LECTURE 3

Finite State Automaton

Introduction

Any automaton whose outputs are either "yes" or "no" acts as a language acceptor. A finite automaton is a model of simple computing device, which acts as a language acceptor.

1. Definition of Finite State Automaton:

Languages that can be accepted by finite state automaton are regular languages, which can be generated by regular grammar.

A finite-state automaton (also called Finite State Machine (FSM)) is an idealized machine. It described as M = (S, I, f, s0, F)

where

S: a finite set of states,

I: a finite input alphabet

f: a transition function that assigns a next state to every pair of state and input, so that $f : S \times I \rightarrow S$ or S=f(S,I).

s0: an initial (or start) state

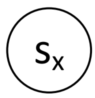
F: a subset F of S consisting of final (or accepting) states.

We can represent FSM using either next-state table or state diagram (also called transition diagram).

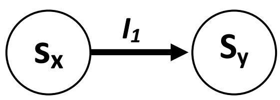
2. State Diagram (Transition Diagram):

The FSM can be visualized by state diagram as follows:

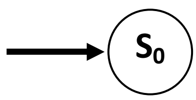
states are represented by circles.



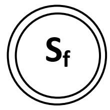
transition functions are represented by arrows with input symbols as labels over those arrows.



The initial state will have an arrow pointing to it that doesn't come from any state.



Final states are indicated by double circles.



3. Next-State Table:

The next-state table (see the table below) for an FSM shows the values of the transition function f for all possible states S and input symbols I .In the next-state table, the initial state is indicated by an arrow and the final states are marked by double circles.

The transition function, for instance, $f(s_0, I_1)= s_1$ can be translated as the following: when the FSM is in the state s_0 and the input symbol is I_1 the next state is s_1.

		f		
	state \ Input symbol	I_1	……	I_n
→	s_0	s_1	s_1	s_m
	s_1	s_0	s_1	s_0
	⋮	…	….	….
◎	s_m	….	…	…

input symbol

State of FSM

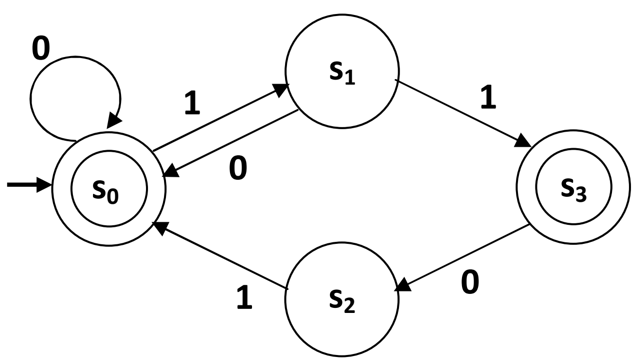
Transition function: where next State depending on the current state and the current input symbol

Example 1:

Construct the state diagram for the FSM $M = (S,I,f,s_0,F)$, where $S = \{s_0,s_1,s_2,s_3\}$, $I =\{0, 1\}$, $F =\{s_0,s_3\}$, and the transition function f is given below:

		f	
	State \ Input symbol	0	1
→ ◎	s_0	s_0	s_1
	s_1	s_0	s_3
	s_2	---	s_0
◎	s_3	s_2	---

Solution:



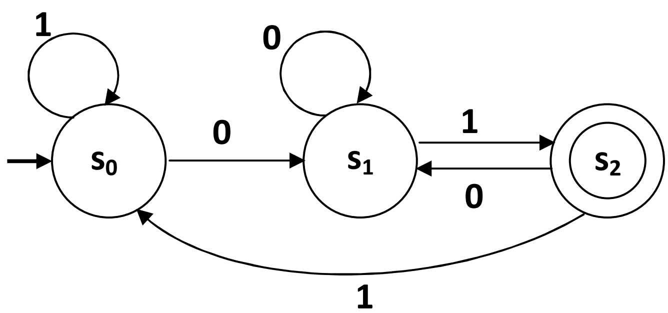
4. Language Recognition by FSM:

A string of input symbols is fed into a FSM in sequence. The state of the machine is changed after each successive input symbol. When the string is ends, the automaton ends up in a certain state, which may be either an final state or a not. The strings that send the FSM from initial state to a final state are said to be accepted by that machine.

The language accepted by M machine, denoted L(M), is the set of all strings that are accepted by M.

