Theory of Automata

Transducers

Dr. Sabina Akhtar

Revision

FINITE STATE AUTOMATA WITH OUTPUT/ TRANSDUCERS

Transducer

- The only output that we have seen finite automata produce so far is a yes/no at the end of processing.
- It is a generalisation of FSAs with an input/output pair on each arc. Its called a *Finite State Transducer.*
- There are two types of finite state machines that generate output:
  - Mealy Machine
  - Moore machine

Finite State Machines with Output (Mealy and Moore Machines)

- Finite automata are like computers in that they receive input and process the input by changing states.
- We will now look at two models of finite automata that produce more output than a yes/no.

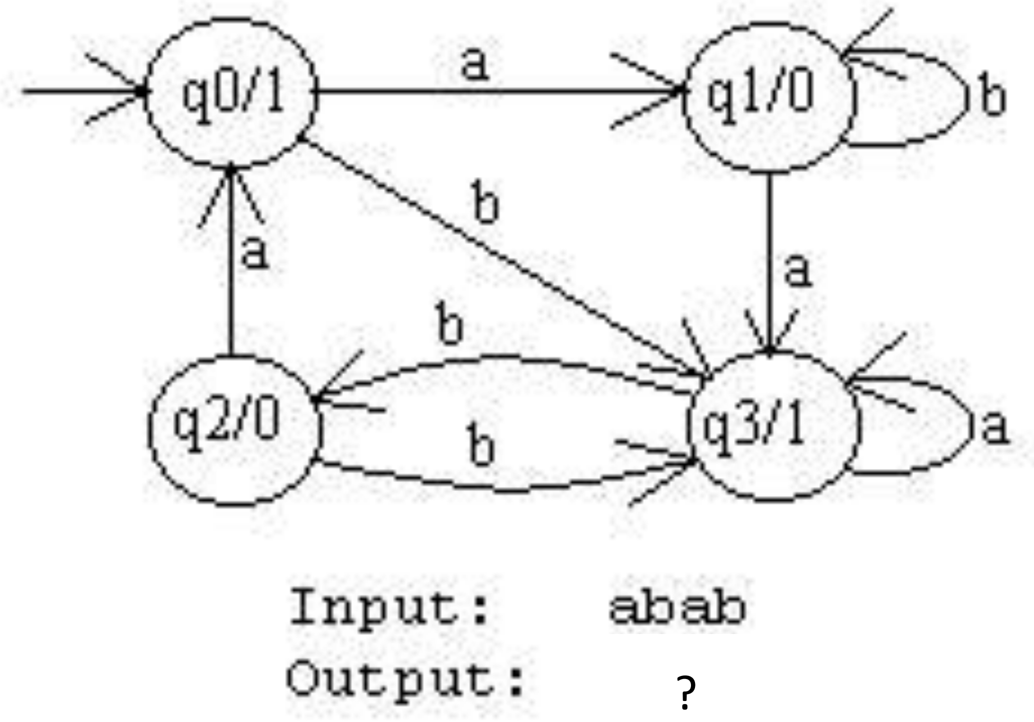
Moore machine

- Basically a Moore machine is just a FA with two extras.
  1. It has two alphabets, an input and output alphabet.
  2. It has an output letter associated with each state. The machine writes the appropriate output letter as it enters each state.

Formal definition

- Machine *M* can be described by a 6 tuple (Q, Σ, λ, Δ, δ, q_0) where:
  - Q : A nonempty finite set of states in M.
  - Σ : A nonempty finite set of input symbols.
  - λ : A nonempty finite set of outputs.
  - δ : It is a transition function that takes two arguments input state and input symbol.
  - Δ : It is a mapping function which maps Q to λ giving output associated with each transition.
  - q_0 : intial state

