Formal Languages and Automata Theory
Exercises Turing Machines
Unit 7

Authors:

Araceli Sanchis de Miguel
Agapito Ledezma Espino
Jose A. Iglesias Martínez
Beatriz García Jiménez
Juan Manuel Alonso Weber

* Several exercises are based on the ones proposed in the following books:

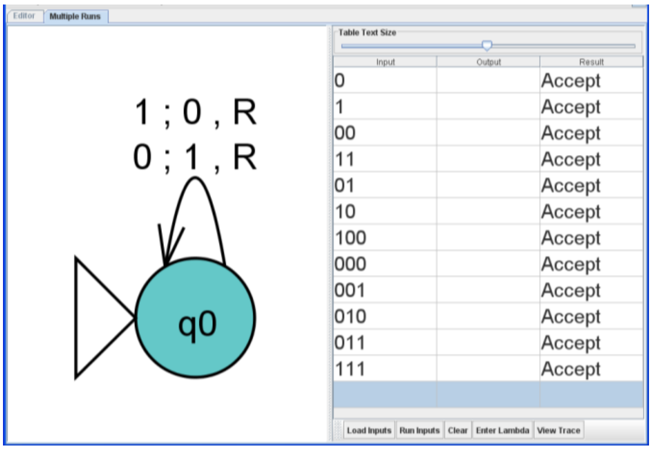
- Enrique Alfonseca Cubero, Manuel Alfonseca Cubero, Roberto Moriyón Salomón. *Teoría de autómatas y lenguajes formales.* McGraw-Hill (2007).
- Manuel Alfonseca, Justo Sancho, Miguel Martínez Orga. *Teoría de lenguajes, gramáticas y autómatas.* Publicaciones R.A.E.C. (1997).
- Pedro Isasi, Paloma Martínez y Daniel Borrajo. *Lenguajes, Gramáticas y Autómatas. Un enfoque práctico.* Addison-Wesley (1997).

Universidad Carlos III de Madrid
www.uc3m.es

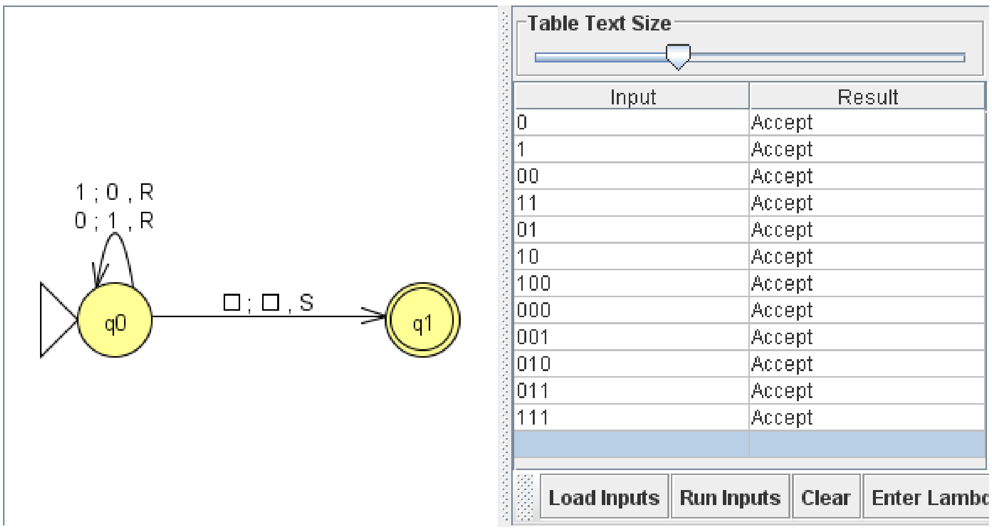
1. Design a Turing Machine to calculate the 1-complement of a binary number (i.e. replace 0's with 1's and 1's with 0's).