Example 2:

Consider the FSM M that is shown below.



Which of the following string is accepted by M and which one is rejected?

1) 01
2) 0011
3) 0101100
4) 10101

Solution:

1) 01: accepted, because the FSM begins with initial state and ends at final state

$S_0 \xrightarrow{0} s_1 \xrightarrow{1} s_2$

2) 0011: rejected, the FSM begins with initial state but ends at a state which is not final

$S_0 \xrightarrow{0} s_1 \xrightarrow{0} s_1 \xrightarrow{1} s_2 \xrightarrow{1} s_0$

3) 0101100: rejected, the FSM begins with initial state but ends at a state which is not final

$S_0 \xrightarrow{0} s_1 \xrightarrow{1} s_2 \xrightarrow{0} s_1 \xrightarrow{1} s_2 \xrightarrow{1} s_0 \xrightarrow{0} s_1 \xrightarrow{0} s_1$

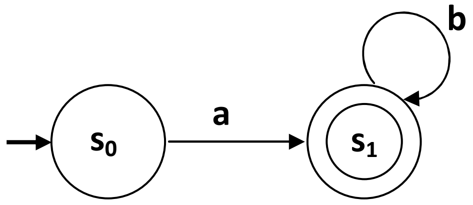
4) 10101: accepted, because the FSM begins with initial state and ends at final state

$S0 \xrightarrow{1} s0 \xrightarrow{0} s1 \xrightarrow{1} s2 \xrightarrow{0} s1 \xrightarrow{1} s2$

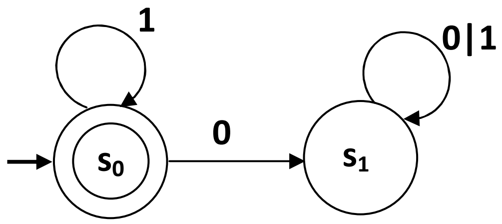
Example 3:

Determine the languages recognized by the following FSMs.

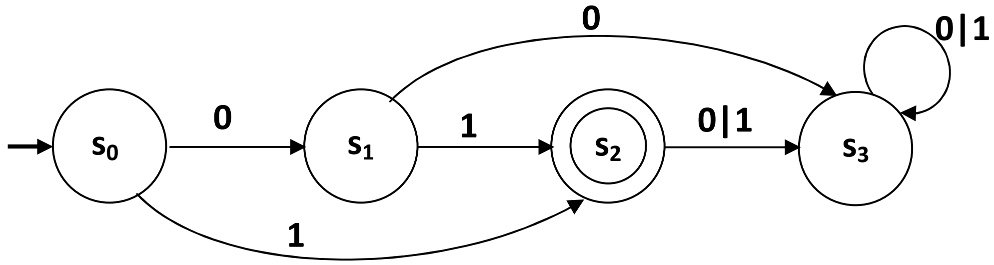
a) M_1:



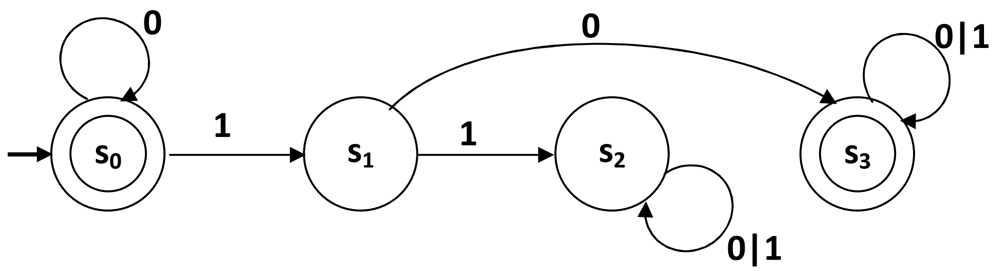
b) M_2:



c) M_3:



d) M_4:



Solution:

a) $L(M_1) = \{ a, ab, abb, abbb, abbbb, \ldots\}$
$L(M_1) = \{ ab^n \mid n \geq 0 \}$

b) $L(M_2) = \{ \varepsilon, 1, 11, 111, 1111, \ldots\}$
$L(M_2) = \{ 1^n \mid n \geq 0 \}$

c) $L(M_3) = \{ 1, 01\}$
$L(M_3) = \{ 0^n1 \mid n = 0, 1 \}$

d) $L(M_4) = \{ \varepsilon, 0, 00, 000, 00\ldots, 10, 10010, 0010, 00010, 0010101101\}$
$L(M_4) = \{ \varepsilon, 0^i10X \mid i \geq 0$ and X is any string of $0, 1 \}$

