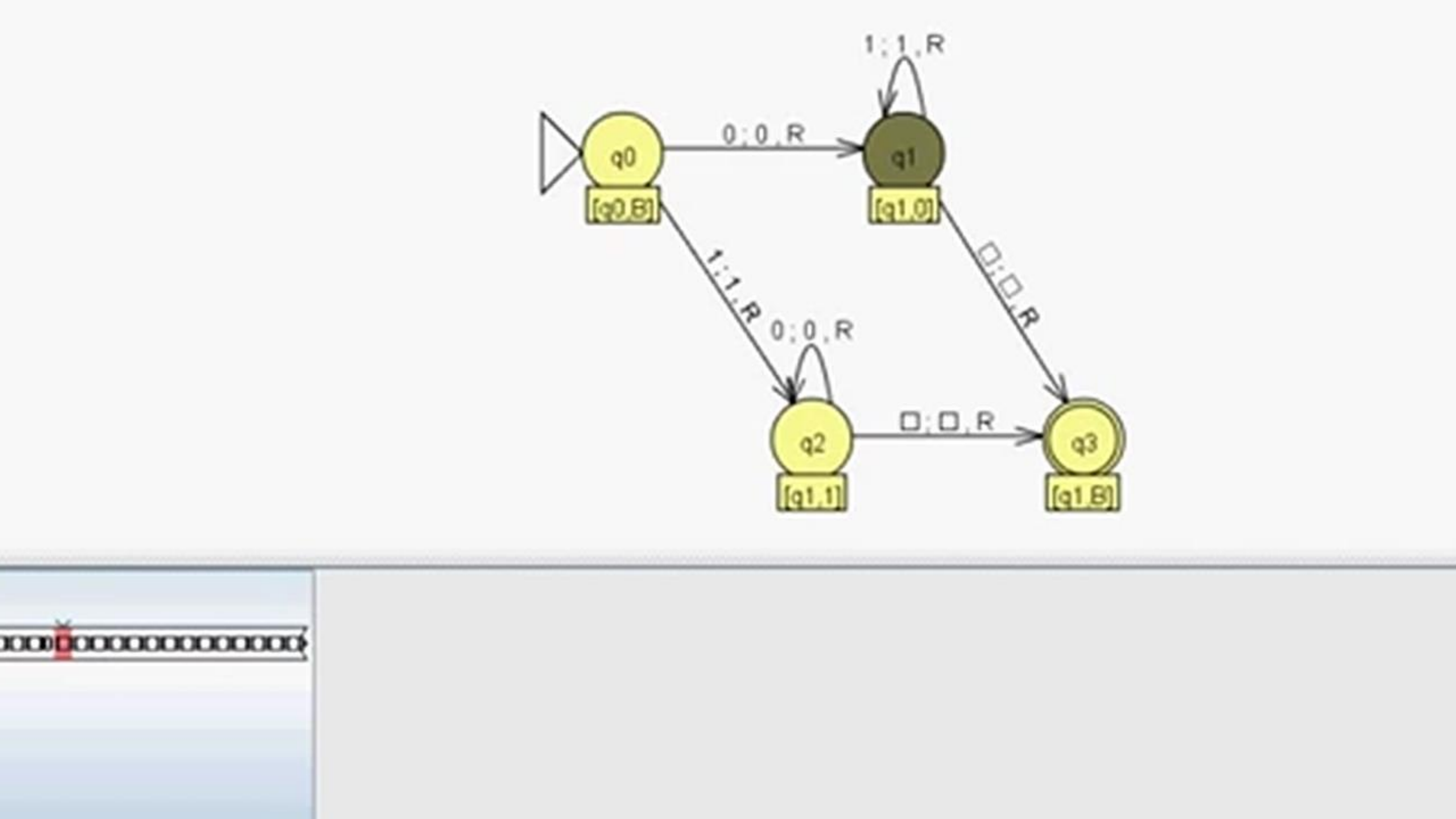

1 ; 1 , R
0 ; 0 , R
q0
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
□ ; □ , R
q3
[q1,1]
[q1,B]

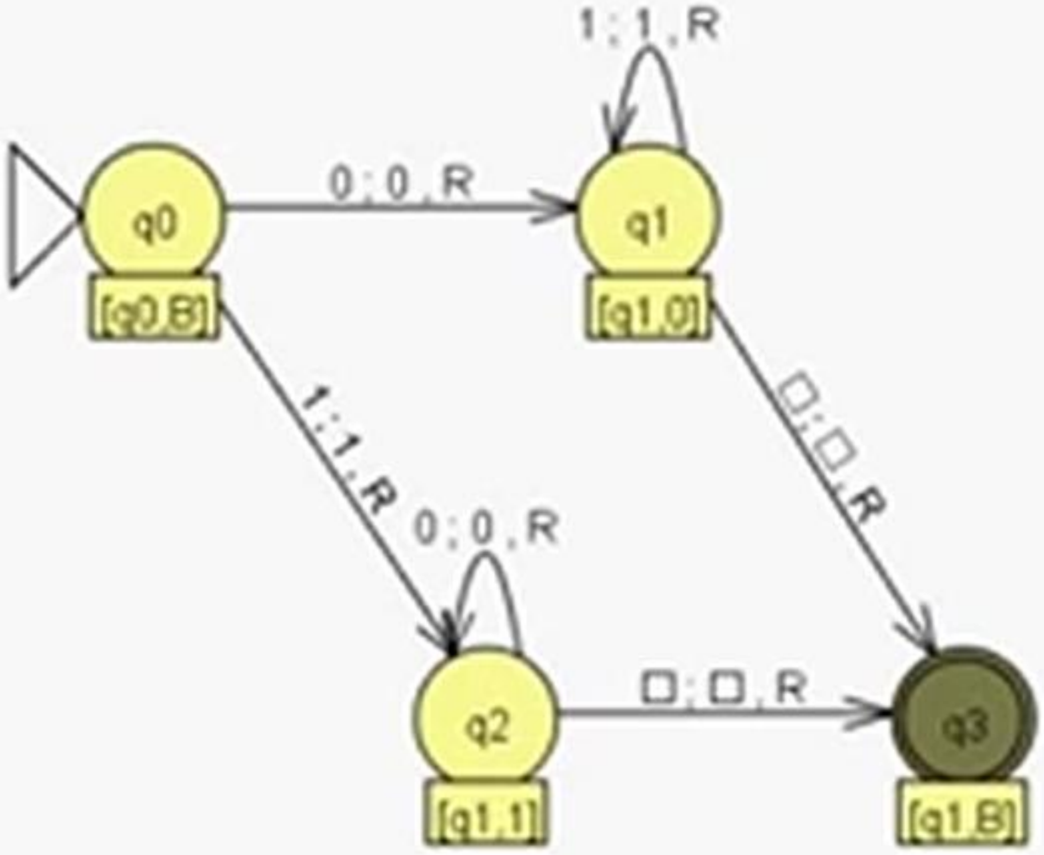


1 ; 1 , R
0 ; 0 , R
q0
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
□ ; □ , R
q3
[q1,1]
[q1,B]


1 ; 1 , R
0 ; 0 , R
q0
[q0,B]
q1
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
[q1,1]
□ ; □ , R
q3
[q1,B]


1;1,R
0;0,R
q0
q1
[q0,B]
[q1,0]
1;1,R
□;□,R
0;0,R
q2
q3
□;□,R
[q1,1]
[q1,B]

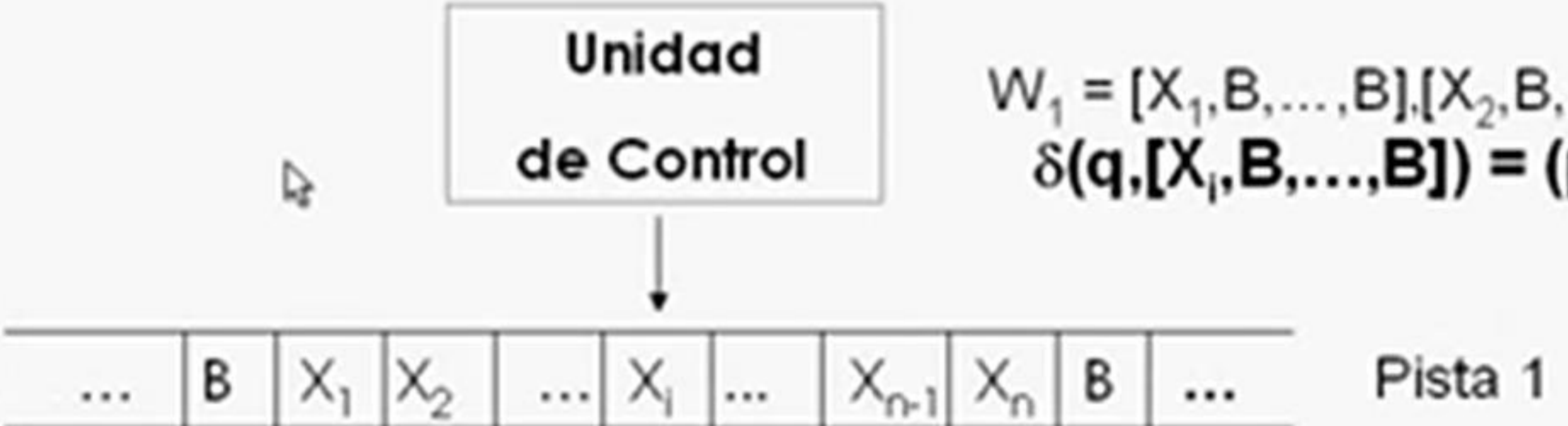

1 ; 1 , R
q0
0 ; 0 , R
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
□ ; □ , R
q3
[q1,1]
[q1,B]


1 ; 1 , R
q0
0 ; 0 , R
q1
[q0,B]
[q1,0]
1 ; 1 , R
□ ; □ , R
0 ; 0 , R
q2
□ ; □ , R
q3
[q1,1]
[q1,B]

MT con múltiples pistas

- Un solo cabezal. La cinta está dividida en un número finito de k pistas, la función de transición tiene la siguiente forma:

$$\delta: Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{D,I\}$$



$W_1 = [X_1,B,\ldots,B],[X_2,B,\ldots,B],\ldots\ [X_n,B,\ldots,B]$

$\boldsymbol{\delta(q,[X_i,B,\ldots,B]) = (p,[Y_i,B,\ldots,B],D)}$

MT con múltiples pistas

- Pistas múltiples (*multipistas*)
- La cinta almacena en cada celda un vector ***k*** dimensional de símbolos a los que se accede simultáneamente.

Un *movimiento* de la máquina implica:

- (a) Cambiar el estado del control finito
- (b) Escribir **k** símbolos en la celda analizada
- (c) Mover el cabezal de la cinta a la izquierda o a la derecha

- Hay k cintas diferentes y k cabezales. La función de transición para máquinas de Turing con n cintas:

$$\delta: Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{D,I,E\}^k$$

MT con múltiples cintas

- Un movimiento de la máquina *multicinta* depende del estado del control finito y de los símbolos analizados por cada cabezal de cada cinta
- Un movimiento de la máquina *multicinta* implica:

(a) Cambiar el estado del control finito

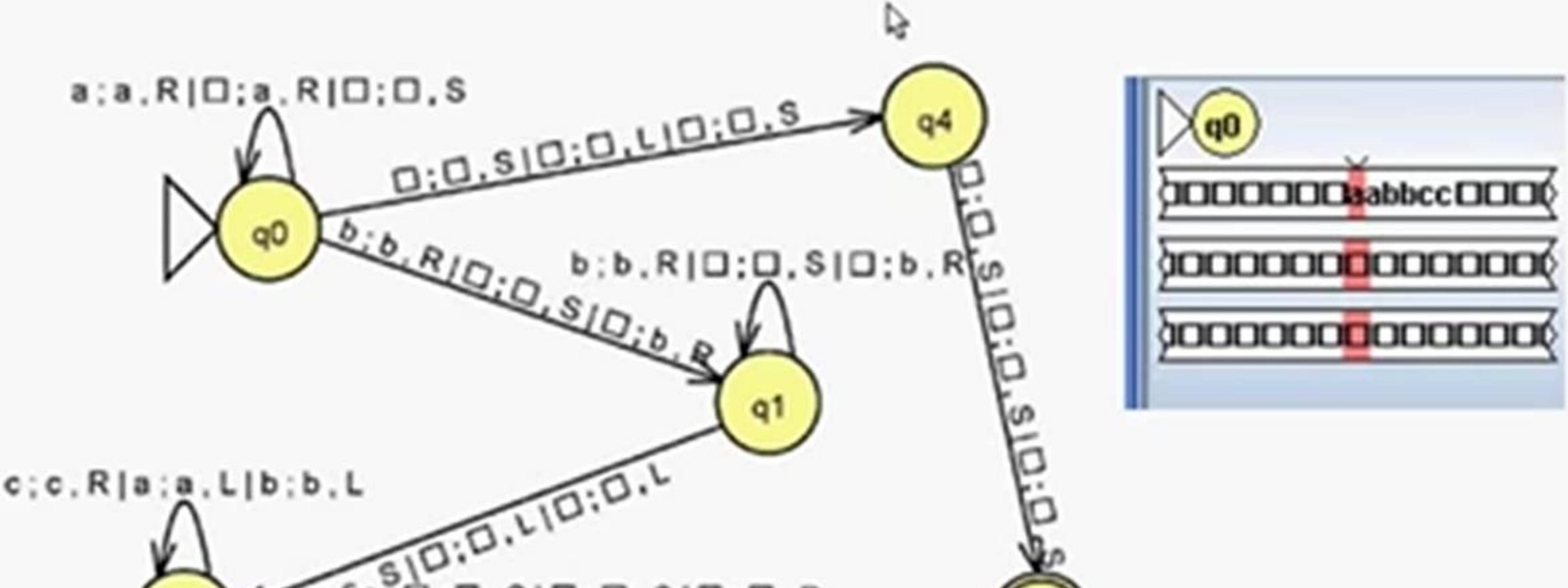
(b) Escribir un símbolo en cada una de las celdas analizadas

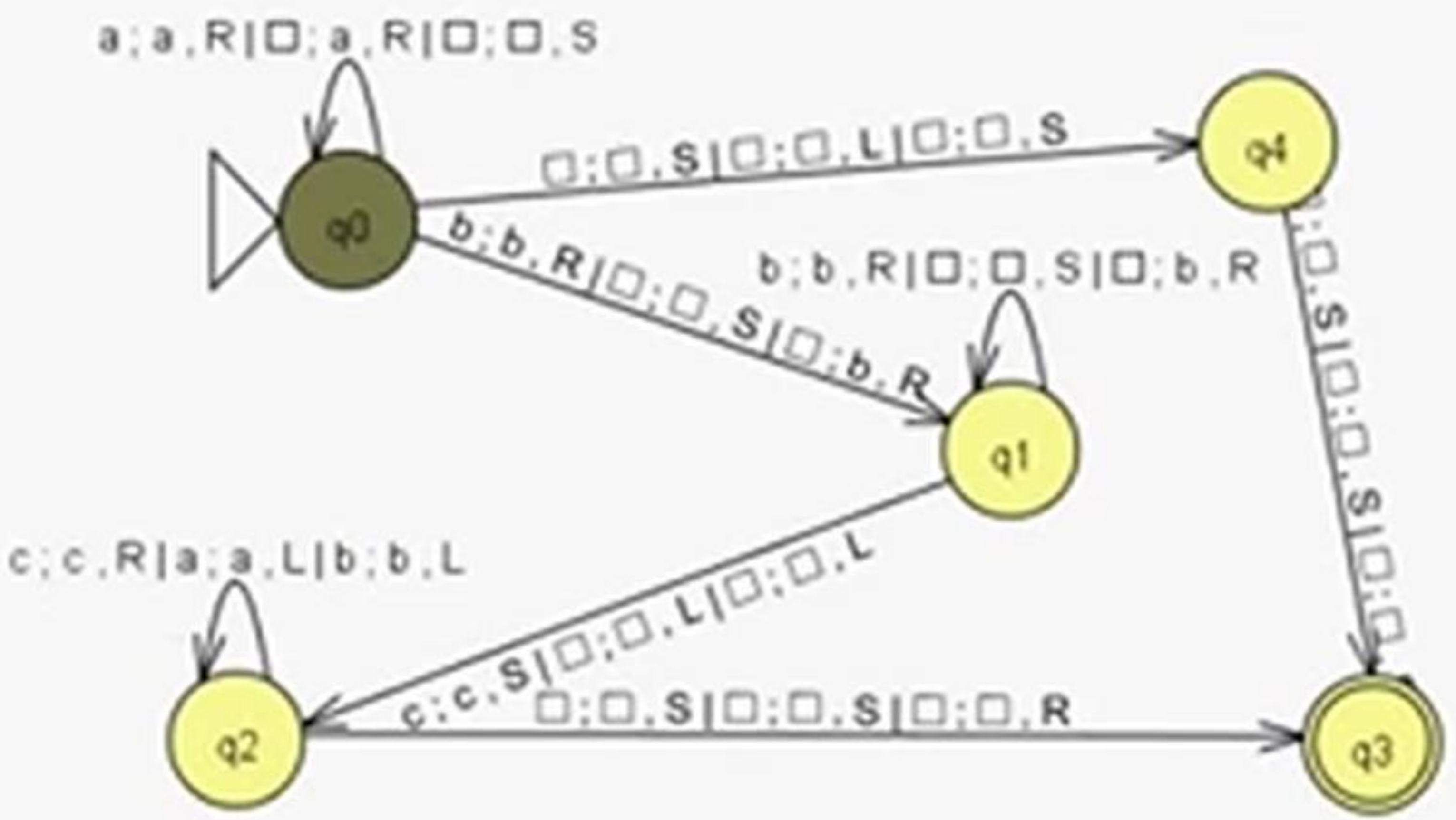
(c) Mover cada cabezal de cinta a la izquierda o