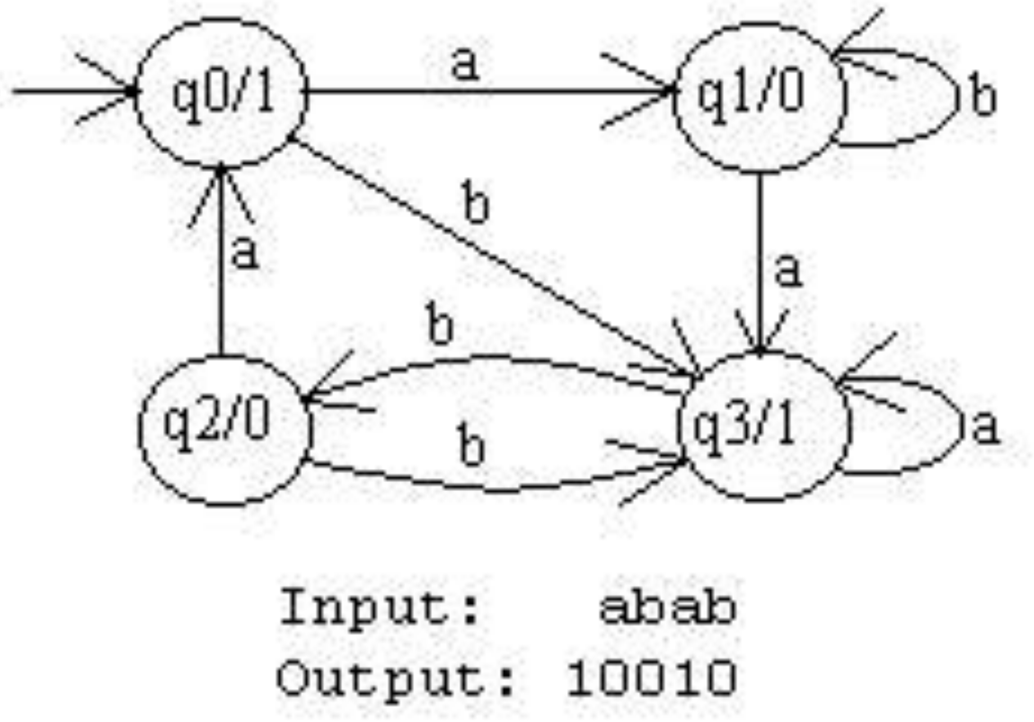
Example



?

- What is the output for abab?

Example



- What is the output for aabaabb?
- Output: 10101000

Formal definition

- Previous machine can be described by a 6 tuple $(Q, \Sigma, \lambda, \Delta, \delta, q_0)$ as:
  - Q : $\{q_{0,}\ q_{1,}\ q_{2,}\ q_3\}$.
  - Σ : {a, b}.
  - λ :{0, 1}.
  - δ : It is a transition function that takes two arguments input state and input symbol.
  - Δ : It is a mapping function which maps Q to λ giving output associated with each transition.
  - Initial state : q_0