<u>Solution</u>:

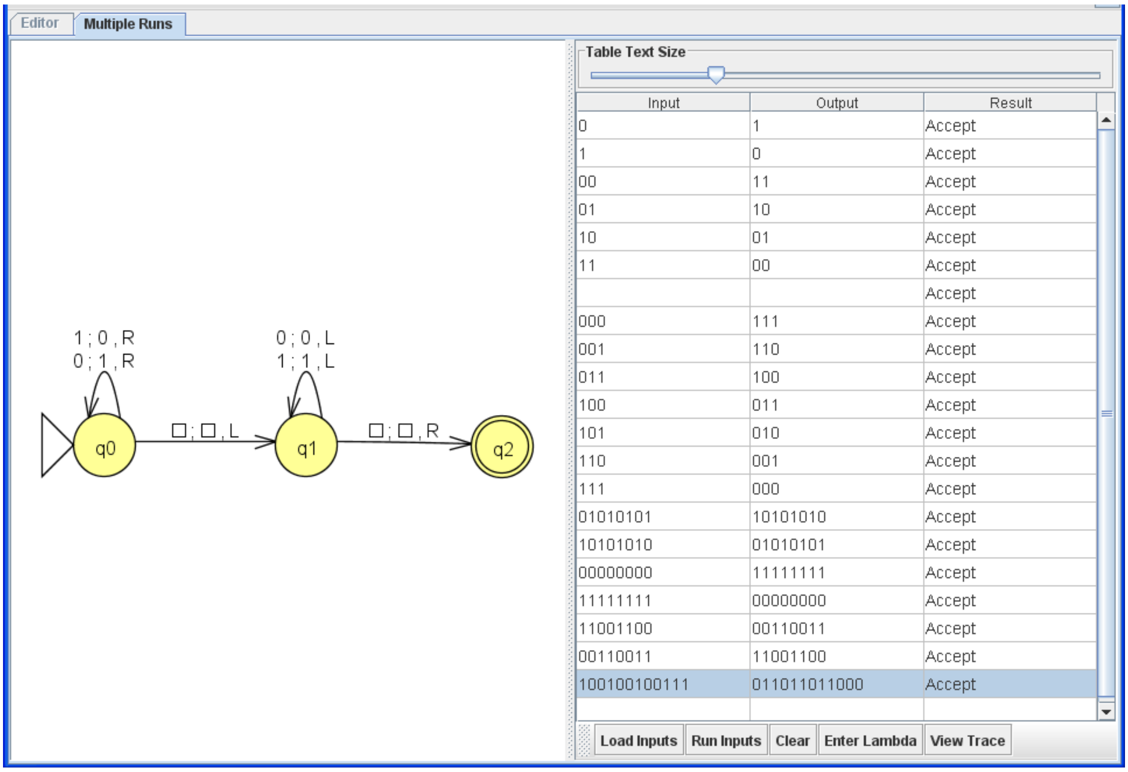
MT_1=({0,1},{0,1,□},□,{q0},q0,f,{Φ}), where f:



<u>Alternative solution</u>: $MT_{1.2}$=({0,1},{0,1,□},□,{q0,q1},q0,f,{q1}), where f:



Alternative solution: $MT_{1.3}$=({0,1},{0,1,□},□,{q0,q1,q2},q0,f,{q2}), where f:

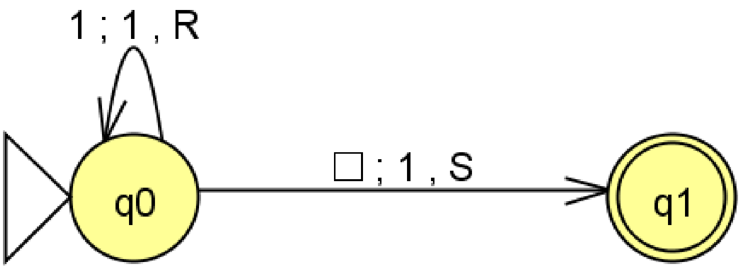


Input	Output	Result
0	1	Accept
1	0	Accept
00	11	Accept
01	10	Accept
10	01	Accept
11	00	Accept
		Accept
000	111	Accept
001	110	Accept
011	100	Accept
100	011	Accept
101	010	Accept
110	001	Accept
111	000	Accept
01010101	10101010	Accept
10101010	01010101	Accept
00000000	11111111	Accept
11111111	00000000	Accept
11001100	00110011	Accept
00110011	11001100	Accept
100100100111	011011011000	Accept

