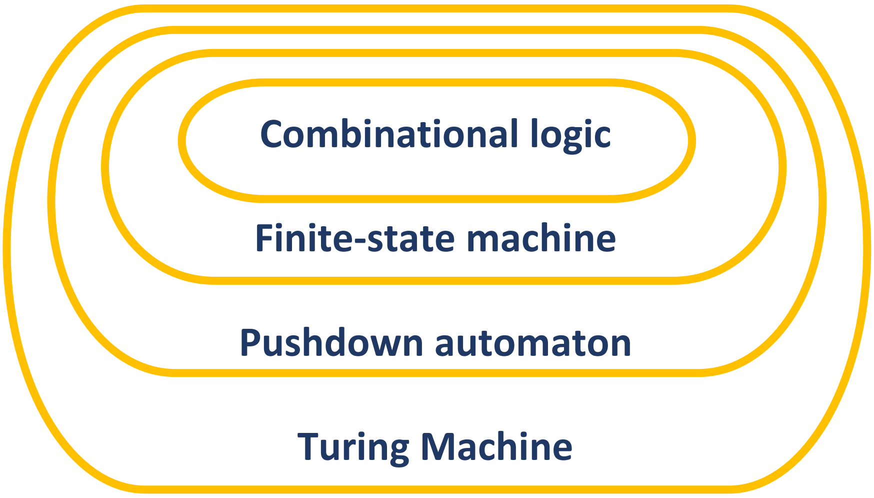
Turing Machine

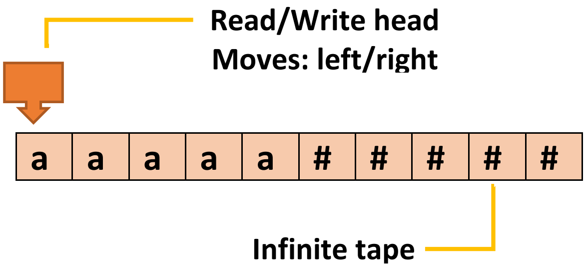
A Turing machine is a mathematical model of computation describing an abstract machine that manipulates symbols on a strip of tape according to a table of rules. Despite the model's simplicity, it is capable of implementing any computer algorithm.



Abilities of Turing machine

1. A Turing machine is similar to a DFA or PDF
2. It has the following abilities:
    - It can **read** or **write** to tape.
    - It can **move left** or **move right** on the tape.
    - It **halts** as soon as it reaches either the special **accept state** or the special **reject state**

Scheme of the Turing machine

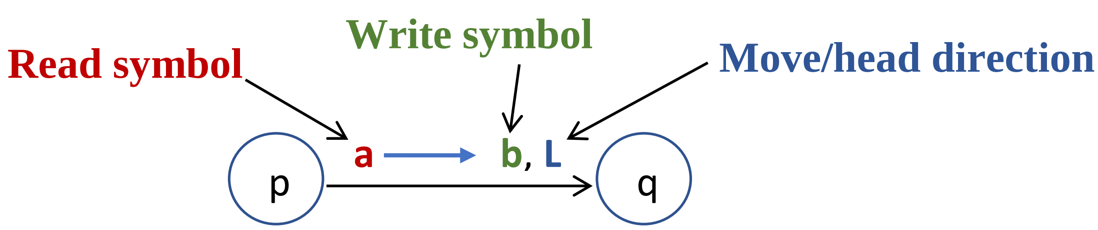


The formal definition of TM

A deterministic Turing Machine is a tuple consisting of several objects:

- Q is a finite set called the states.
- ∑ is a finite alphabet, a finite set not containing the **blank** symbol
- Γ is the tape alphabet, where ∑ ⊆ Γ and ϵ Γ
- δ:Q x Γ→ δ(Q x (Γ x{L,R})) is a function called the transition function
- S is an element of Q called the start state
- F is a subset of Q called the accepted states
- TM M= (Q, Γ, b, ∑ δ, S, F)

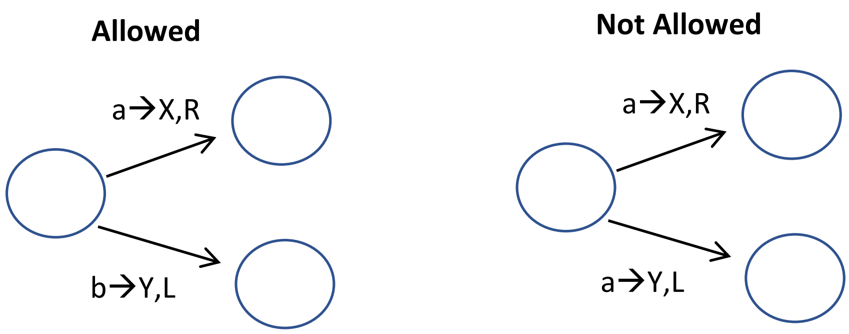
Transitions



δ (p, a) = (q, b, L)

The Language TM Accepts

- A Turing Machine accepts its input if it reaches an accepting configuration.
- The set of inputs it accepts is called its **language**.
- Turing Machine are deterministic



- Final states

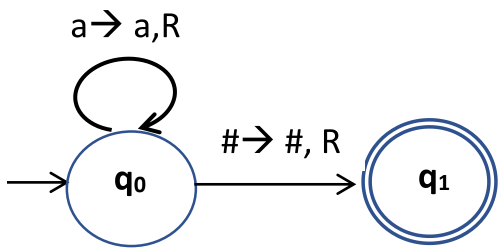
  - ➢ Final states have no outgoing transitions.
  - ➢ In a final state the machine halts.