MT con múltiples cintas

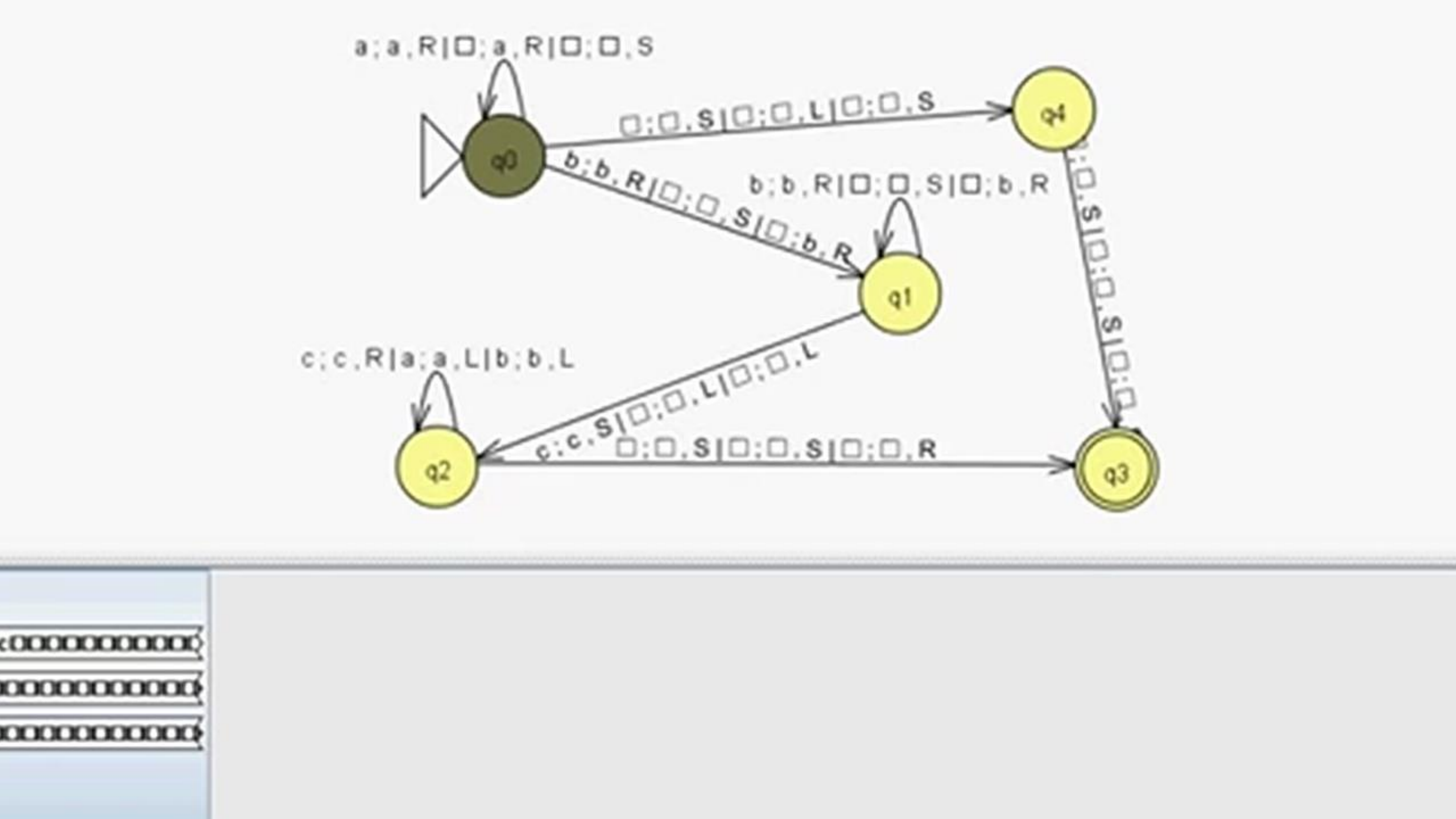
- Por ejemplo: L = { $a^n b^n c^n$ | $n \geq 0$ }

JFLAP: [turingAnBnCnMulti.jff]

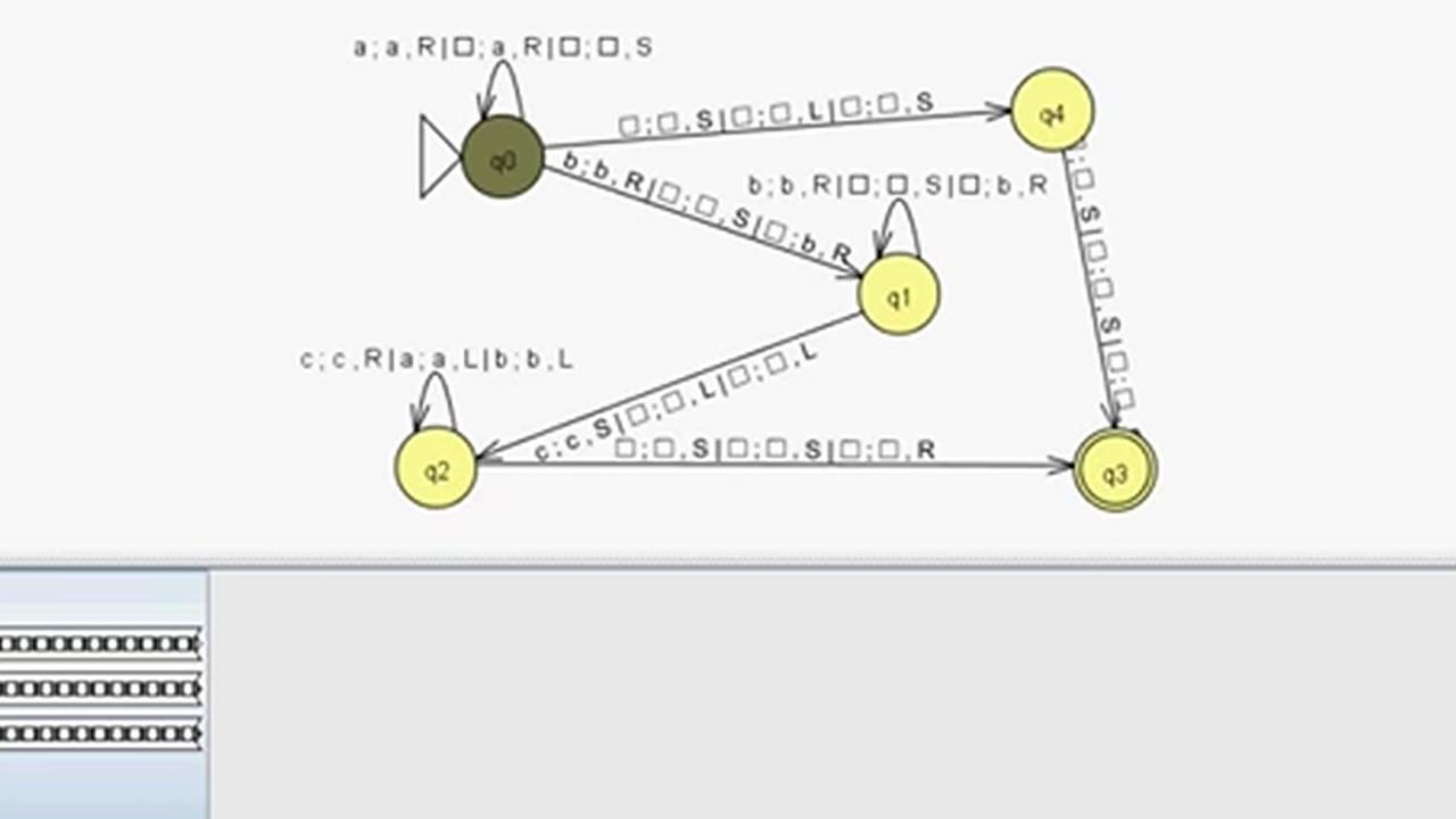


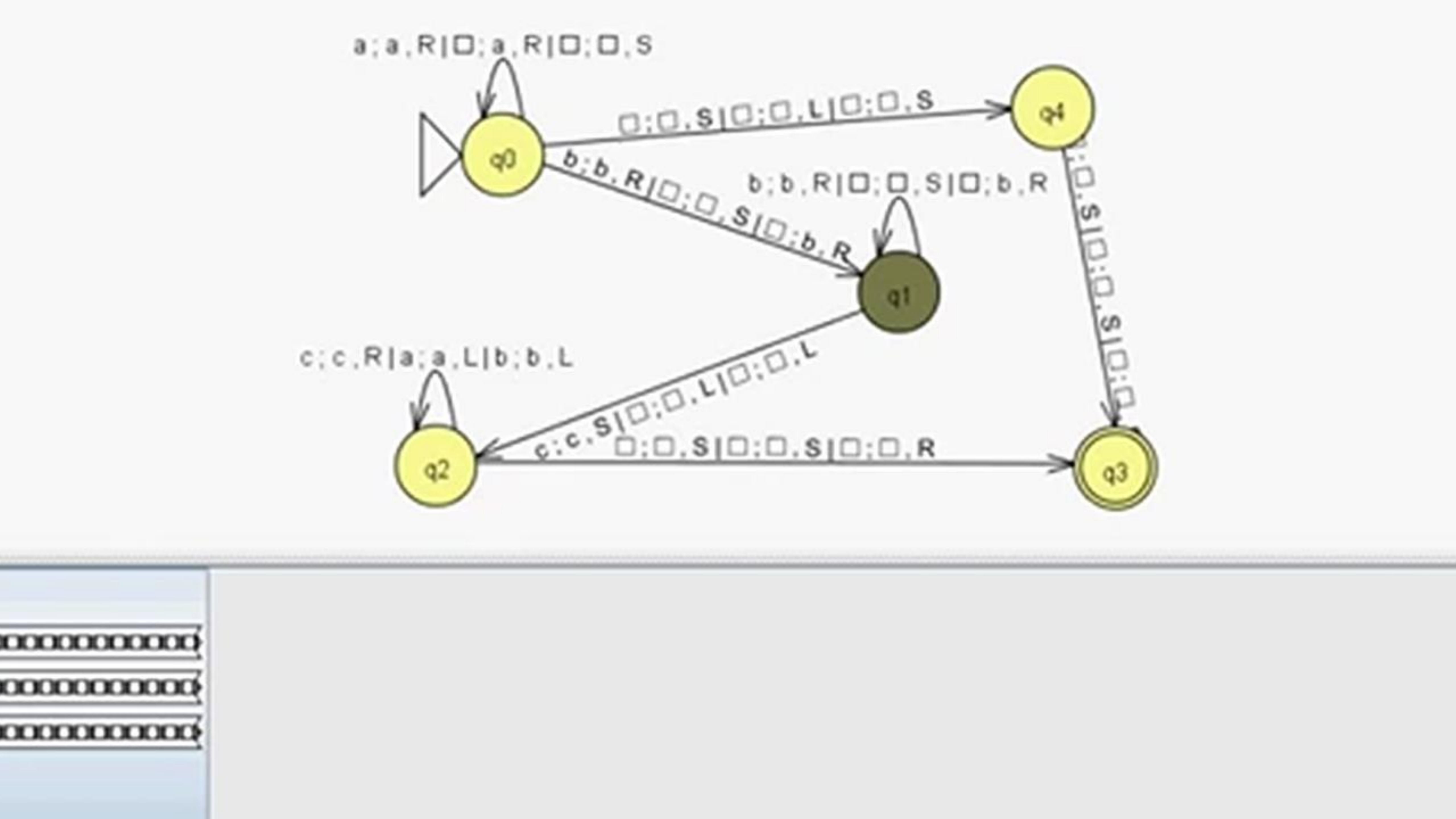


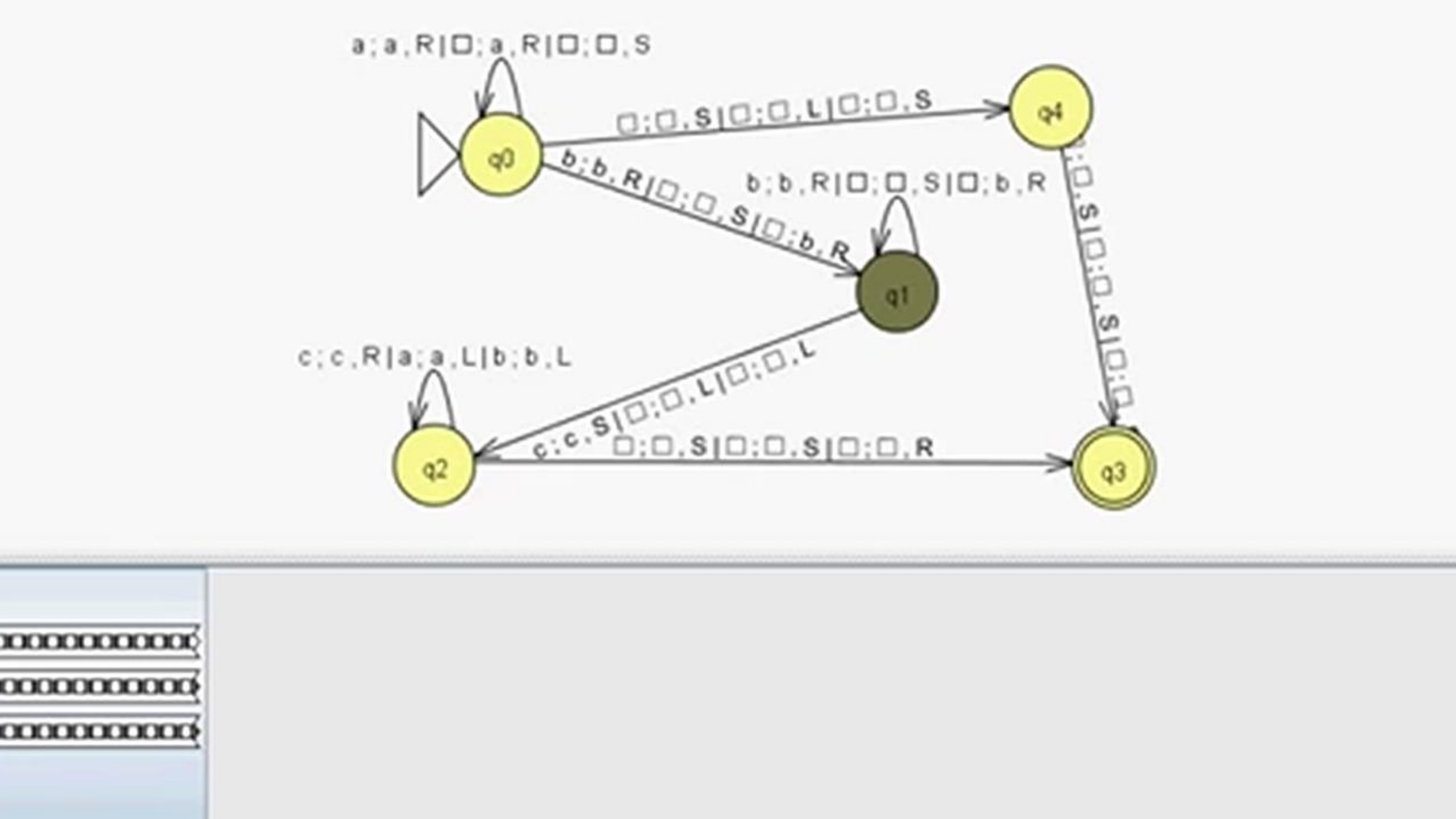
a;a,R|☐;a,R|☐;☐,S
☐;☐,S|☐;☐,L|☐;☐,S
b;b,R|☐;☐,S|☐;b,R
b;b,R|☐;☐,S|☐;b,R
c;c,R|a;a,L|b;b,L
c;c,S|☐;☐,L|☐;☐,L
☐;☐,S|☐;☐,S|☐;☐,R
q0
q1
q2
q3
q4