Transition and mapping function

- Draw transition table for the previous example

Present state	**Input**		**Output**
	a	b	
q_0			
q_1			
q_2			
q_3			

Transition and mapping function

- Draw transition table for the previous example

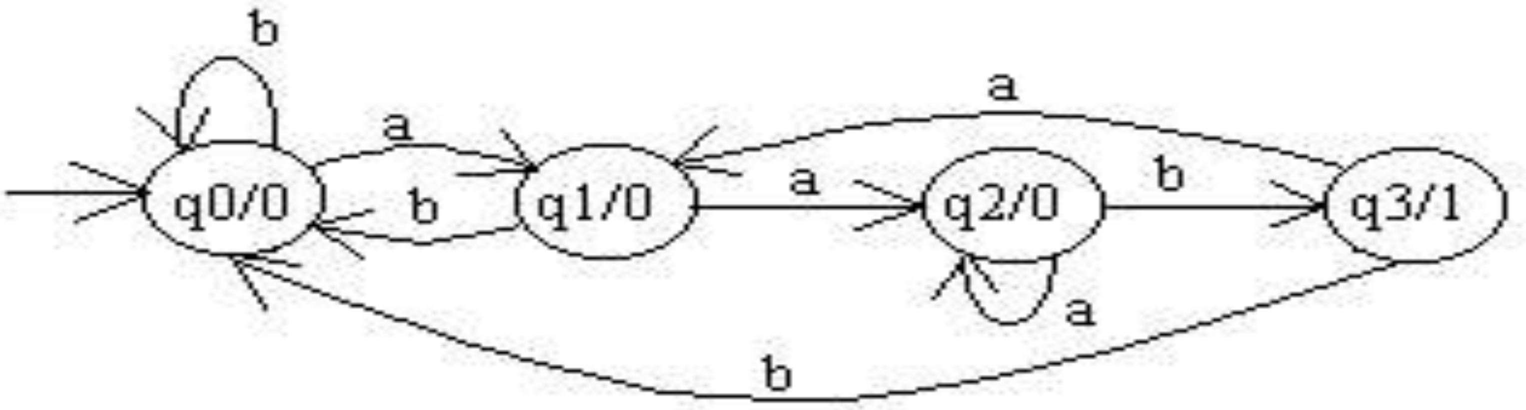
Present state	Input		Output
	a	b	
q_0	q_1	q_3	1
q_1	q_3	q_1	0
q_2	q_0	q_3	0
q_3	q_3	q_2	1

Class Activity

- Design a moore machine that outputs 1 each time it encounters a and 0 for b. In other words it counts the occurrences of a’s.
- Design a moore machine that counts the occurrences of ‘aab’ in the input string.

- e.g.,   aaababaaab
- Output : 00001000001

Solution




```
Input:    babaababaab
Output: 000000100001
```

Mealy Machine

- A Mealy Machine is an FSM whose output depends on the present state as well as the present input.

Mealy machine

Transitions are labelled *i/o* where

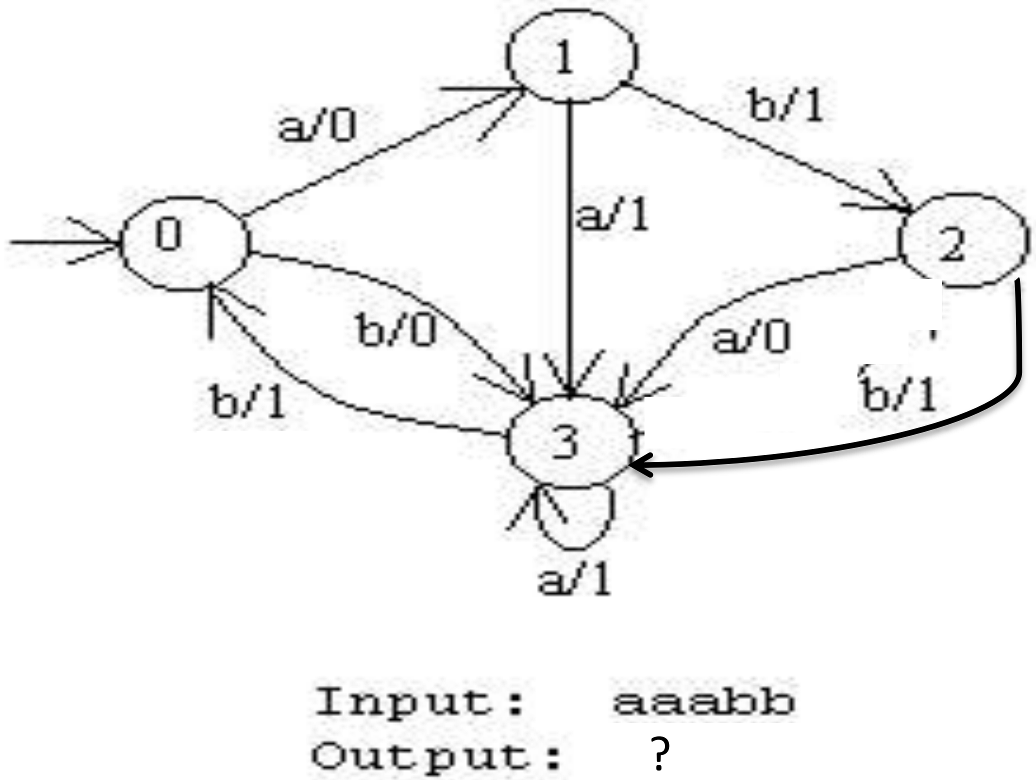
- ***i*** is a character in the input alphabet and
- ***o*** is a character in the output alphabet.