2. **Design a Turing Machine to obtain the successor of a number in unary codification. Consider that the unary representation of 0 is the empty string, 1 is represented by 1, 2 is represented by 11, etc.**

Solution:

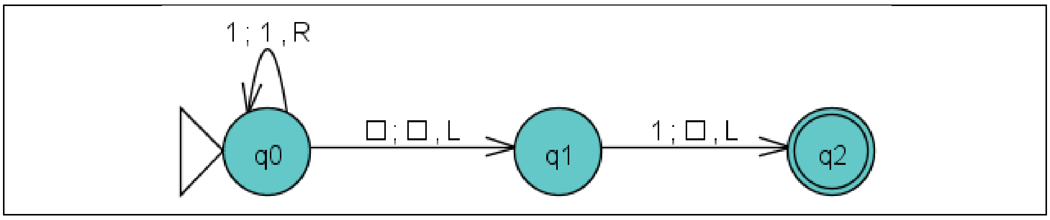
MT_2=({1},{1,□},□,{q0,q1},q0,f,{q1}), where f:



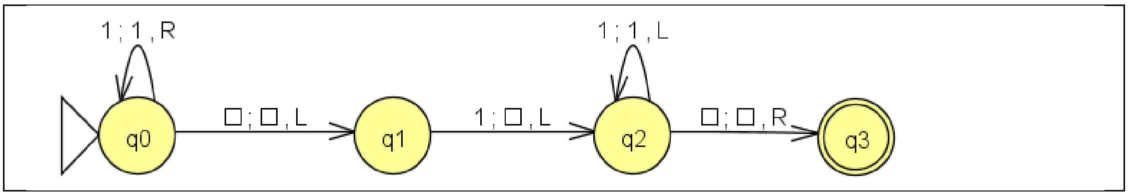
3. Design a Turing Machine to obtain the predecessor of a number in unary codification. Consider the same representation described in the previous exercise.

<u>Solution</u>:

$MT_{3.1}$=({1},{1,□},□,{q0,q1,q2},q0,f,{q2}), where f:



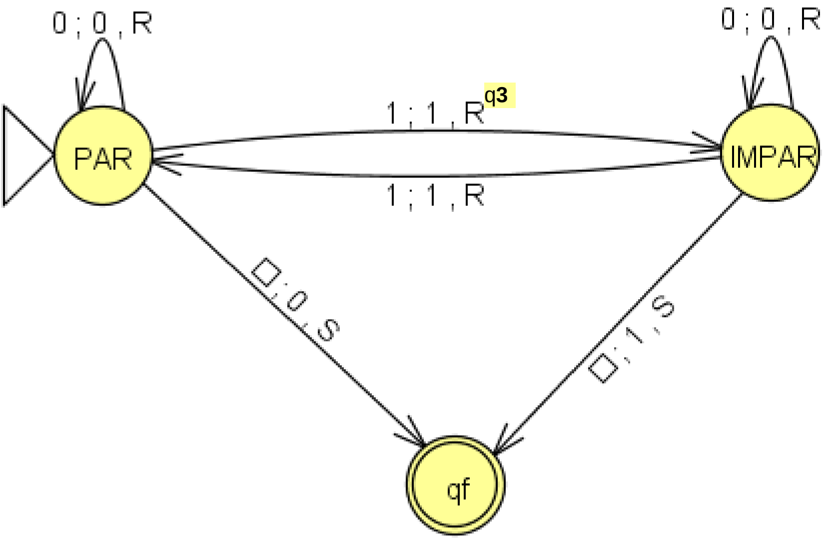
<u>Alternative solution</u>: $MT_{3.2}$=({1},{1,□},□,{q0,q1,q2,q3},q0,f,{q3}), where f:



4. Design a Turing Machine to calculate the parity of a binary number, i.e. add a 0 at the end if the number of 1's in the input string is even or a 1 is this number is odd.