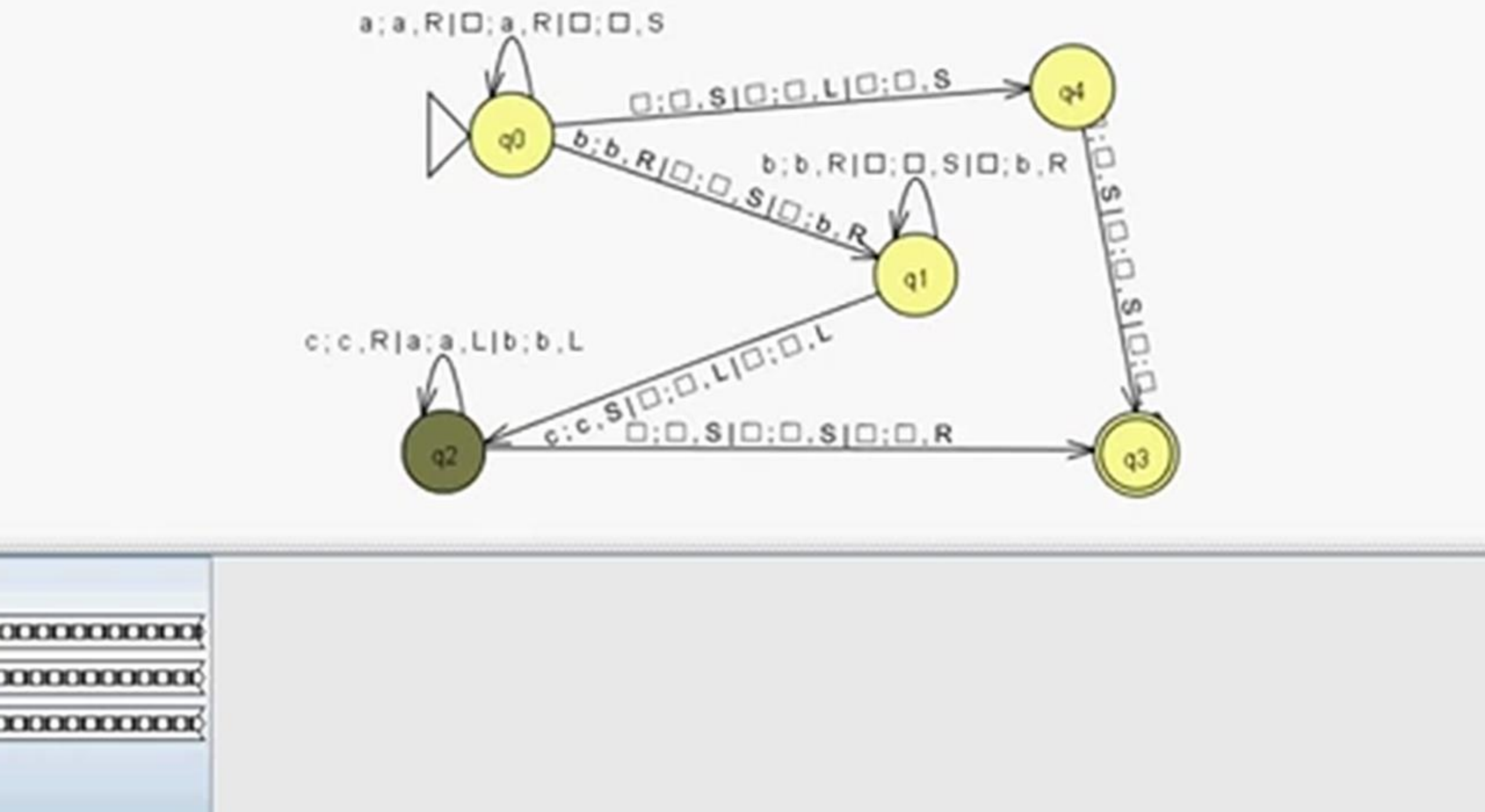
a;a,R|□;a,R|□;□,S
□;□,S|□;□,L|□;□,S
b;b,R|□;□,S|□;b,R
b;b,R|□;□,S|□;b,R
c;c,R|a;a,L|b;b,L
c;c,S|□;□,L|□;□,L
□;□,S|□;□,S|□;□,R
□;□,S|□;□,S|□;□
q0
q1
q2
q3
q4

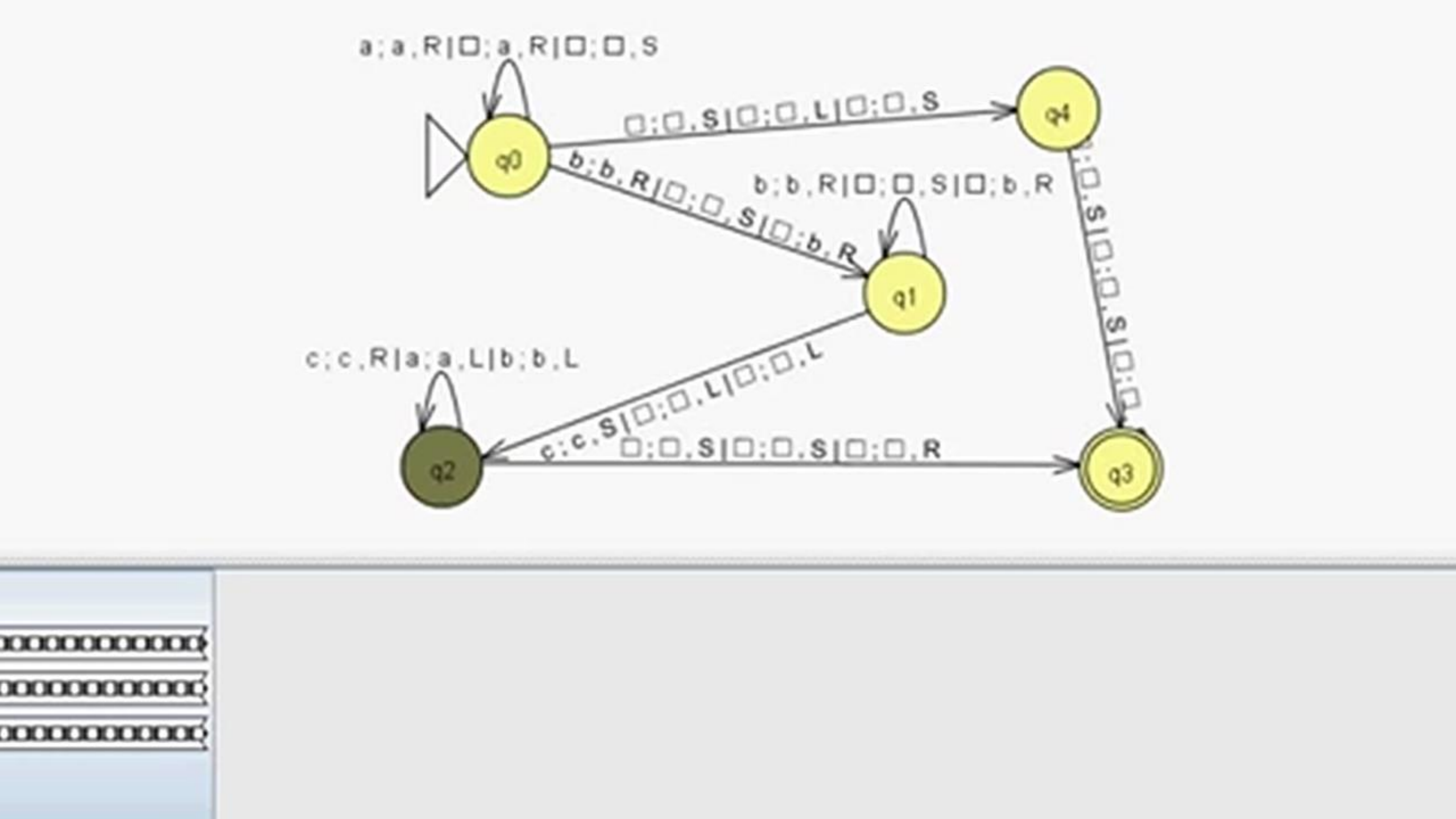

a;a,R|□;a,R|□;□,S
□;□,S|□;□,L|□;□,S
q0
q4
b;b,R|□;□,S|□;b,R
b;b,R|□;□,S|□;b,R
q1
c;c,R|a;a,L|b;b,L
c;c,S|□;□,L|□;□,L
□;□,S|□;□,S|□;□,R
q2
q3

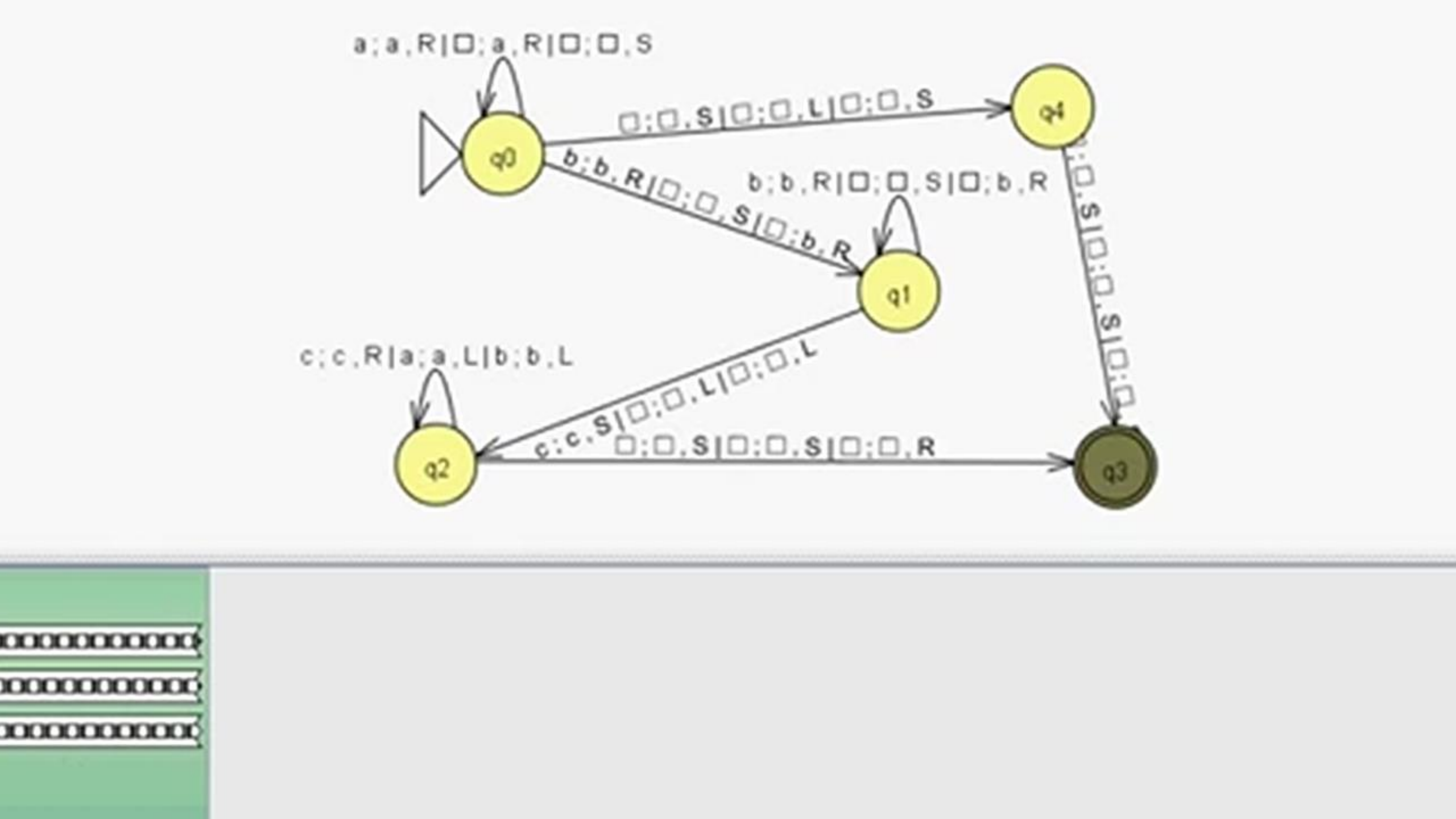

a;a,R|□;a,R|□;□,S
□;□,S|□;□,L|□;□,S
q0
q4
b;b,R|□;□,S|□;b,R
b;b,R|□;□,S|□;b,R
q1
c;c,R|a;a,L|b;b,L
c;c,S|□;□,L|□;□,L
□;□,S|□;□,S|□;□,R
q2
q3



a;a,R|□;a,R|□;□,S
□;□,S|□;□,L|□;□,S
q0
q4
b;b,R|□;□,S|□;b,R
b;b,R|□;□,S|□;b,R
q1
□,S|□;□,S|□;□
c;c,R|a;a,L|b;b,L
c;c,S|□;□,L|□;□,L
□;□,S|□;□,S|□;□,R
q2
q3


a ; a , R | □ ; a , R | □ ; □ , S
q0
□ ; □ , S | □ ; □ , L | □ ; □ , S
q4
b ; b , R | □ ; □ , S | □ ; b , R
b ; b , R | □ ; □ , S | □ ; b , R
q1
□ ; □ , S | □ ; □ , S | □ ; □
c ; c , R | a ; a , L | b ; b , L
c ; c , S | □ ; □ , L | □ ; □ , L
q2
□ ; □ , S | □ ; □ , S | □ ; □ , R
q3


a ; a , R | □ ; a , R | □ ; □ , S
q0
□ ; □ , S | □ ; □ , L | □ ; □ , S
q4
b ; b , R | □ ; □ , S | □ ; b , R
b ; b , R | □ ; □ , S | □ ; b , R
q1
□ ; □ , S | □ ; □ , S | □ ; □
c ; c , R | a ; a , L | b ; b , L
c ; c , S | □ ; □ , L | □ ; □ , L
q2
□ ; □ , S | □ ; □ , S | □ ; □ , R
q3


a ; a , R | □ ; a , R | □ ; □ , S
q0
□ ; □ , S | □ ; □ , L | □ ; □ , S
q4
b ; b , R | □ ; □ , S | □ ; b , R
b ; b , R | □ ; □ , S | □ ; b , R
q1
c ; c , R | a ; a , L | b ; b , L
c ; c , S | □ ; □ , L | □ ; □ , L
q2
□ ; □ , S | □ ; □ , S | □ ; □ , R
□ , S | □ ; □ , S | □ ; □
q3

MT calculadora

- Calcula funciones f: $\Sigma^* \rightarrow \Gamma^*$, es decir $f(w) = v$, donde w en Σ^* y v en Γ^*
- $q_0 w \vdash^* q_f v$, donde $v = f(w)$
- Uso de notación “unaria” para codificar los valores del dominio y el rango.