- Mealy machines are complete in the sense that there is a transition for each character in the input alphabet leaving every state.
- There are no accept states in a Mealy machine because it is not a language recogniser, it is an output producer. Its output will be the same length as its input.

Mealy Machine: Formal definition

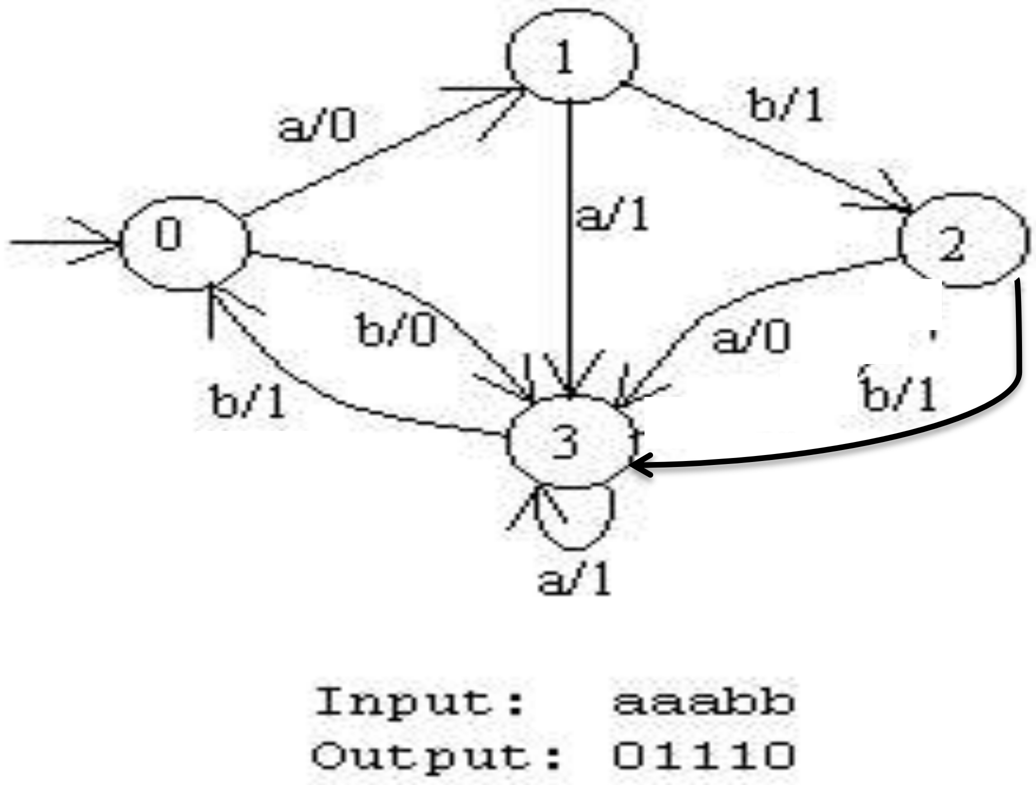
- Machine *M* can be described by a 6 tuple (Q, Σ, λ, Δ, δ, q_0) where:
  - Q : A nonempty finite set of states in M.
  - Σ : A nonempty finite set of input symbols.
  - λ : A nonempty finite set of outputs.
  - δ : It is a transition function that takes two arguments input state and input symbol.
  - Δ : It is a mapping function which maps Q*Σ to λ giving output associated with each transition.
  - q_0 : intial state

Example



- What is the output for aaabb?

