Finite Automata

BITS Pilani
Pilani Campus

Shashank Gupta
Assistant Professor
Department of Computer Science and Information Systems

Today's Agenda

Key Points to be Covered

- Finite Automata
- Deterministic Finite Automata (DFA)
- Designing Problems

Finite Automaton (FA)

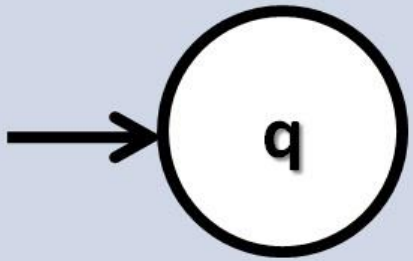
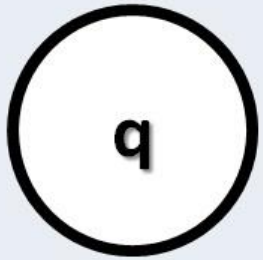
FA

A mathematical model that consists of finite set of states and related transitions.

States are generally represented by circles (labelled with state name).

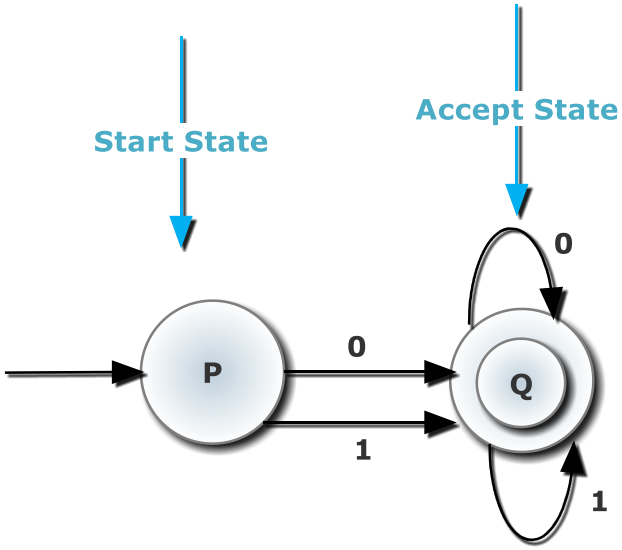
Transitions are usually represented by directional arrows labelled with input alphabet symbols.

Symbols Utilized in FA

STATES	NOTATIONS
Initial State	
Intermediate State	
Final State	

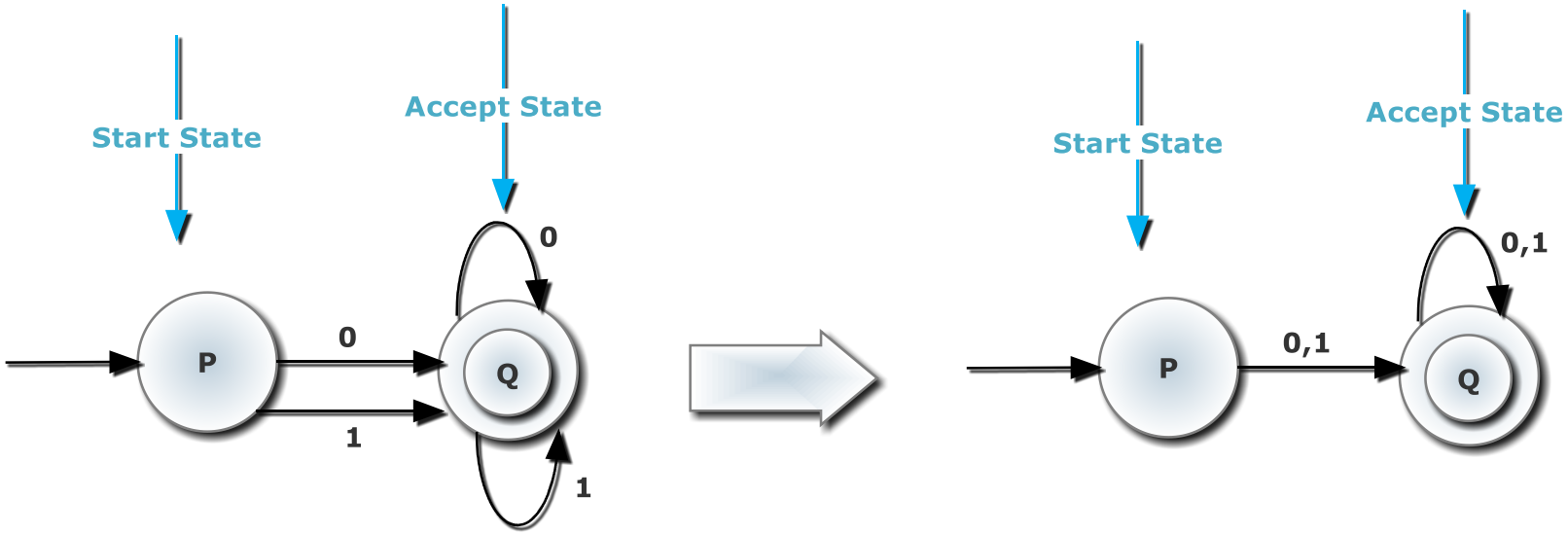
Representation of FA (Continued…)