  $X = 0^x$, p.e. $3 = 0^3 = 000$
- No tiene estados de aceptación

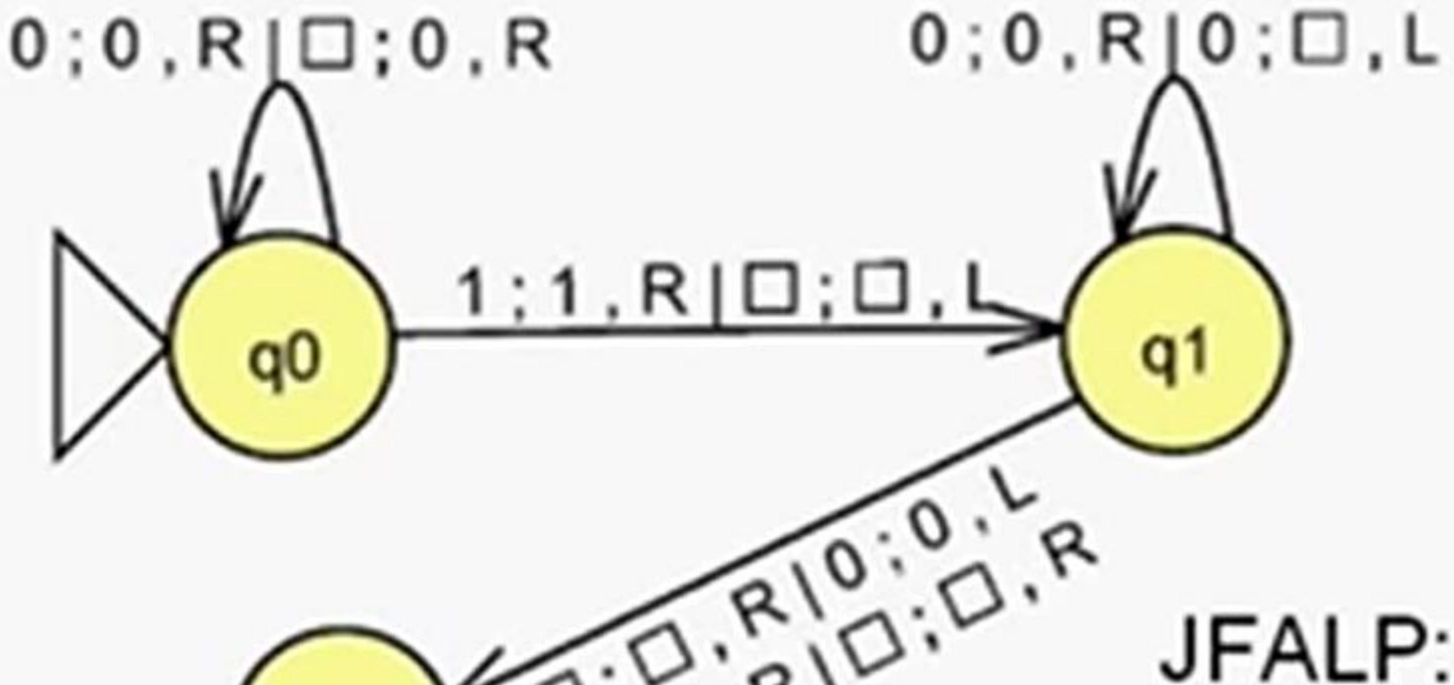
- Por ejemplo: sustracción propia

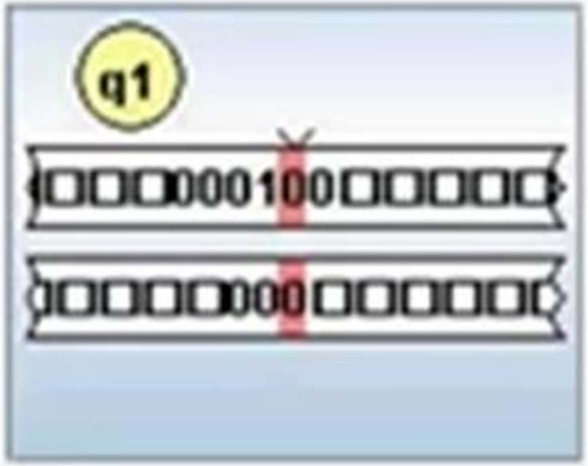
$$\mathbf{m - n} \begin{cases} m-n & \text{si } m \geq n \\ 0 & \text{si } m < n \end{cases}$$