<u>Solution</u>:

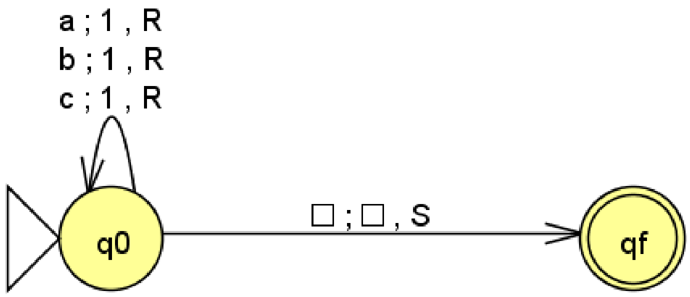
MT_4=({0,1},{0,1,□},□,{PAR, IMPAR, qf},PAR,f,{qf}), where f:



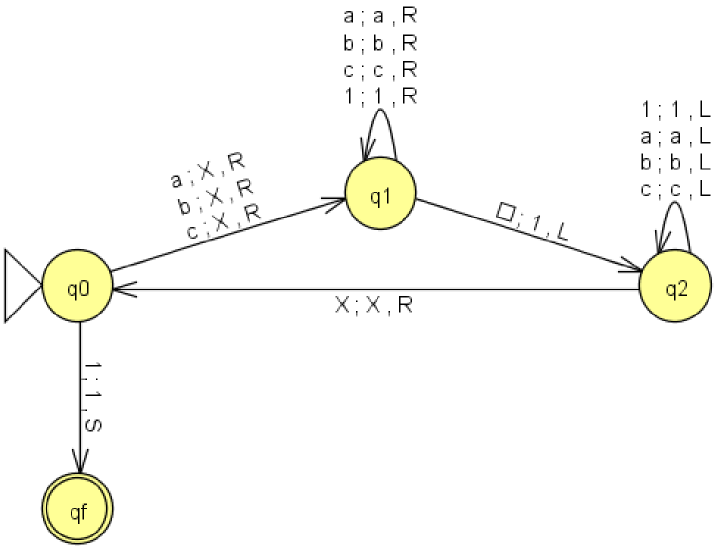
5. Design a Turing Machine to be a unary counter of characters in the language with alphabet Σ = {a,b,c}, i.e., the machine must generate as 1's as output as characters in the input word. Consider the same representation for 0 defined in the exercise 2.

<u>Solution</u>:

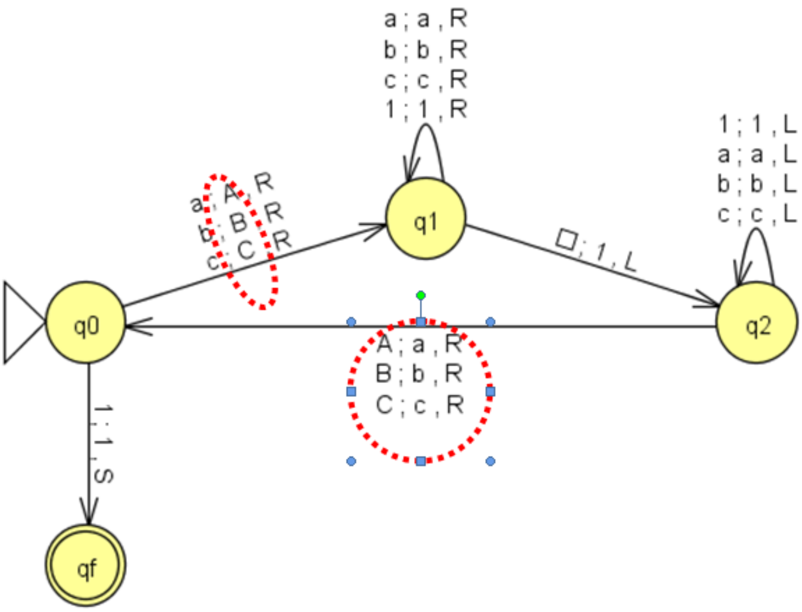
$MT_{5.a}$=({a,b,c},{1,□},□,{q0,qf},q0,f,{qf}), where f:



<u>Alternative solution</u>: $MT_{5.b}$=({a,b,c},{1,X,□},□,{q0,q1,q2,qf},q0,f,{qf}), where f:



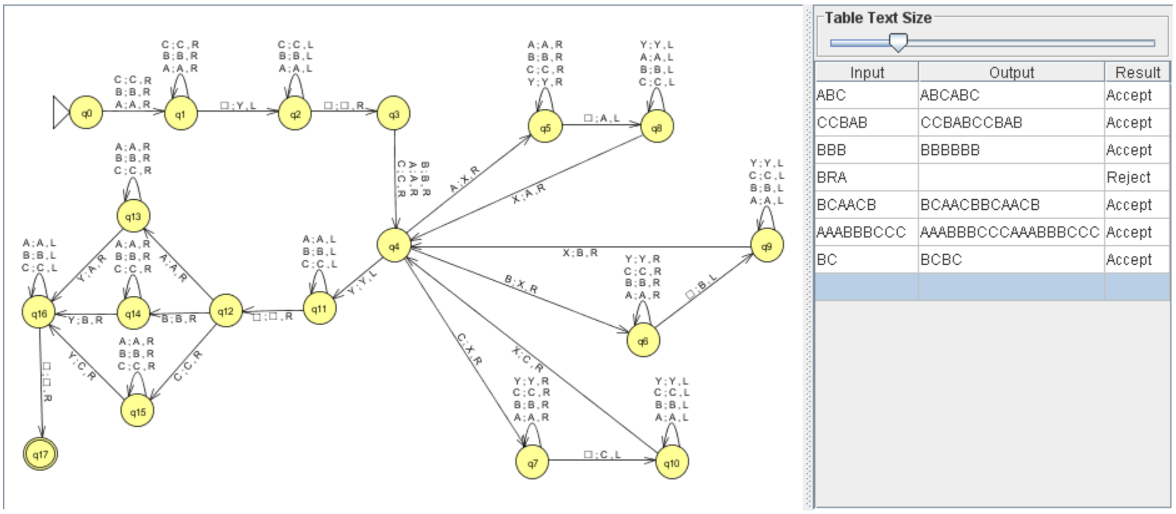
<u>Alternative solution</u>: $MT_{5.c}$=({a,b,c},{a,b,c,1,A,B,C,□},□,{q0,q1,q2,qf},q0,f,{qf}),



6. **Design a Turing Machine to generate a copy of a string with symbols {A,B,C}. For instance, given the input “bAABCAb”, the resulting input tape would be “bAABCAAABCAb”, where b represents the blank symbol.**

<u>Solution</u>:

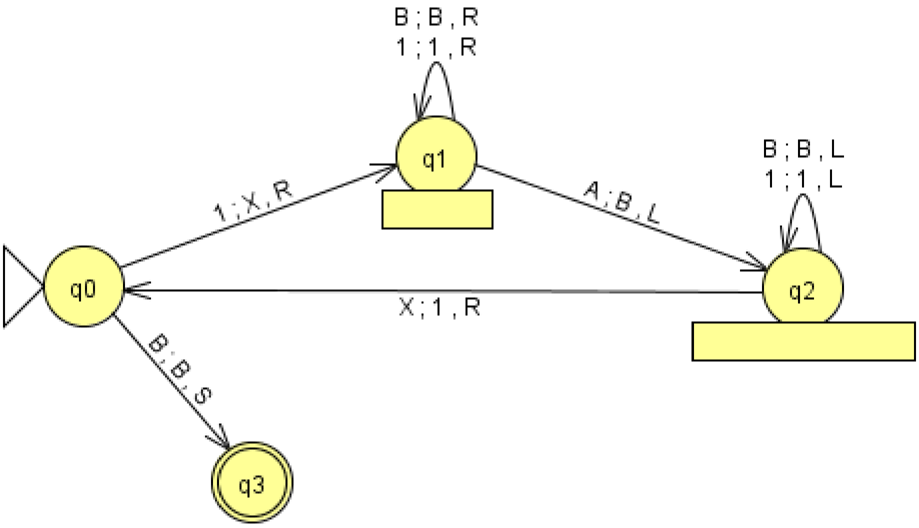
MT$_6$=({A,B,C},{A,B,C,X,Y,□},□,{q0,q1,q2,q3,q4,q5,q6,q7,q8,q9,q10,q11,q12,q13,q14,q15,q16,q17},q0,f,{q17}), where f:



7. **Design a Turing Machine which takes a input string with M 1’s and N A’s (M≤N), and replaces the M first A’s with B’s. For instance, given the input “b11AAAAAb” it would generate the input tape “b11BBAAAb”, where b represents the blank symbol (i.e., empty cells in the tape).**

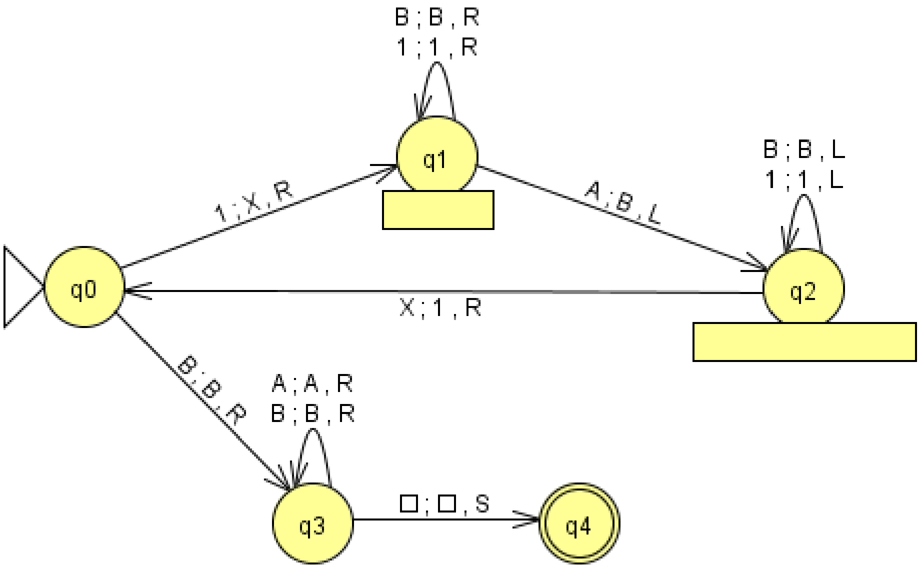
<u>Solution</u>:

MT$_{7.1}$=({1,A,B},{1,A,B,X,□},□,{q0,q1,q2,q3},q0,f,{q3}), where f:



<u>Alternative solution</u>:

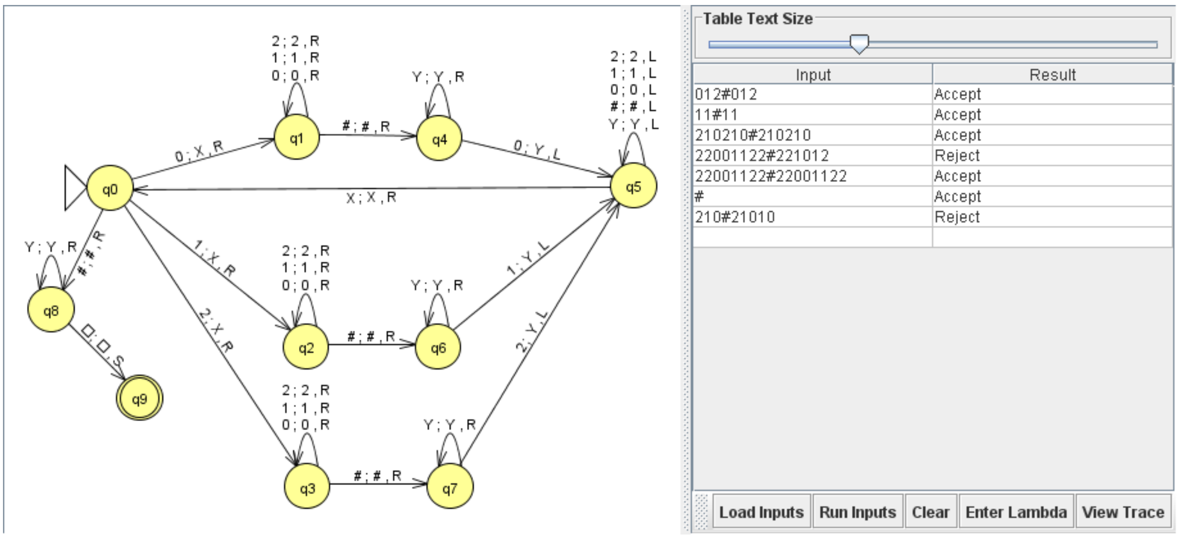
$MT_{7.2}$=({1,A,B},{1,A,B,X,☐},☐,{q0,q1,q2,q3,q4},q0,f,{q4}), where f:



8. Design a Turing Machine which takes two input words generated with the alphabet {0,1,2}, separated using the symbol {#}, and verifies whether they are the same. For instance, given the input b2101#2101b the Turing Machine would inform that both words are the same, where b represents empty cells in the tape.

Solution:

MT_8=({0,1,2,#},{0,1,2,#,X,Y,☐},☐,{q0,q1,q2,q3,q4,q5,q6,q7,q8,q9},q0,f,{q9}), where f:



Input	Result
012#012	Accept
11#11	Accept
210210#210210	Accept
22001122#221012	Reject
22001122#22001122	Accept
#	Accept
210#21010	Reject

9. Design a Turing Machine which obtains the successor of a binary number

Solution:

MT_9=({0,1},{0,1,□},□,{q0,q1,q2,q3,q4},q0,f,{q3}), where f:

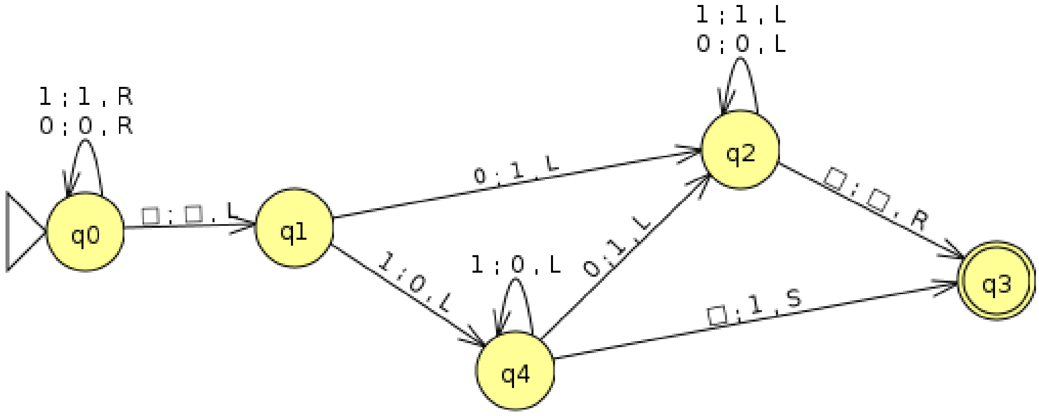


Table Text Size

Input	Output	Result
0	1	Accept
1	10	Accept
10	11	Accept
11	100	Accept
100	101	Accept
101	110	Accept
110	111	Accept
111	1000	Accept
1000	1001	Accept
1001	1010	Accept
1010	1011	Accept