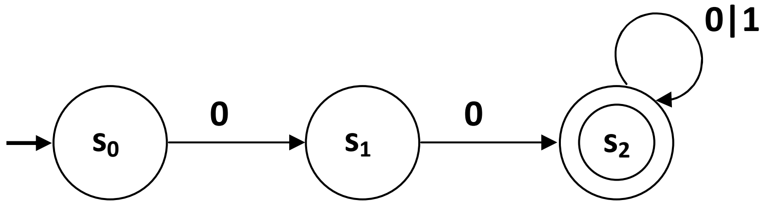
Example 4:

Construct FSMs that recognize each of the following languages:

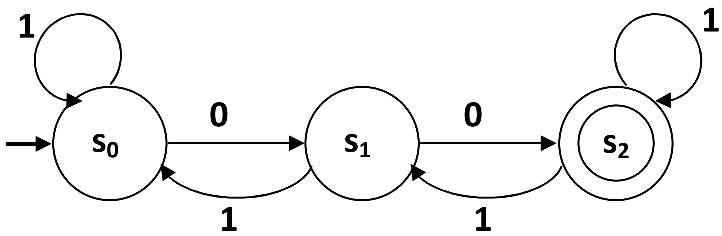
1) Set of bit string that begin with two 0s.
2) Set of bit string that contain two consecutive 0s.
3) Set of bit string that do not contain two consecutive 0s.
4) Set of bit string that end with two consecutive 0s.

Solution:

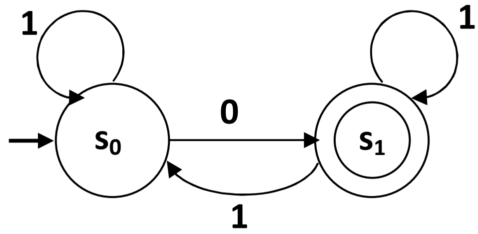
1)



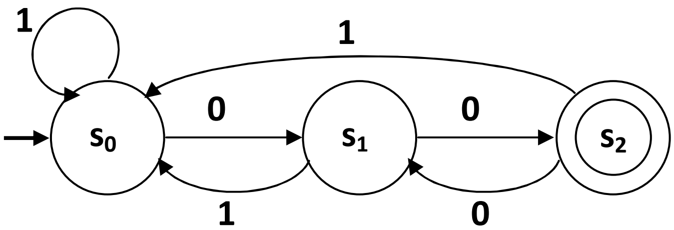
2)



3)



4)

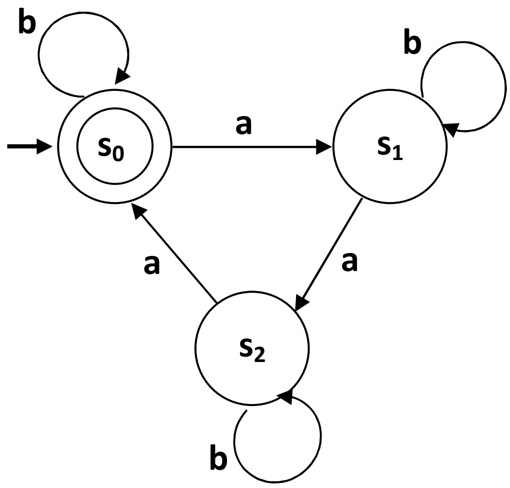


Example 5:

Design a FSM that accepts the set of all string of a and b such that the number of a in the string is divided by 3, what is the next-state table?

Solution:

	F	
Input symbol / state	a	B
s_0	s_1	s_0
s_1	s_2	s_1
s_2	s_0	s_2



5. Homework:

HW 1:

Construct the state diagram for the FSM M = (S, I, f, s0, F), where $S = \{s_0, s_1\}$, $I = \{0, 1\}$, $F = \{s_1\}$, and the transition function *f* is given below, what is the language recognized by that FSM?

		f	
	State \ Input symbol	0	1
→	s_0	s_0	s_1
◎	s_1	s_1	---

HW 2:

Design a FSM that accepts the set of all string of x and y that contain an odd number of x with any number of y but the string must end with at least two consecutive y.