p.e: operandos 3 y 2 para la operación 3 -

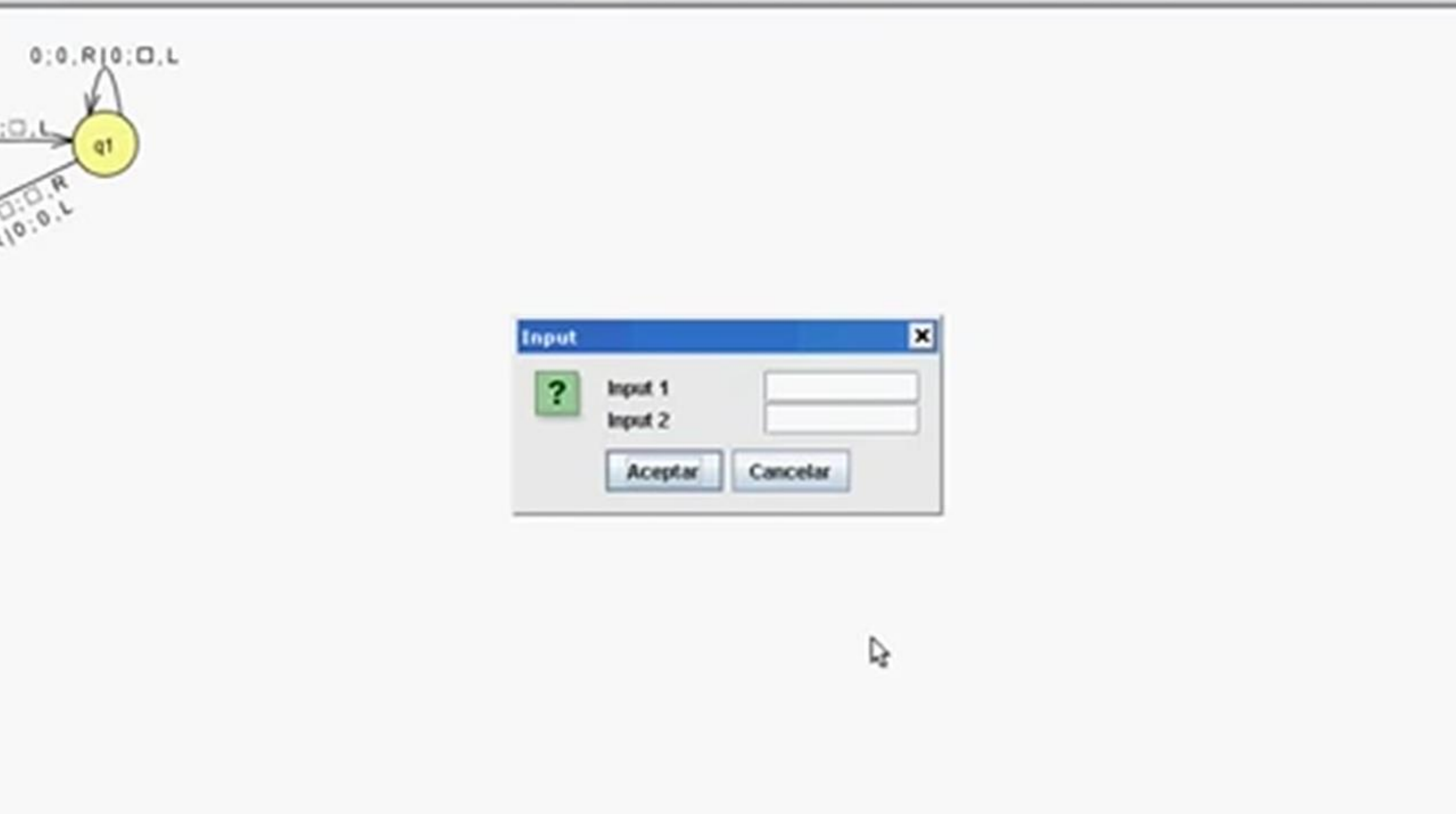
Entrada = 000100

Salida = 0



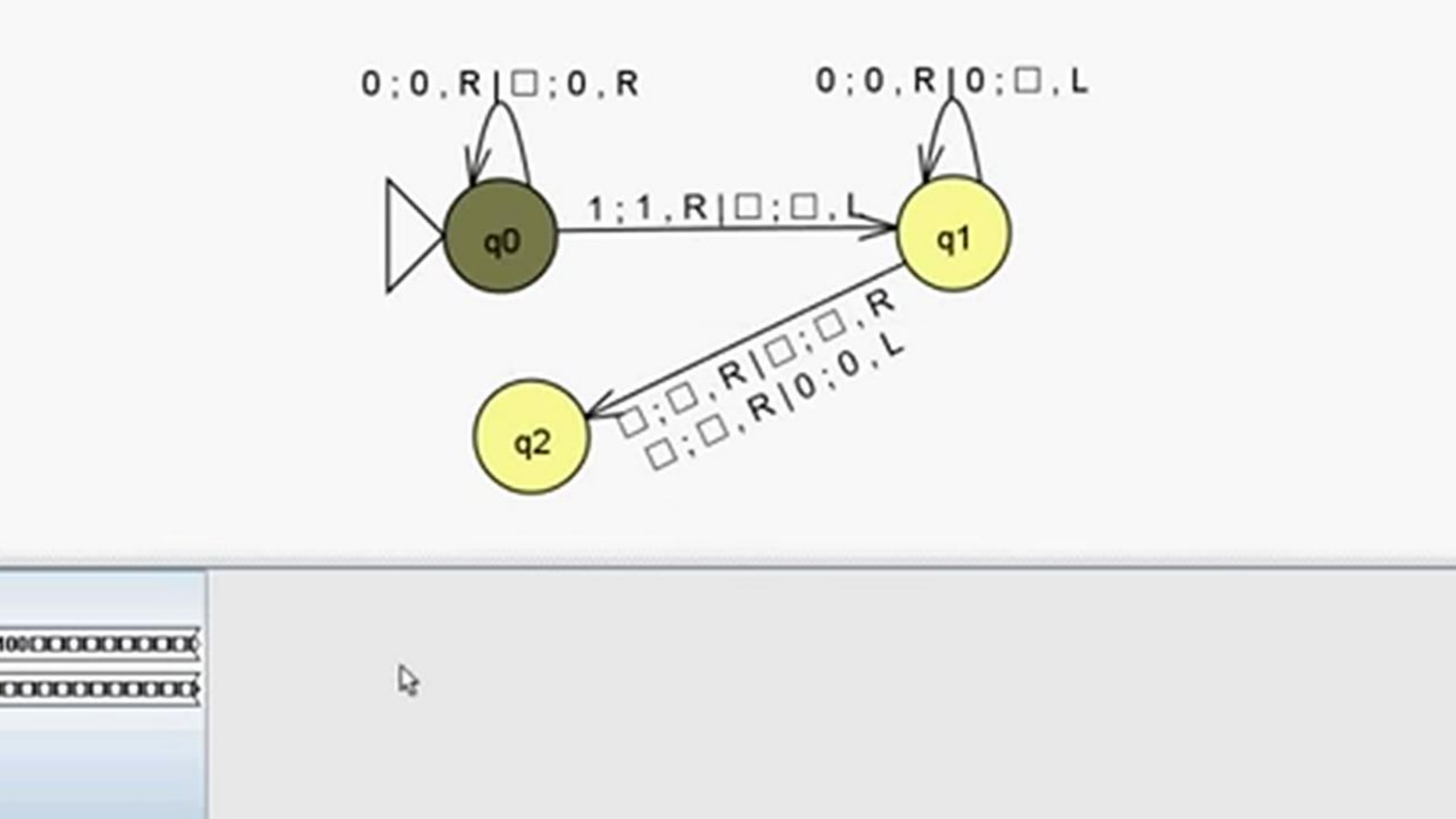


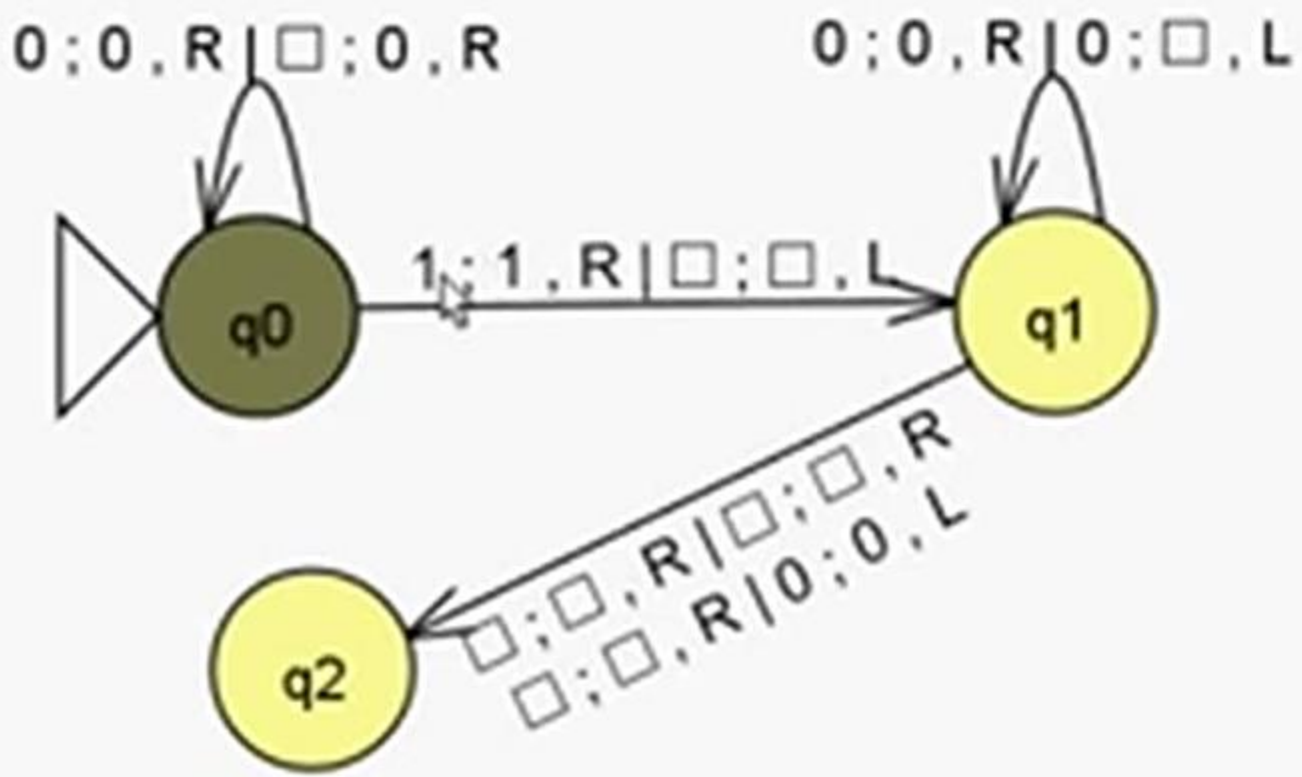
JFALP: calculam-nMulti.jff


0;0,R|0;□,L
q1
Input
Input 1
Input 2
Aceptar
Cancelar

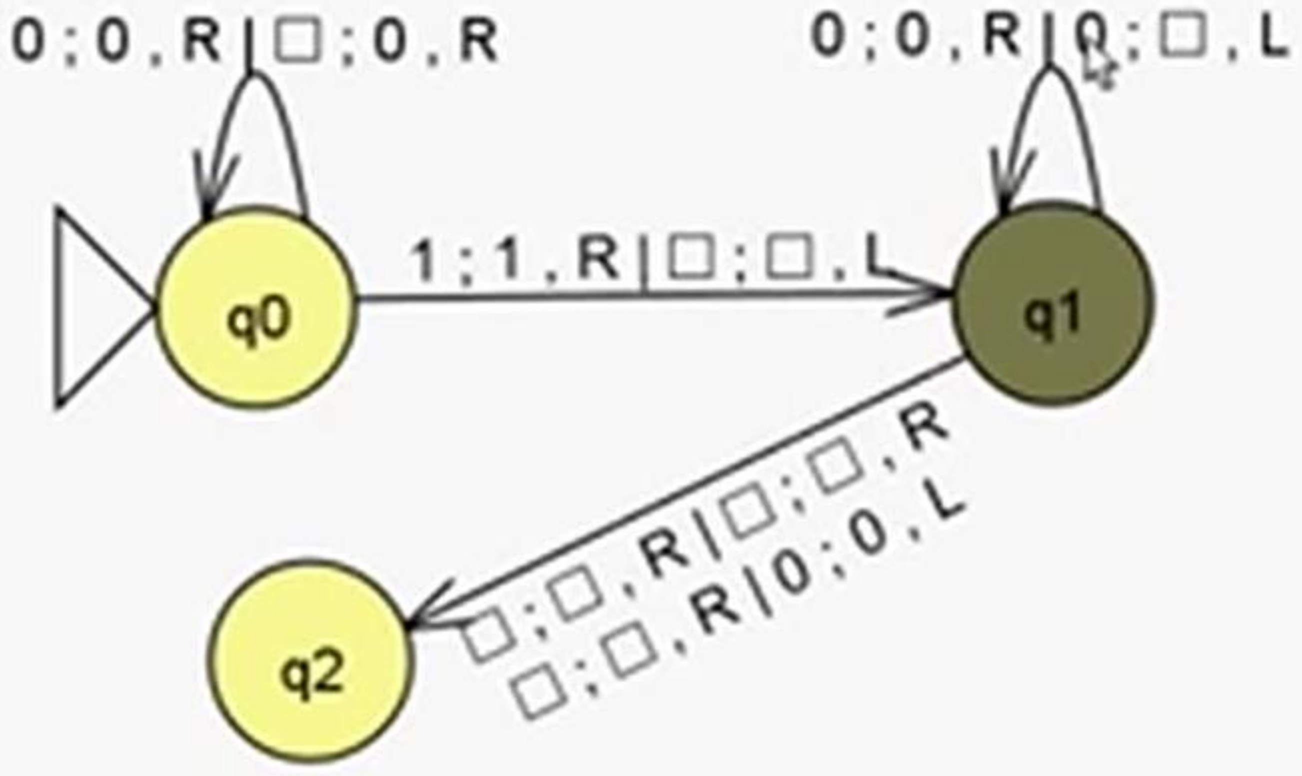

0;0,R|0;□,L
q1
□;□,R
|0;0,L


Input
Input 1
0000100
Input 2
Aceptar
Cancelar

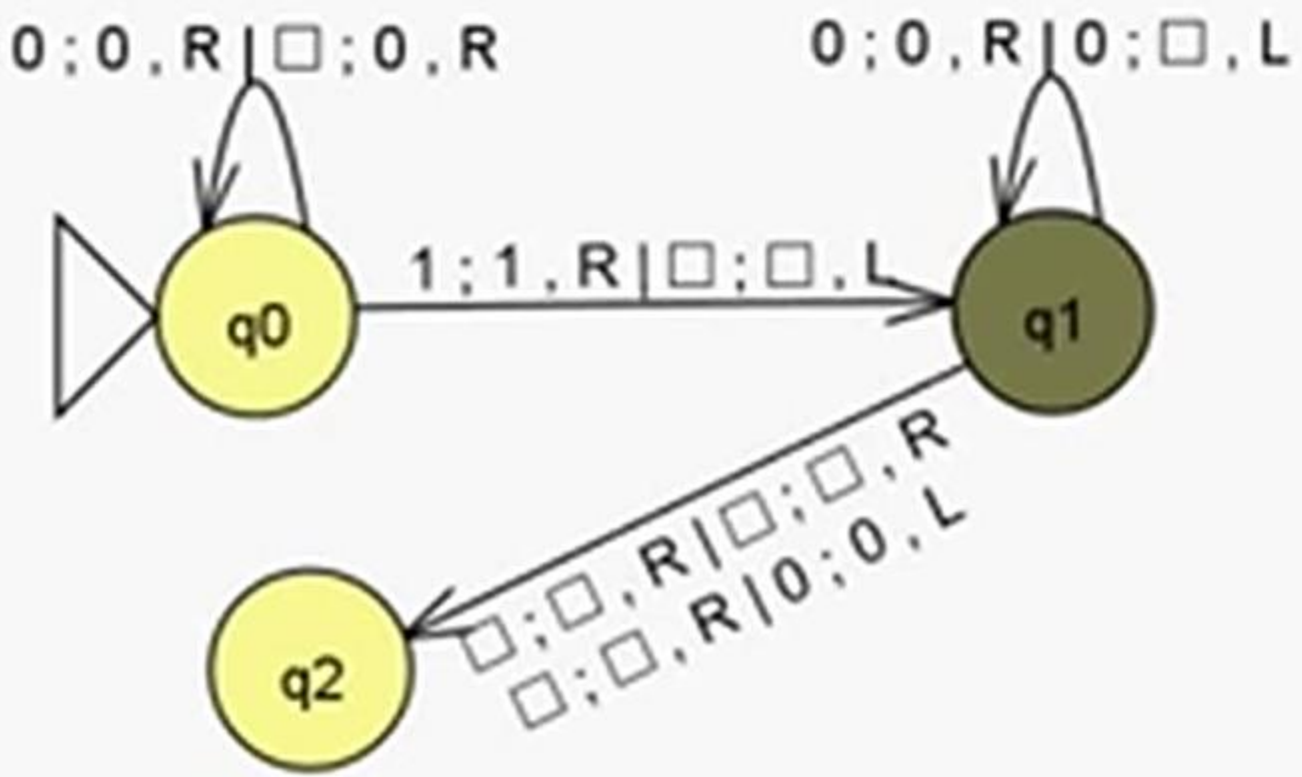

0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
1 ; 1 , R | □ ; □ , L
q0
q1
q2
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L

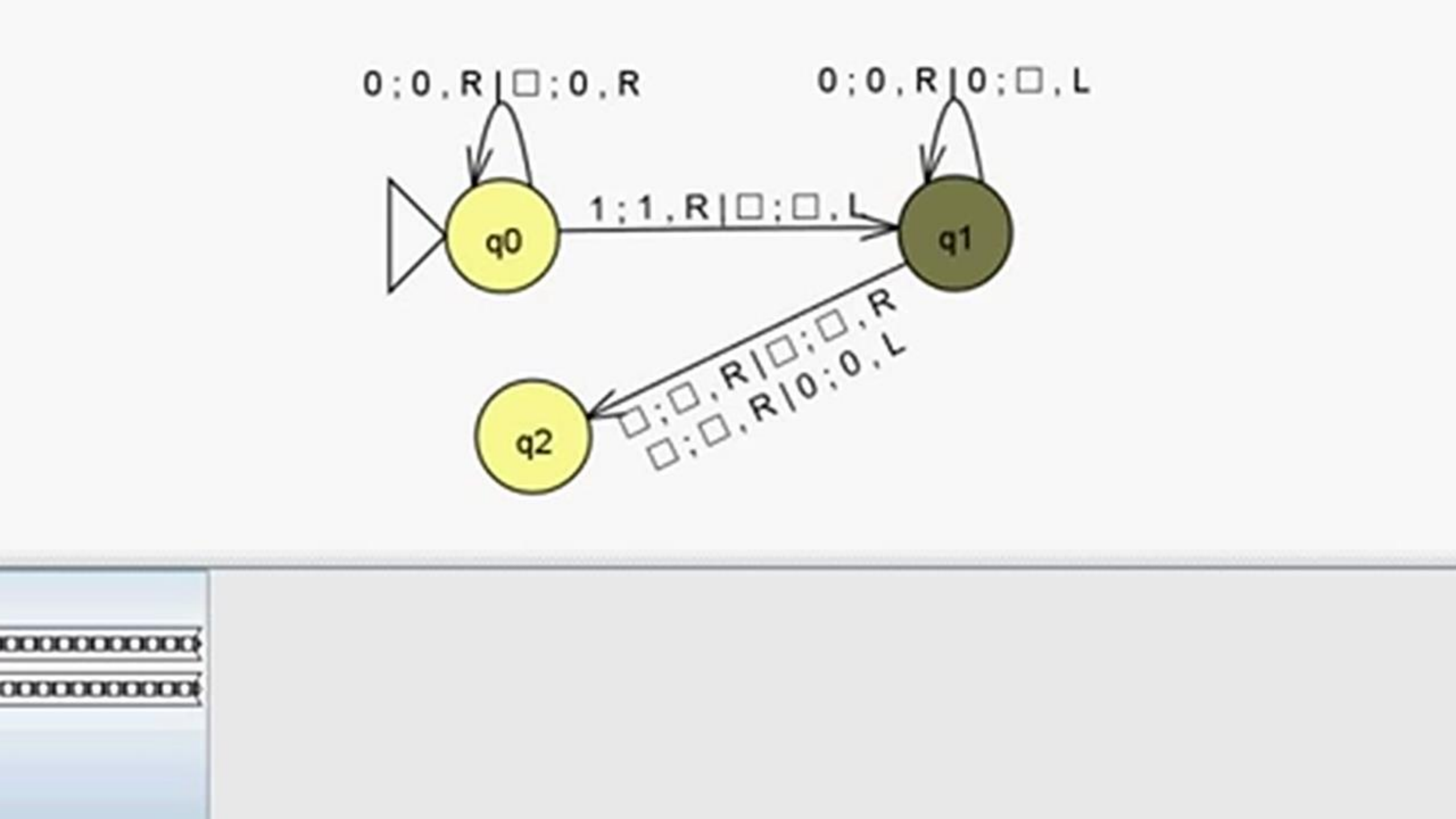


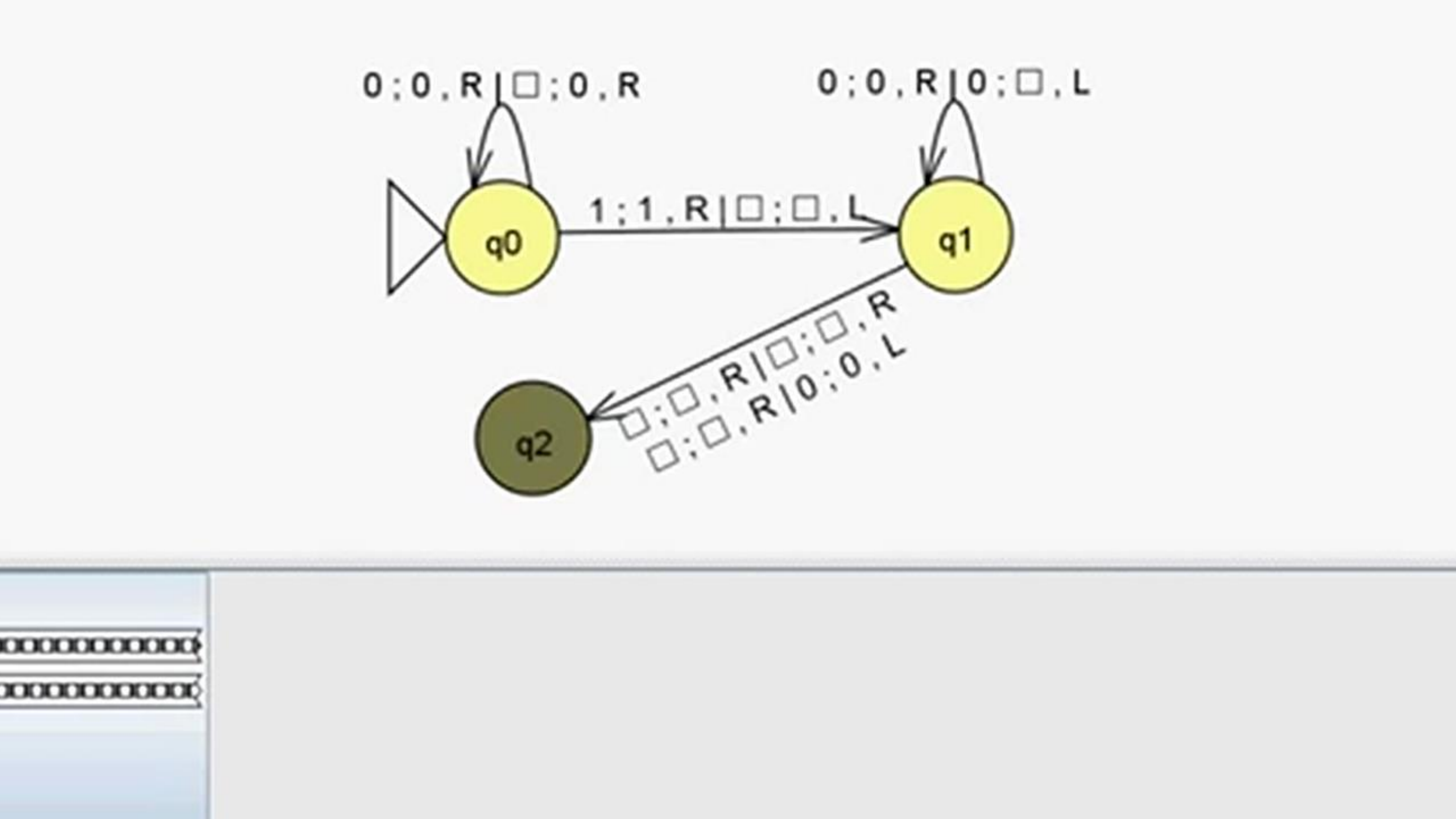
0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
1 ; 1 , R | □ ; □ , L
q0
q1
q2
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L



0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
1 ; 1 , R | □ ; □ , L
q0
q1
q2
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L
□ ; □ , R


0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
q0
1 ; 1 , R | □ ; □ , L
q1
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L
q2


0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
1 ; 1 , R | □ ; □ , L
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L
q0
q1
q2


0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
1 ; 1 , R | □ ; □ , L
q0
q1
q2
□ ; □ , R | □ ; □ , R
□ ; □ , R | 0 ; 0 , L

MT calculadora

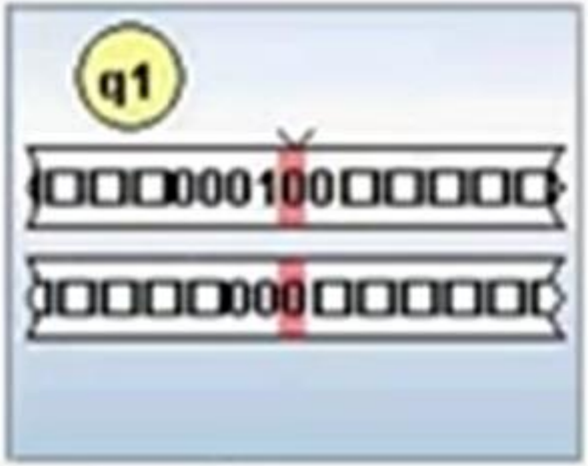
- Por ejemplo: sustracción propia

$$\mathbf{m - n} \begin{cases} m\text{-}n & \text{si } m \geq n \\ 0 & \text{si } m < n \end{cases}$$

p.e: operandos 3 y 2 para la operación 3 -

Entrada = 000100

Salida = 0

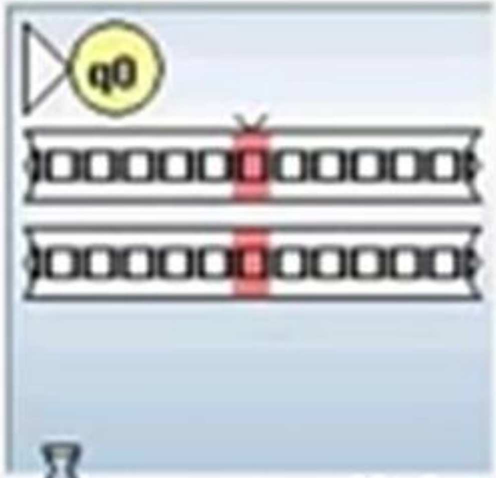


JFLAP: calculam-nMulti.jff

MT generadora de lenguajes

- MT no necesita cadena de entrada
- MT comienza a operar con la cinta en blanco en el estado inicial q_0.
- Cada vez que MT retorna al estado inicial q_0, hay una cadena w en L escrita sobre la cinta (salida de solo escritura).
- Todas las cadenas de L son, eventualmente, generadas por M.