Example



- What is the output for abbbaaab?

Mealy Machine: Formal definition

- Previous machine can be described by a 6 tuple $(Q, \Sigma, \lambda, \Delta, \delta, q_0)$ as:
  - Q : {0, 1, 2, 3}
  - Σ : {a, b}
  - λ : {0, 1}
  - δ : It is a transition function that takes two arguments input state and input symbol.
  - Δ : It is a mapping function which maps Q*Σ to λ giving output associated with each transition.
  - Initial state: 0

Transition and mapping function

- Draw transition table for the previous example

Present state	**Input a**		**Input b**	
	state	output	state	output
0				
1				
2				
3				

Transition and mapping function

- Draw transition table for the previous example

Present state	Input a		Input b	
	state	output	state	output
0	1	0	3	0
1	3	1	2	1
2	3	0	3	1
3	3	1	0	1

Class activity

- Design a mealy machine that takes the one's complement of its binary input. In other words, it flips each digit from a 0 to a 1 or from a 1 to a 0.

- Provide six-tuple values.
- Draw the transition table.

Solution


```
Input:  010110
Output: 101001
```

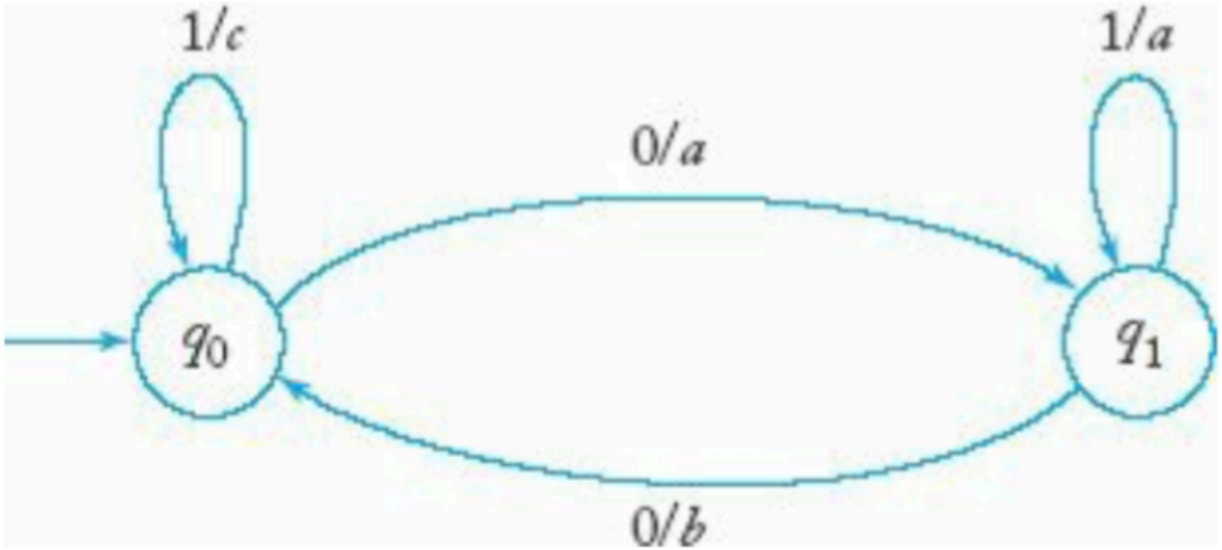
Class Activity

- $Q = \{q_0, q_1\}$,
- $\Sigma = \{0,1\}$,
- $O = \{a,b,c\}$, δ, θ,
- q_0 = initial state

$$\delta(q_0, 0) = q_1, \quad \delta(q_0, 1) = q_0,$$
$$\delta(q_1, 0) = q_0, \quad \delta(q_1, 1) = q_1,$$
$$\theta(q_0, 0) = a, \quad \theta(q_0, 1) = c,$$
$$\theta(q_1, 0) = b, \quad \theta(q_1, 1) = a$$

- Design the mealy machine for the above specifications.
- Provide the transition table.