Alternative solution: $MT_{9.2}$=({0,1},{0,1,□},□,{q0,q1,q2,q3},q0,f,{q3}), where f:

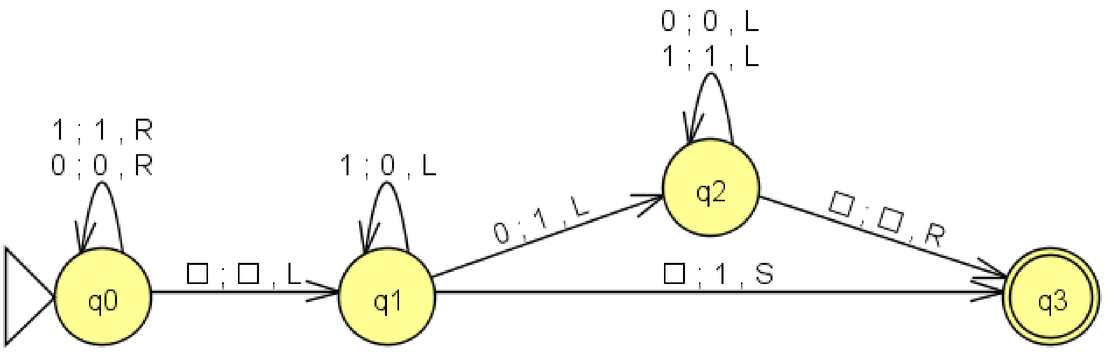


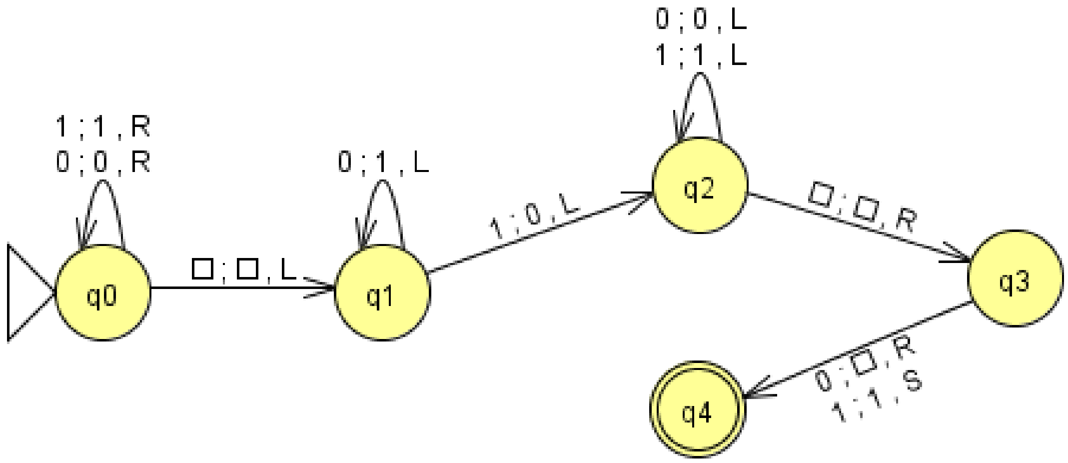
Table Text Size

Input	Output	Result
0	1	Accept
1	10	Accept
10	11	Accept
11	100	Accept
100	101	Accept
101	110	Accept
110	111	Accept
111	1000	Accept
1000	1001	Accept
1001	1010	Accept
1010	1011	Accept

10. Design a Turing Machine which obtains the predecessor of a binary number.

<u>Solution</u>:

MT_{10}=({0,1},{0,1,□},□,{q0,q1,q2,q3,q4},q0,f,{q4}), where f:



Input	Output	Result
1		Accept
10	1	Accept
11	10	Accept
100	11	Accept
101	100	Accept
110	101	Accept
111	110	Accept
1000	111	Accept
1001	1000	Accept
1010	1001	Accept
1011	1010	Accept