MT generadora de lenguajes



Por ejemplo: $L = \{ 0^{2n} \mid n > 0 \}$

MT de 2 cintas:

δ(q0,B,B) = (q1,(0,I),(#,D)) // inicio de la cadena salida

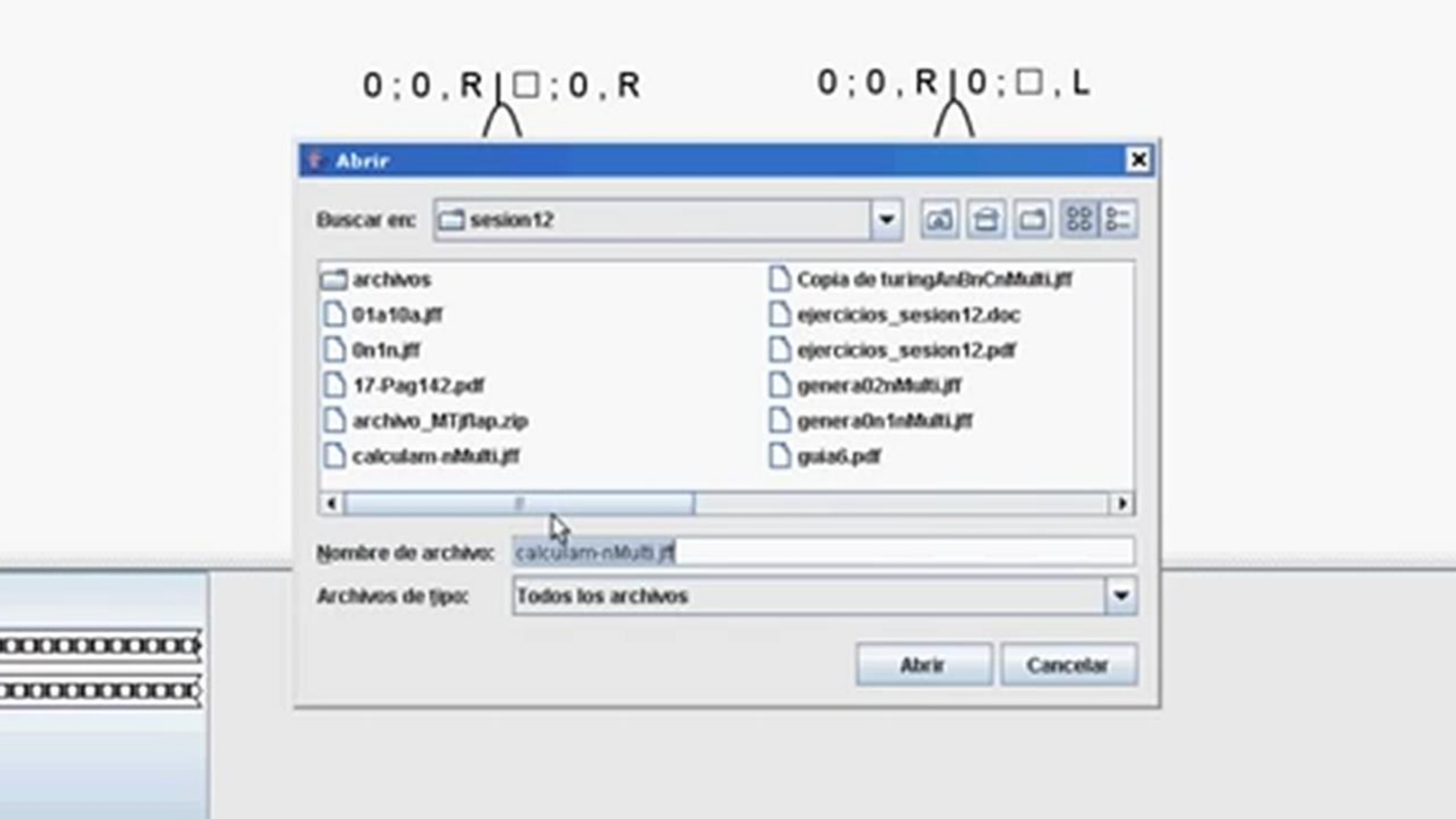
δ(q1,0,B) = (q1,(0,I),(B,E)) // busca inicio del contador n

δ(q1,B,B) = (q2,(B,D),(B,E)) // busca inicio del contador n

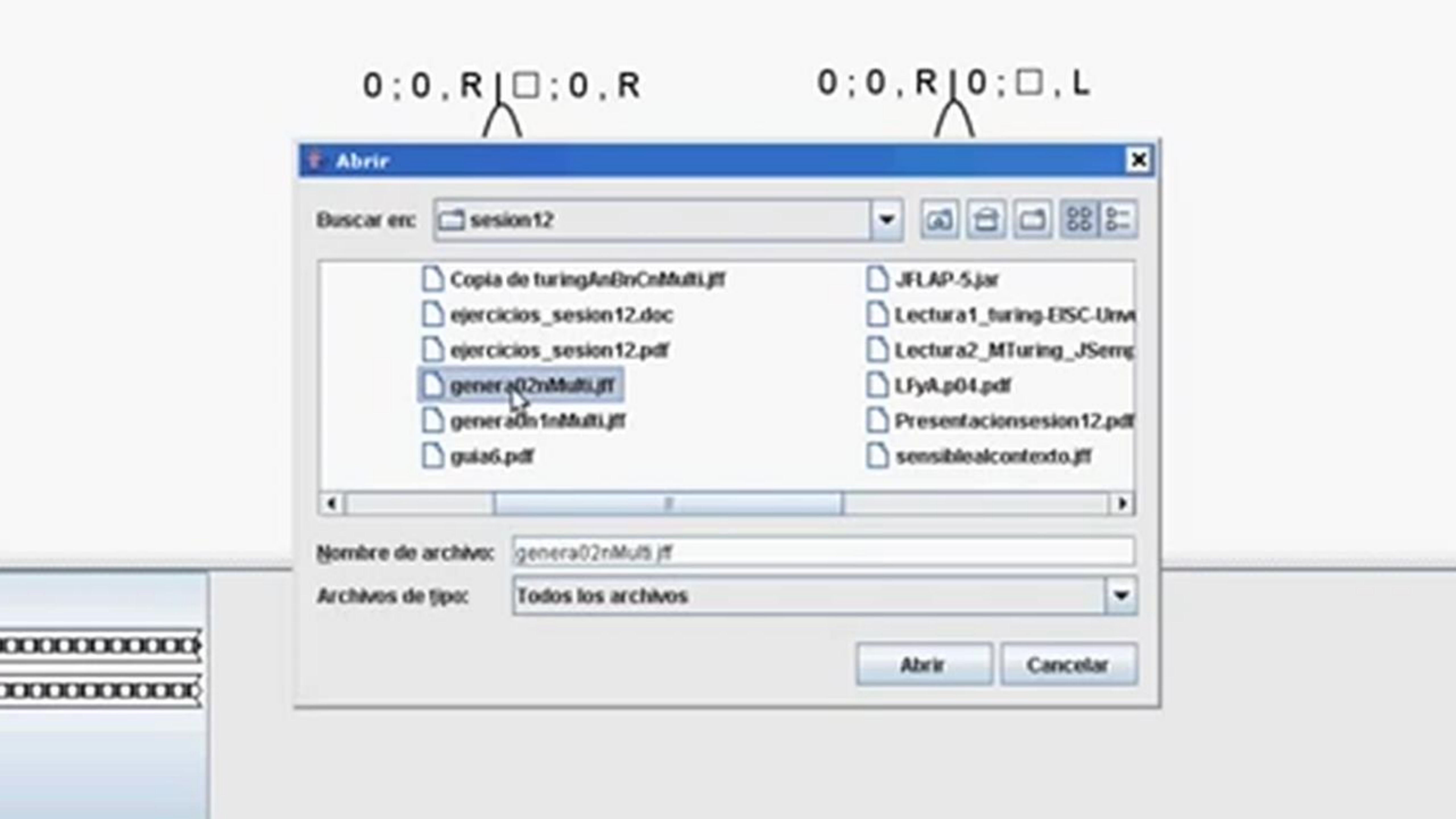
δ(q2,0,B) = (q3,(0,E),(0,D)) // coloca cinda 2 el 0 impar

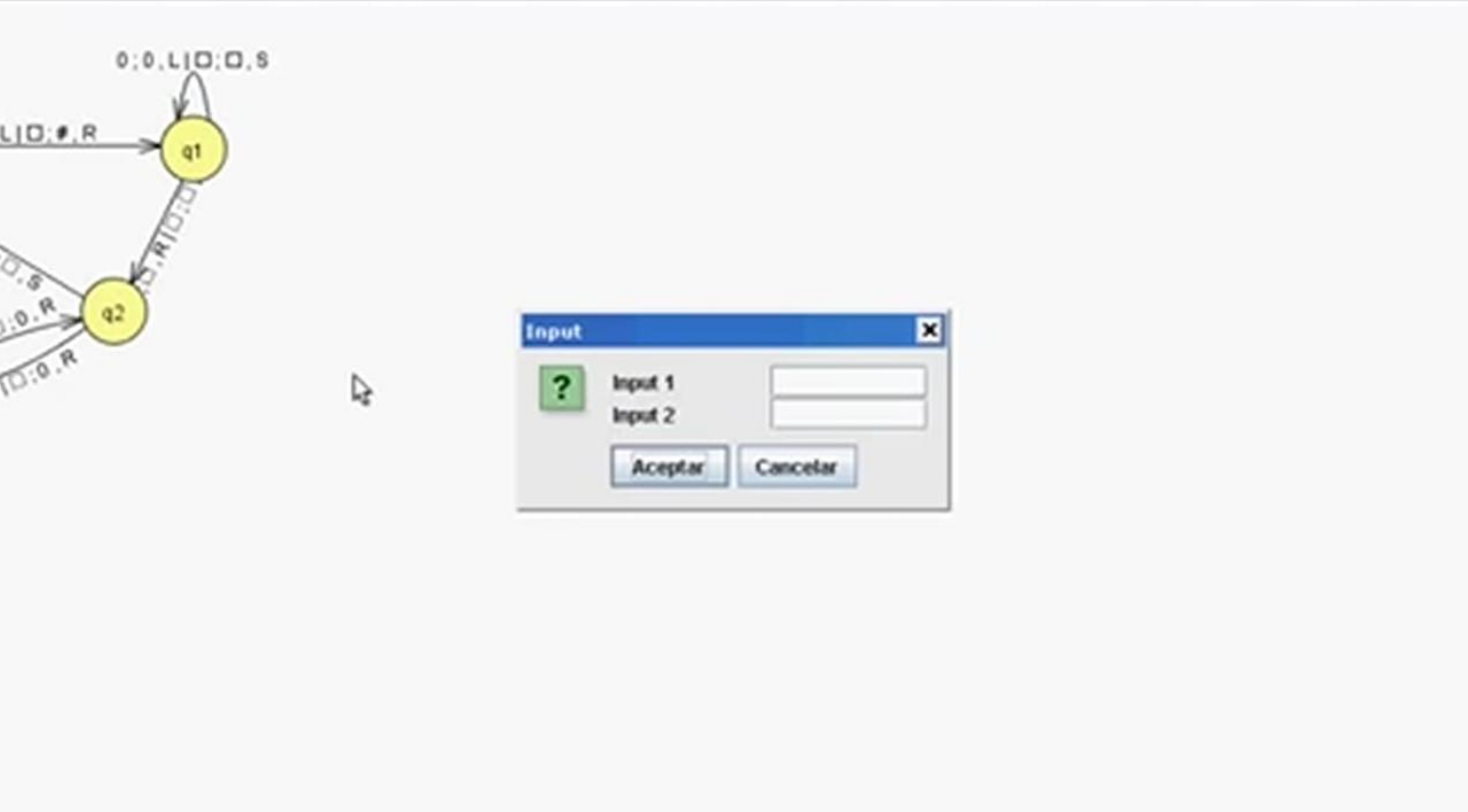
δ(q3,0,B) = (q2,(0,D),(0,D)) // coloca cinda 2 el 0 par

δ(q2,B,B) = (q0,(B,E),(B,E)) // fin del contador n, próxima cadena

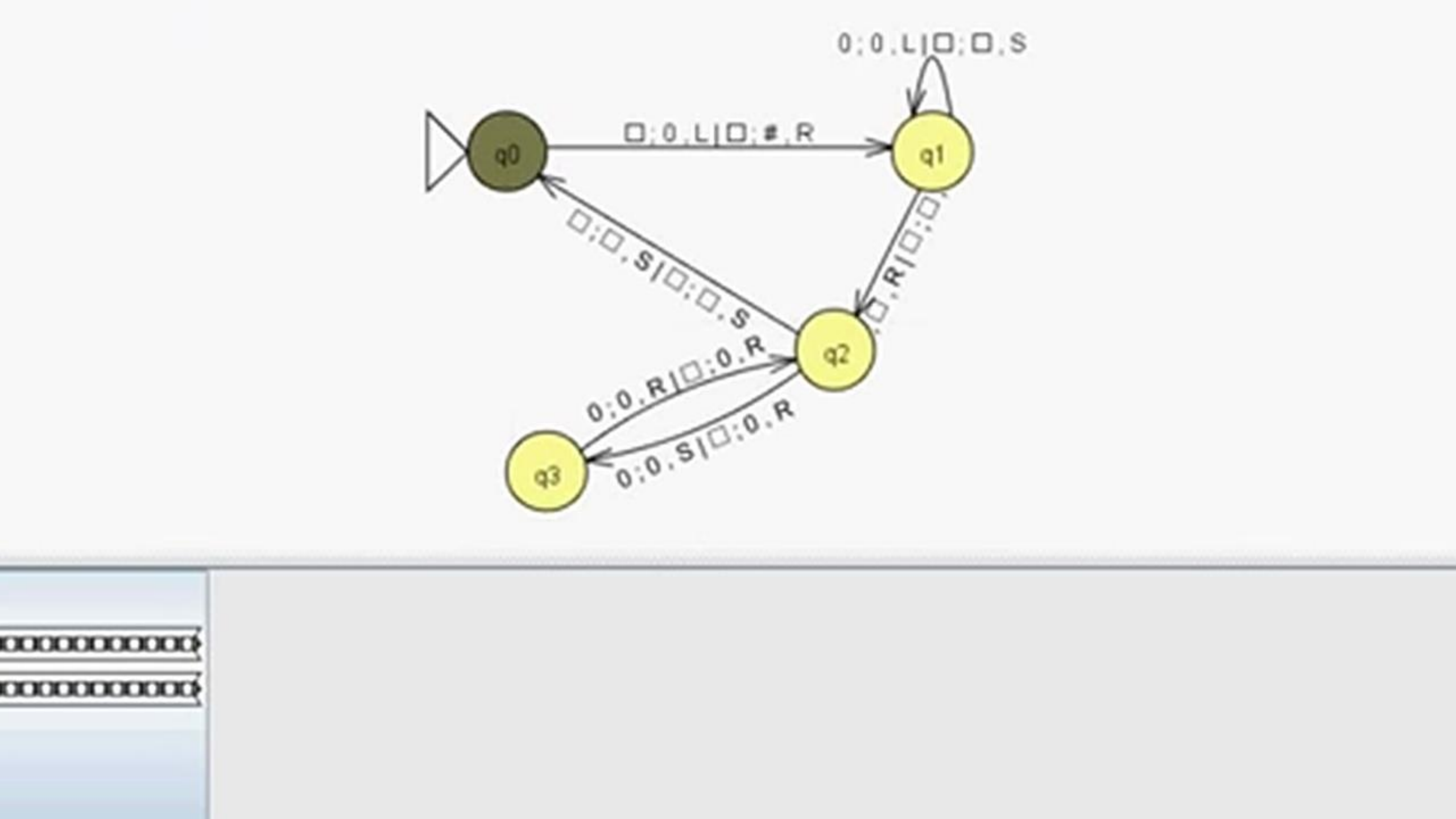


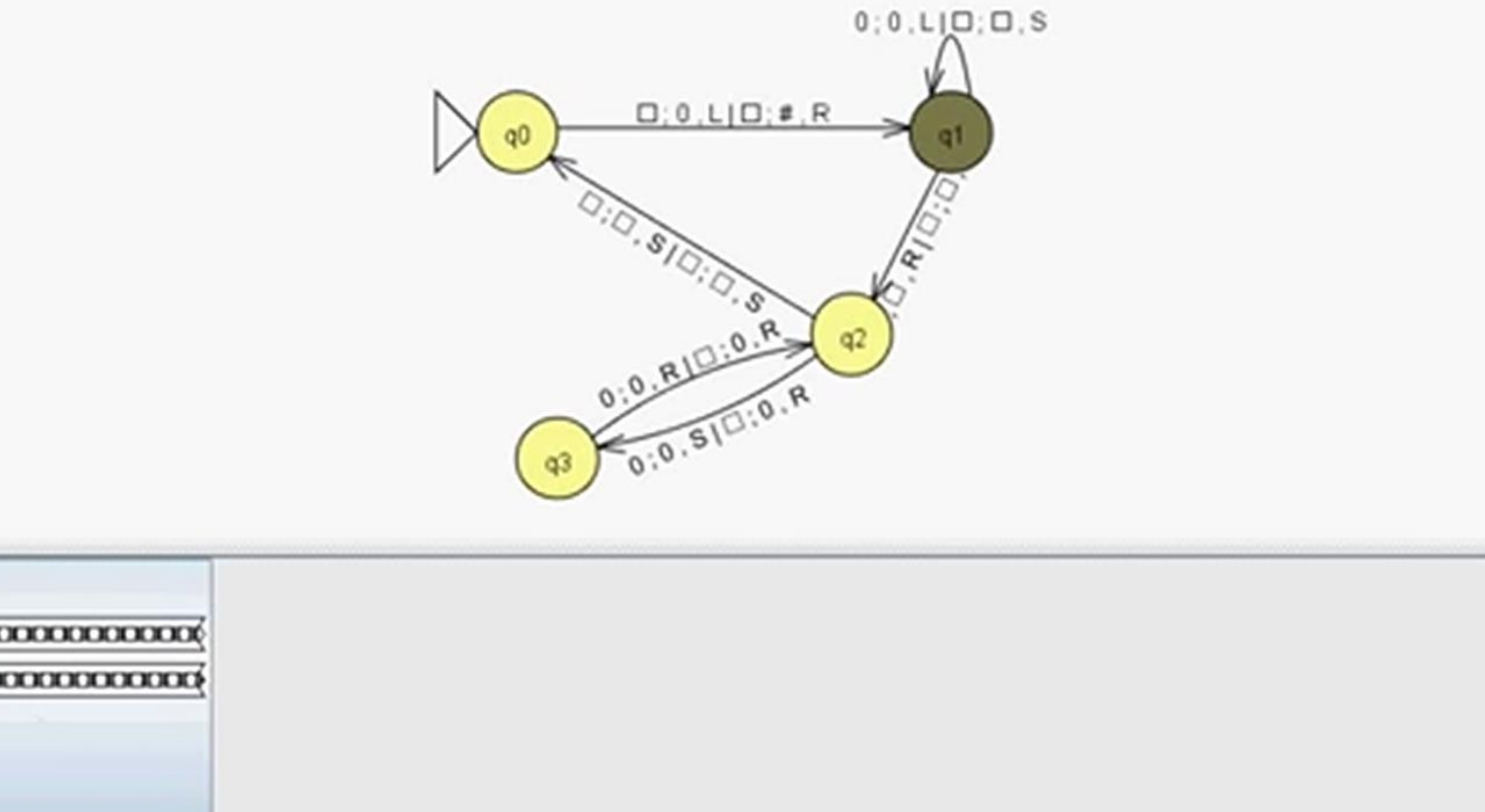
0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
Abrir
Buscar en:
sesion12
archivos
01a10a.jff
0n1n.jff
17-Pag142.pdf
archivo_MTjflap.zip
calculam-nMulti.jff
Copia de turingAnBnCnMulti.jff
ejercicios_sesion12.doc
ejercicios_sesion12.pdf
genera02nMulti.jff
genera0n1nMulti.jff
guia6.pdf
Nombre de archivo:
calculam-nMulti.jff
Archivos de tipo:
Todos los archivos
Abrir
Cancelar