- Acceptance

  - ➢ **Accept input:** If machine halts in a final state.
  - ➢ **Reject input:** If machine halts in a non-final state
    Or
    If machine enters an infinite loop.

TM example1:

Turing machine that accepts language a*



a	a	a	a	a	a	a	#	#	#

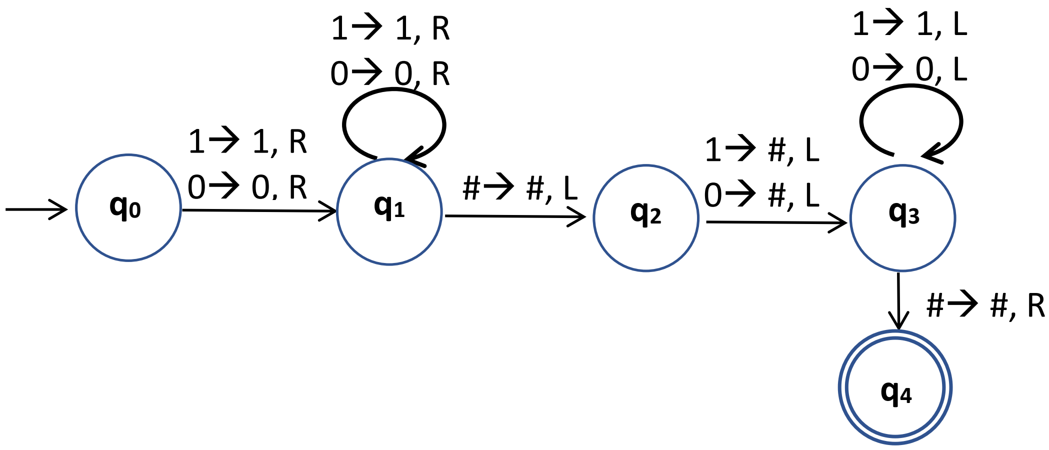
Ex2: Turing Machine to erase the first symbol in the right side

Sol:

Steps

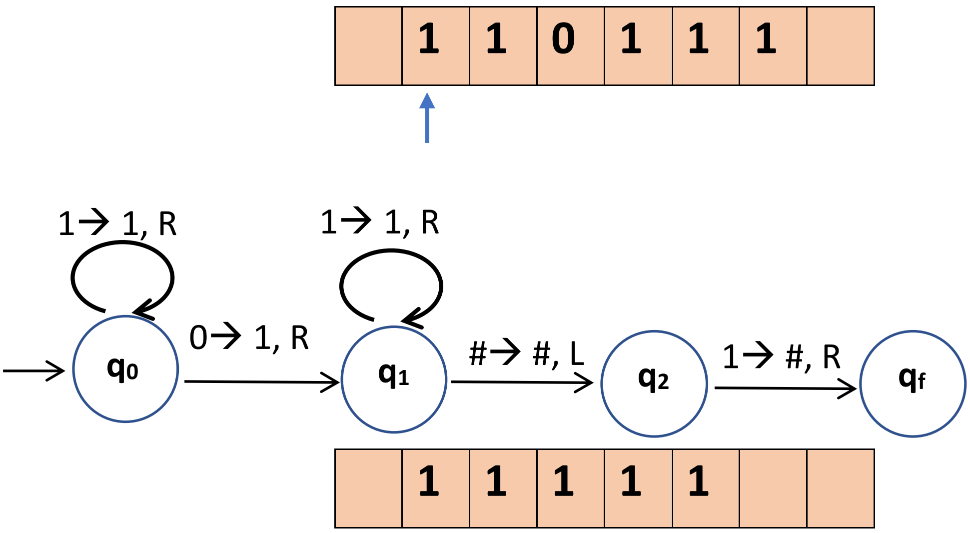
1. **Move the head in the right direction**
2. **Read symbols without change**
3. **When reading the symbol # move to the left**
4. **Change the symbol into # and move to the left**
5. **Move the head in the right direction and read without change**
6. **When reach the first symbol in the left move to the right and stop.**