Solution



•

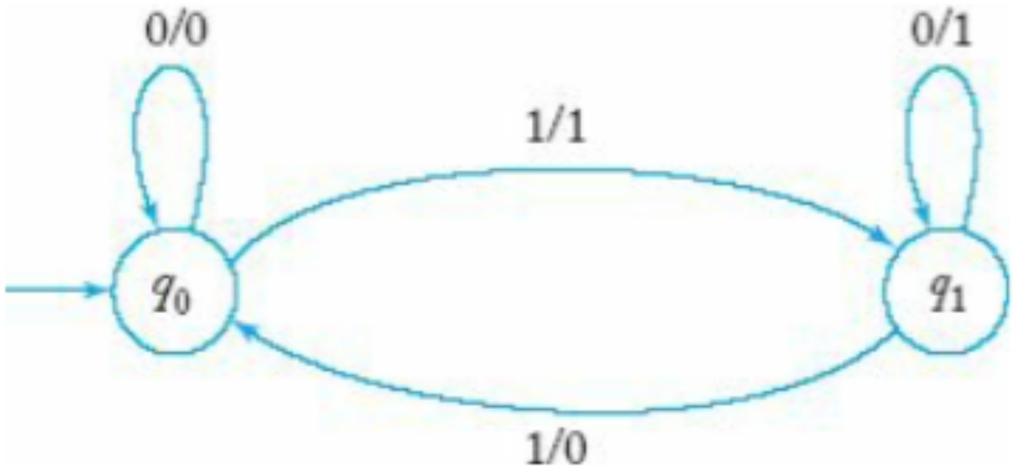
Class Activity

- Construct a mealy machine M that takes as input strings of 0’s and 1’s. Its output is to be a string of 0’s until the first 1 occurs in the input, at which time it will switch to print 1’s . This is to continue until the next 1 in encountered in the input, when the output reverts to 0. The alternation continues everytime a 1 is encountered.
- For example:
  - Input : 0010010
  - Output : 0011100

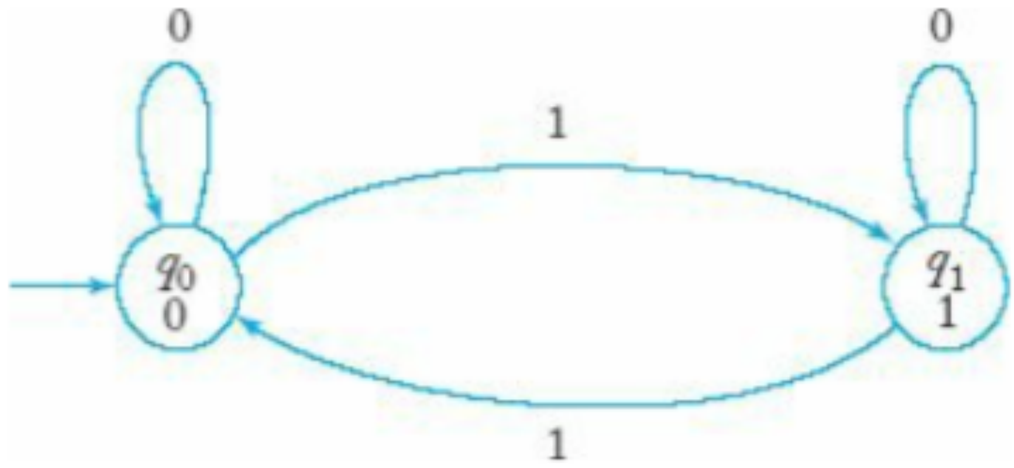
Class Activity

- Construct a Moore machine

Solution

- 

Solution 2



References

- Book: An introduction to formal languages and automata, fifth edition by Peter Linz

- Lectures by Ralf Möller, Hamburg Univ. of Technology