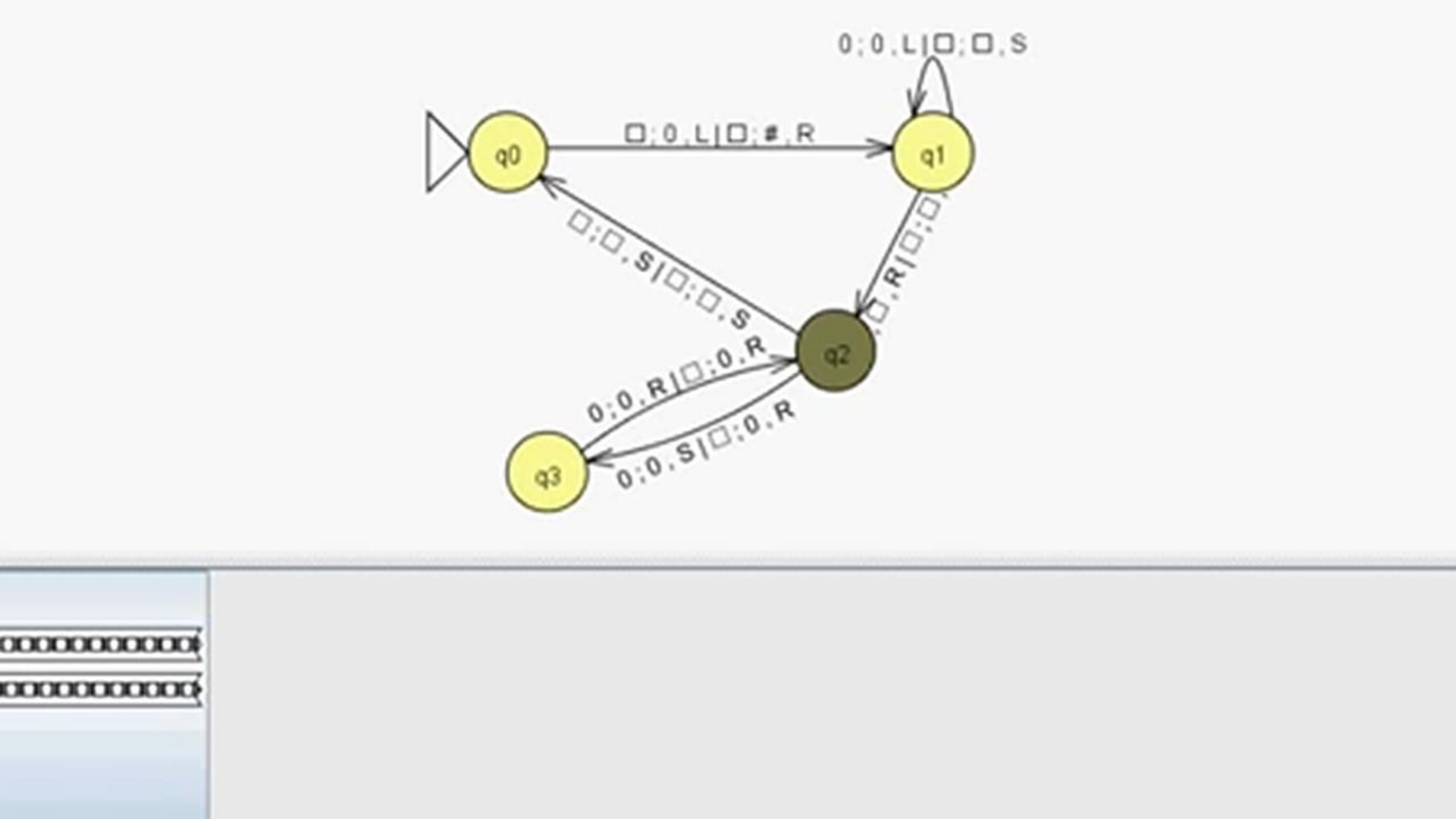

0 ; 0 , R | □ ; 0 , R
0 ; 0 , R | 0 ; □ , L
Abrir
Buscar en:
sesion12
Copia de turingAnBnCnMulti.jff
ejercicios_sesion12.doc
ejercicios_sesion12.pdf
genera02nMulti.jff
genera0n1nMulti.jff
guia6.pdf
JFLAP-5.jar
LFyA.p04.pdf
Presentacionsesion12.pdf
sensiblealcontexto.jff
Nombre de archivo:
genera02nMulti.jff
Archivos de tipo:
Todos los archivos
Abrir
Cancelar


0;0,L|□;□,S
q1
q2
Input
?
Input 1
Input 2
Aceptar
Cancelar

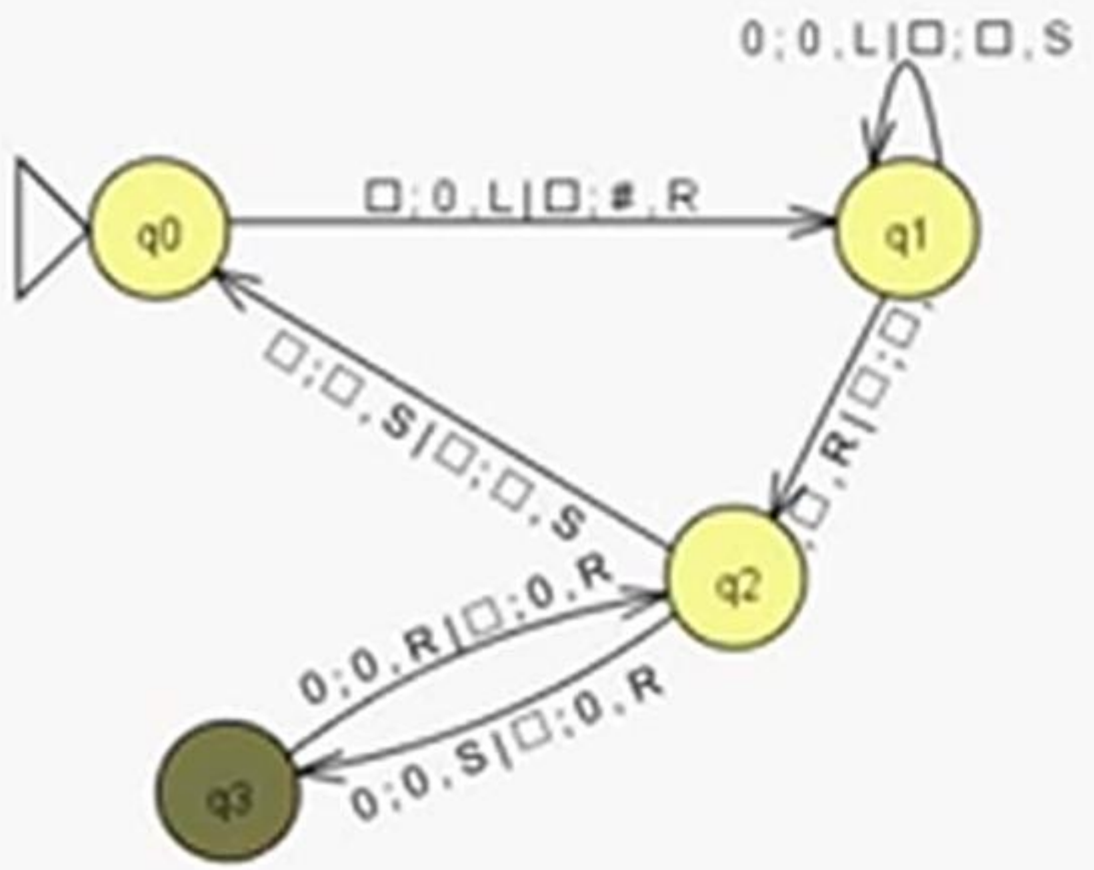

0;0,L|□;□,S
L|□;#,R
q1
q2
Input
Input 1
Input 2
Cancelar

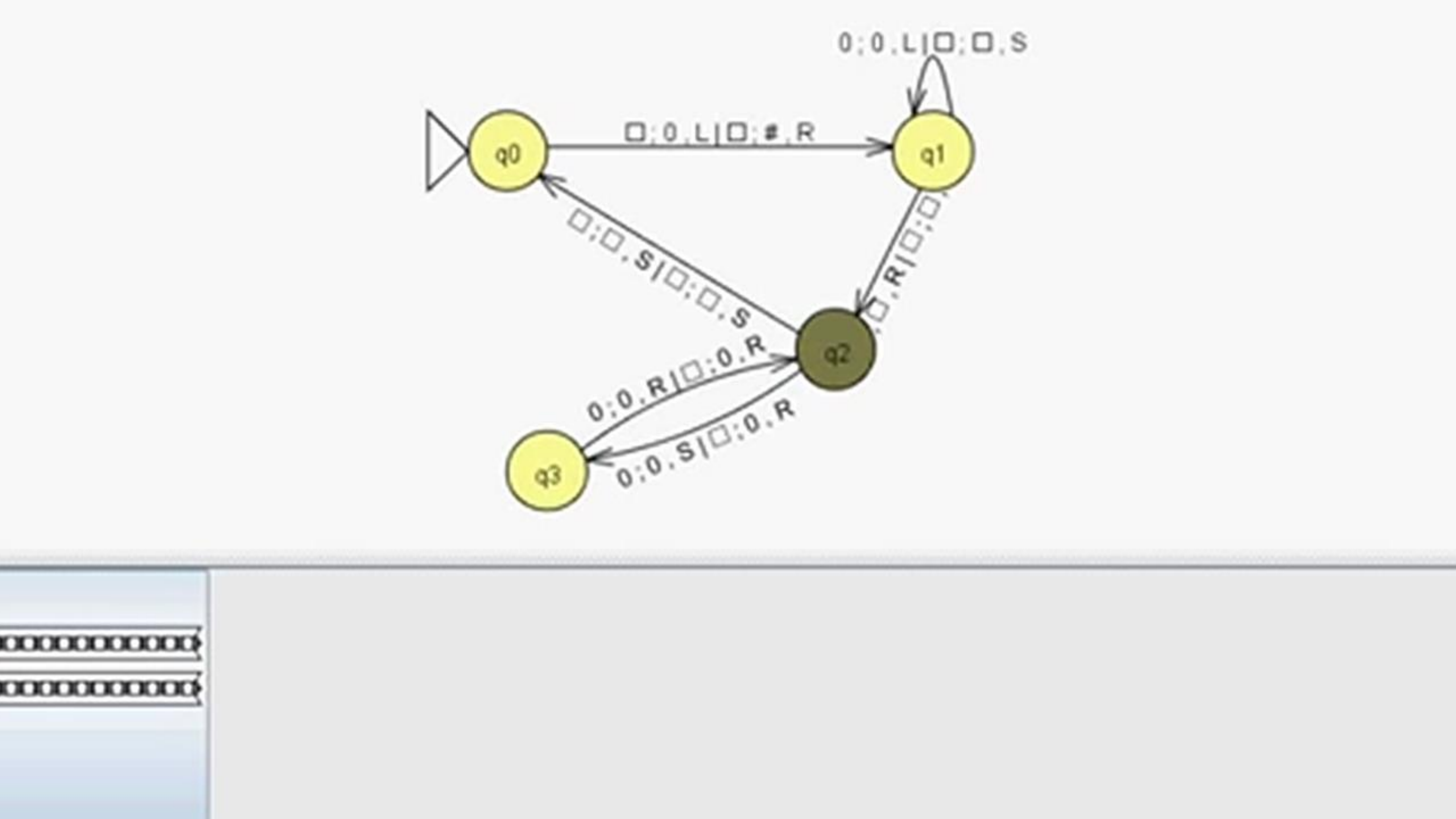

0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3

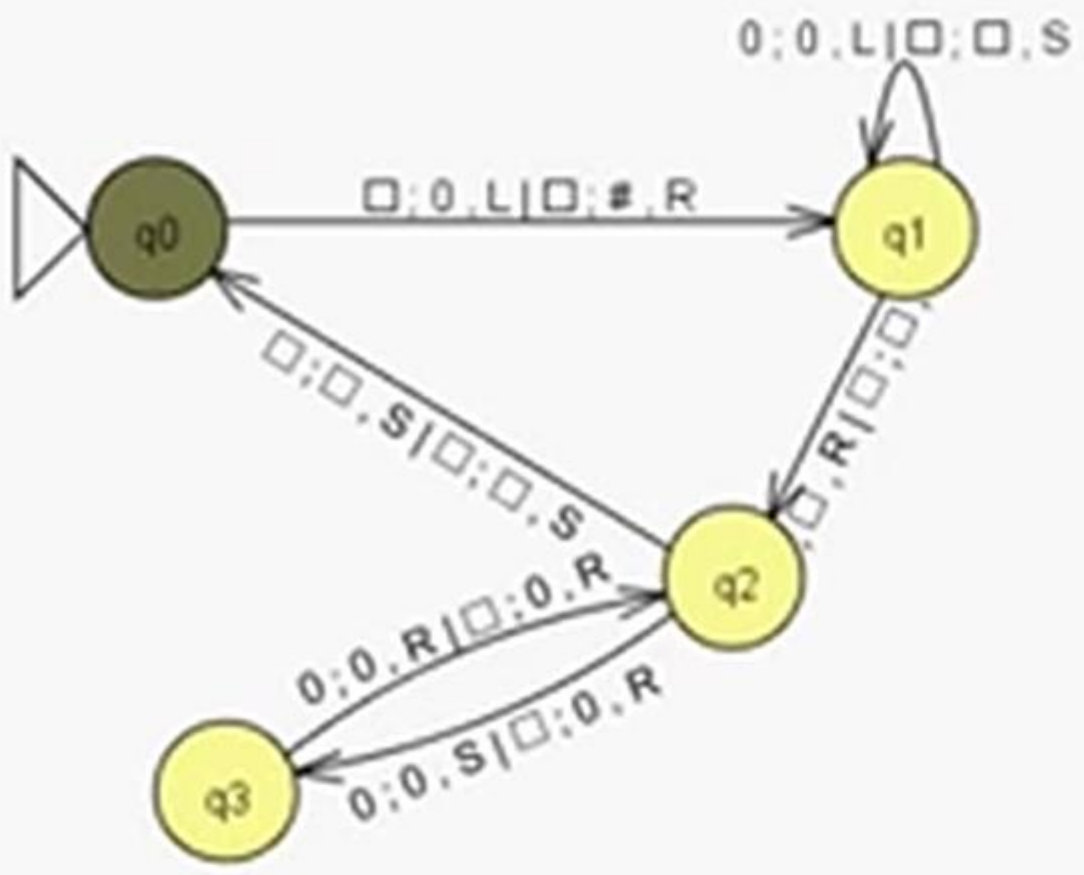

0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□,R|□;□
□;□,S|□;□,S
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3