#	0	0	1	1	1	#	#	#	#

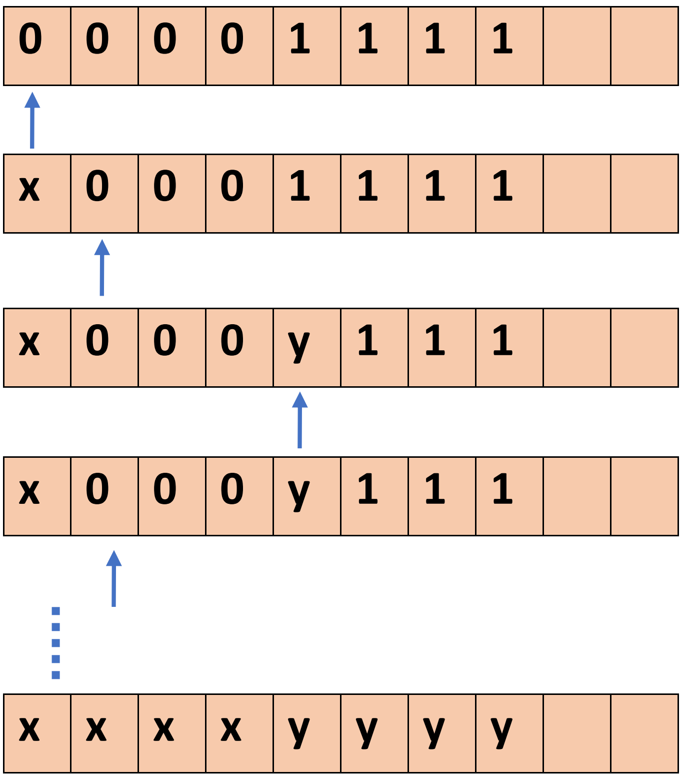


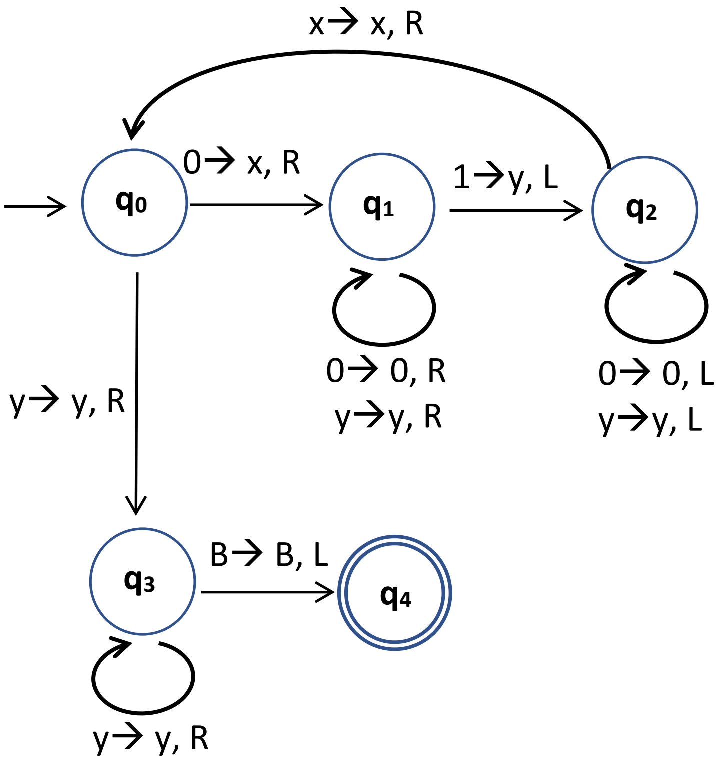
Ex2: design TM for binary addition is computable

Sol: test input (2,3) = 110111



Ex3: A Turing Machine M that accepts the language **L= {0^n1^n|n>=0}**





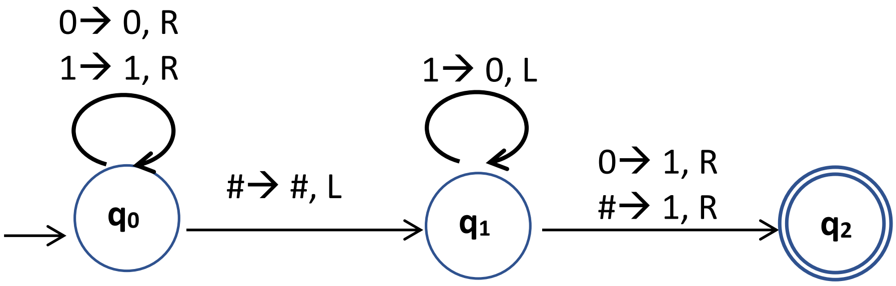
Ex3:

Design a TM to add 1 to a binary number x.

carry	**0**	**1**	**1**	**0**
X	**0**	**0**	**1**	**1**
				1
result	**0**	**1**	**0**	**0**

خطوات الحل

1. تقرأالالة رمز دون تغييره وتتحرك نحو اليمين.
2. تكرر الخطوة 1 حتى تصل الى الرمز #.
3. تتحرك خطوة واحدة نحو اليسار تقف عند اول مرتبة من جهة اليمين least significant bit
4. اذا كان الرمز 0 غيره الى 1 وتوقف .
5. اذا كان الرمز 1 غيره الى 0 وانتقل خطوة نحو اليسار.
6. كرر الخطوات 4 و5 حتى تصل الى الرمز #الذي يعني انتهاء العملية والتوقف عندها.



- X= 0110

#	0	1	1	0	#	#	#	#	#
#	0	1	1	1	#	#	#	#	#

- X=111

#	0	1	1	1	#	#	#	#	#
#	1	0	0	0	#	#	#	#	#