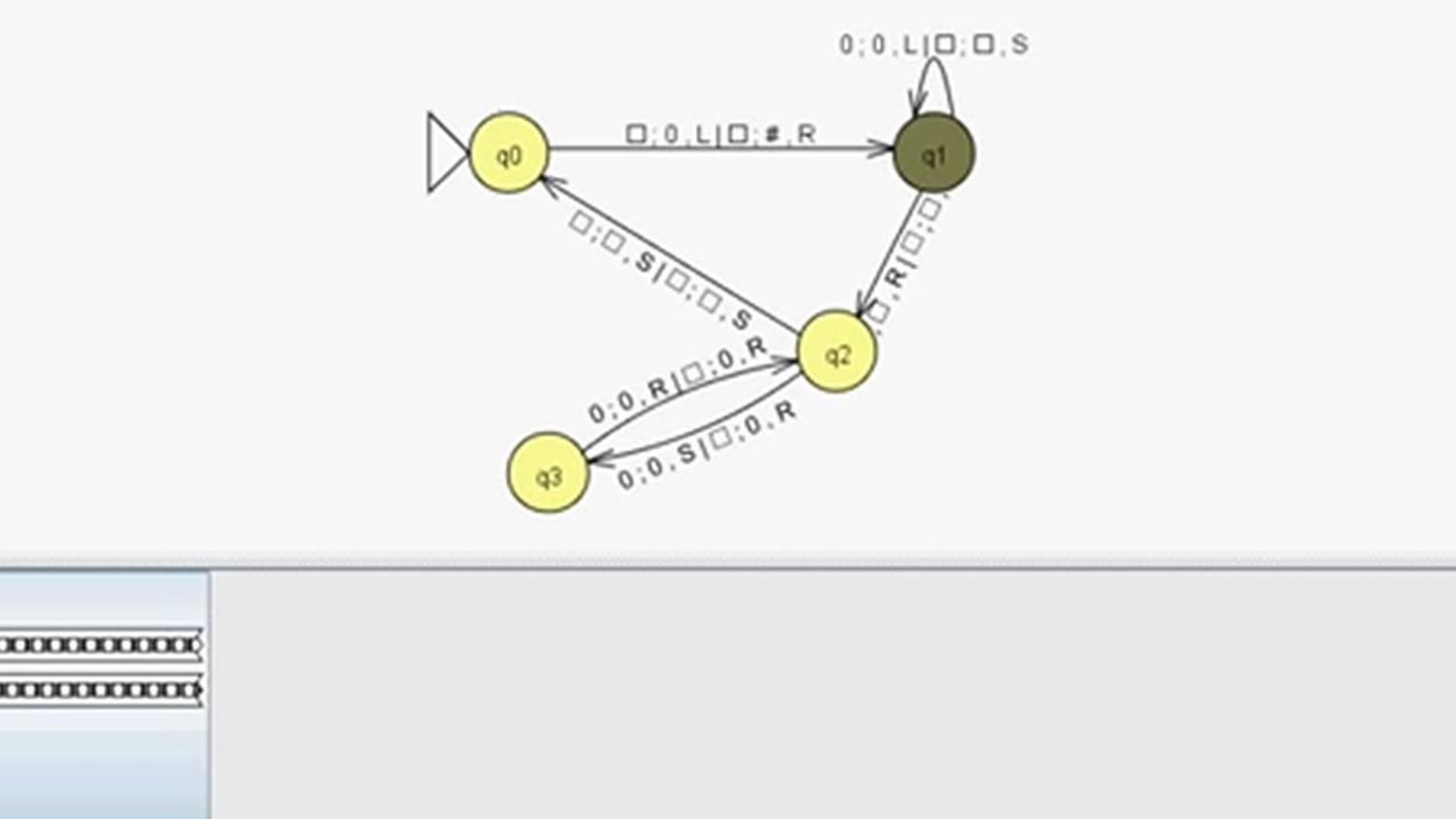
0;0,L|☐;☐,S
q0
☐;0,L|☐;#,R
q1
☐;☐,S|☐;☐,S
☐,R|☐;☐
q2
0;0,R|☐;0,R
0;0,S|☐;0,R
q3


0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
□;□,R|□;□,R
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3

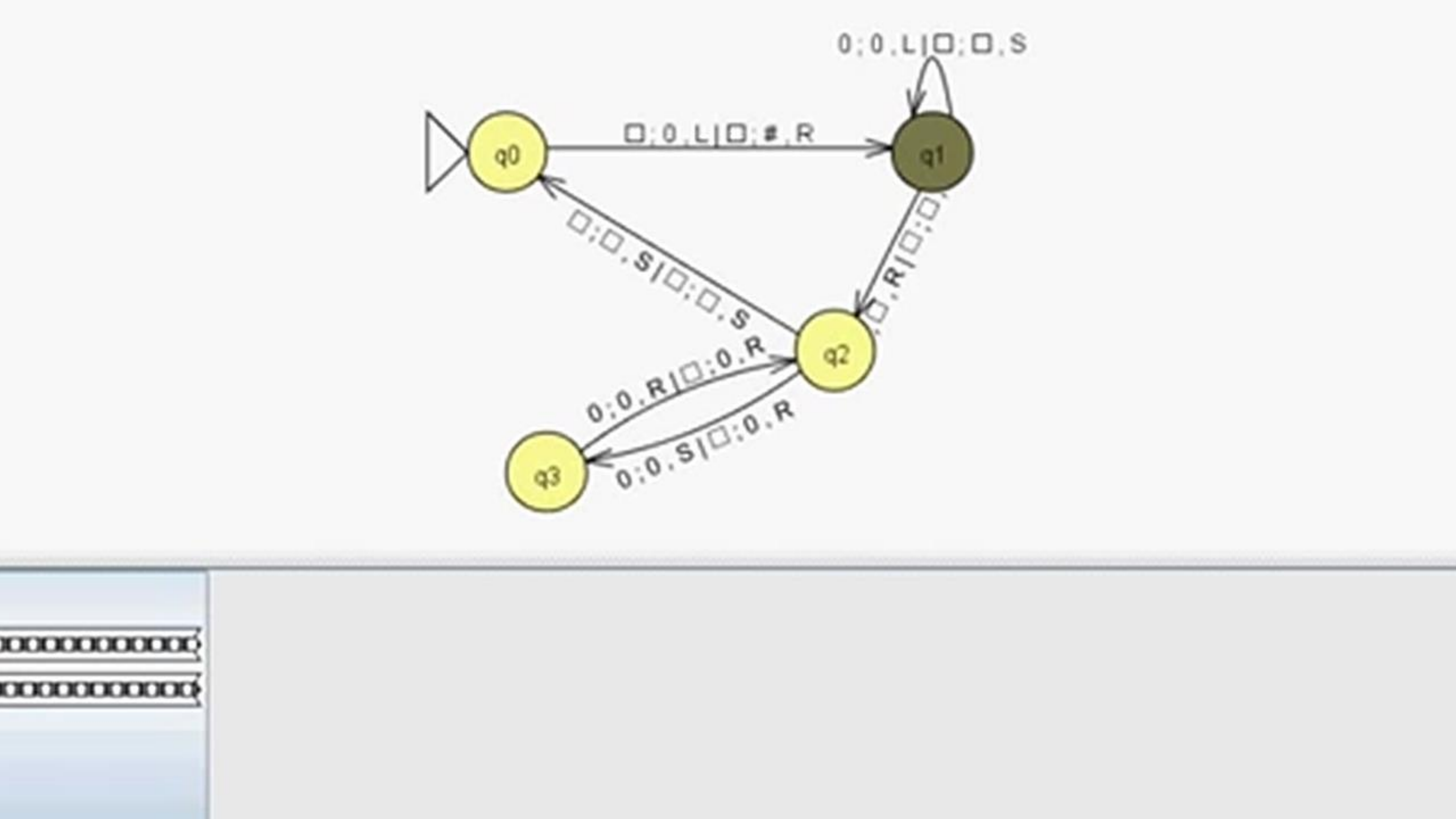

0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
q2
q3
□;□,S|□;□,S
0;0,R|□;0,R
0;0,S|□;0,R

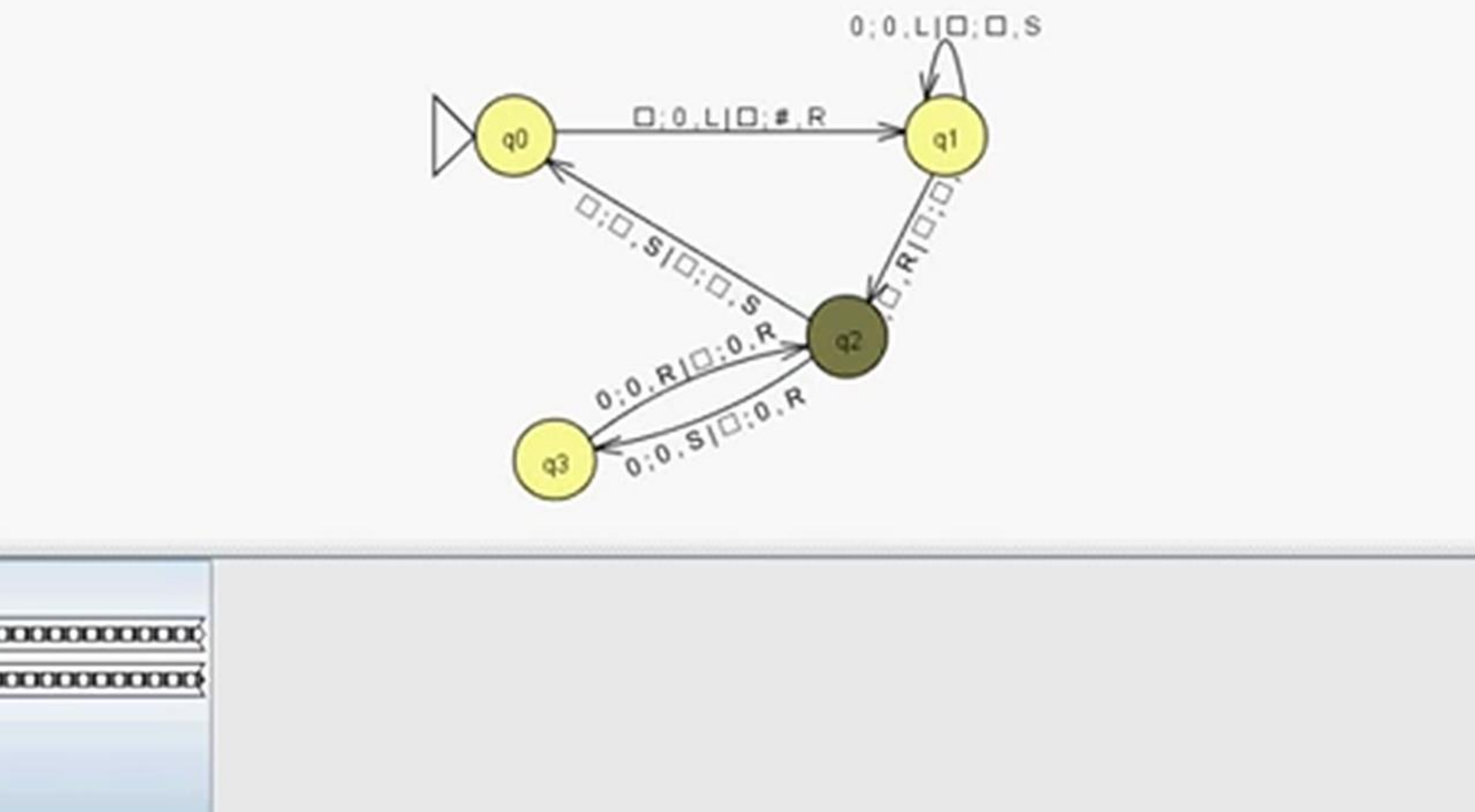


0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
□;□,R|□;□
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3



0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3


0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3


0;0,L|□;□,S
□;0,L|□;#,R
q0
q1
□;□,S|□;□,S
q2
0;0,R|□;0,R
0;0,S|□;0,R
q3

MT No determinista

$M = (\mathbf{Q}, \Sigma, \Gamma, \delta, \mathbf{q_0}, \mathbf{B}, \mathbf{F})$

$\delta: Q \times \Gamma \rightarrow (Q \times \Gamma \times \{D, I\})^*$

$\delta(q, X_i) = \{ (p, Y_i, D), (r, Z_i, D), \ldots, (q, X_i, I) \}$

- *El indeterminismo* de MT se refiere a las opciones que la máquina pueda tener en cualquier configuración por la función δ
- M acepta *w* si existe un camino desde q_0 hasta un

Equivalencias de MT

<u>Teorema</u>: *para cualquier máquina de Turing M_1 con k cintas existe otra equivalente M_2 con una sola cinta*

<u>Teorema</u>: *para cualquier máquina de Turing M_1 no determinista existe otra equivalente M_2 determinista*

<u>Teorema</u>: *para cualquier máquina de Turing M_1 con cinta infinita en ambos sentidos existe otra equivalente M_2 con cinta limitada por la izquierda*

enumerables

Lenguaje recursivamente enumerable: si existe una máquina de Turing M tal que L(M) = L. Define la clase *L*r.e.

Lenguaje recursivo: si existe una máquina de Turing M tal que L(M)=L y M se detiene ante cualquier entrada. Define la clase *L*rec

***L*rec es subconjunto de *L*r.e**

Una función es *computable* si puede ser calculada por una máquina de Turing.

Propiedades de clausura

Propiedad	LR	LLC	LRE
∪ (unión)	S	S	S
∩ (intersección)	S	N	S
complemento	S	N	N
concatenación	S	S	S
∩ LR	S	S	S

Propiedades de clausura

Propiedad	LR	LLC	LRE
Kleene-clausura	S	S	S
Reflejo	S	S	S
Morfismo	S	S	S
Morfismo^{-1}	S	S	S
Diferencia	S	N	S

Algoritmos de decisión

Problemas	LR	LLC	LRE
Equidad	D	N	N
Inclusión	D	N	N
Membresía	D	D	N
Vacuidad	D	D	N
Finitud	D	D	N

Jerarquía de Chomsky

Tipo	Lenguaje	Máquina	Gramática G = (V,T,P,S)
0	**Recursivamente enumerable**	**Máquina de Turing**	**Gramática sin restricciones** $\alpha \rightarrow \beta$ (α, β en $(V \cup T)^*$, α contiene una variables)
1	**Dependiente del Contexto**	**Autómata linealmente acotado**	***Gramática sensible al contexto*** $\alpha \rightarrow \beta$ (α, β en $(V \cup T)^*$, α contiene una variables, $\lvert\beta\rvert \geq \lvert\alpha\rvert$)
2	**Independiente del Contexto**	**Autómata de Pila**	***Gramática libre de contexto*** $A \rightarrow \alpha$ (A en V y α en $(V \cup T)^*$)
3	**Lenguaje**	**Autómata finito**	***Gramática Regular*** $A \rightarrow aB$

Gramáticas sensibles al contexto

- Una gramática $G = (V,T,P,S)$ es una gramática sensible de contexto si todas las producciones son de la forma

$$\alpha \rightarrow \beta$$

- donde α, β en $(V+T)^*$ y $|\alpha| \leq |\beta|$, α contiene al menos una variable V

Lenguajes sensibles al contexto

Gramáticas sensibles al contexto

Por ejemplo: $L = \{ a^n b^n c^n \mid n > 0 \}$

$S \rightarrow A_0BCS_1 \mid A_0BC$

$S_1 \rightarrow ABCS_1 \mid ABC$

$BA \rightarrow AB$

$CA \rightarrow AC$

$CB \rightarrow BC$

$A_0 \rightarrow a, \quad aA \rightarrow aa, \quad aB \rightarrow ab$

al contexto

Autómatas linealmente acotadas

Un *autómata de memoria limitada linealmente* (ALL) es una máquina de Turing no determinista monocinta que satisface las siguientes condiciones:

1. El alfabeto de entrada incluye los limitadores ¢ (izquierda) y $ (derecha)
2. El ALL no mueve el cabezal de la cinta fuera de los limitadores ni escribe sobre ellos

Teorema: *L* es un lenguaje sensible al contexto (de

Gramáticas sin restricciones

- Una gramática G = (V,T,P,S) es una gramática de tipo 0 (irrestricta) si todas las producciones son de la forma

$$\alpha \rightarrow \beta$$

- donde α en $(V+T)^{+}$, β en $(V+T)^{*}$

Teorema: *L* es un lenguaje recursivamente enumerable sii *L*=L(G) donde G es de tipo 0.

Gramáticas irrestricta

Por ejemplo: L = { ww | w en $(a+b)^*$ }

S → FM

F → FaA | FbB

Aa → aA, Ab → bA, Ba → aB, Bb → bB

AM → Ma, BM → Mb

F → ε, M → ε