- A Finite Automaton over $\Sigma = \{0,1\}$



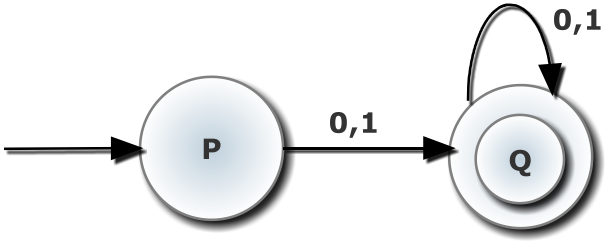
Representation of FA (Continued…)

- A Finite Automaton over $\Sigma = \{0,1\}$



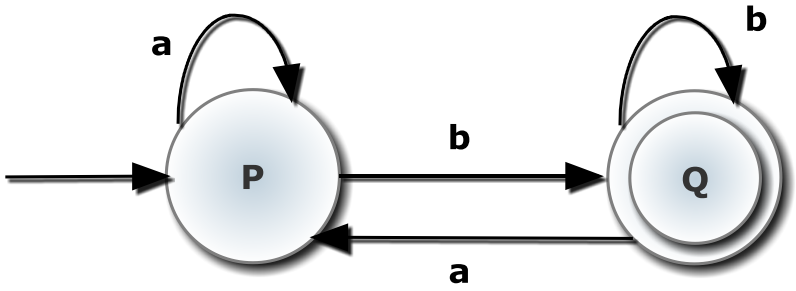
Language Acceptance Mechanism by FA

- FA accepts only those strings in the language L which ends up in any of its accept states.



L ={0, 1, 00, 11, 010, 110, 11110001100, ---------}

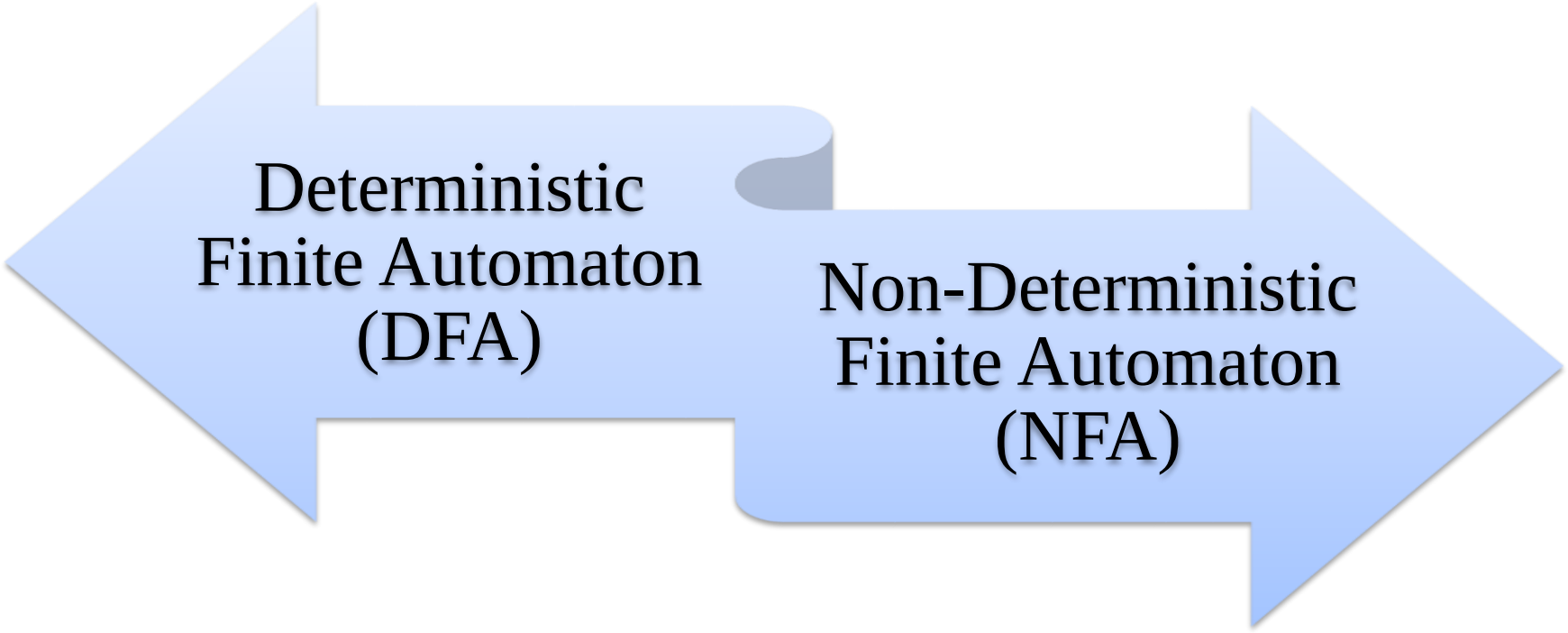
Language Acceptance Mechanism by FA (Continued…..)



Here, Language accepted by the above FA is L = {b, ab, aab, baabb,------}

However, strings like a, aa, ba ------- are not going to be accepted by this FA.

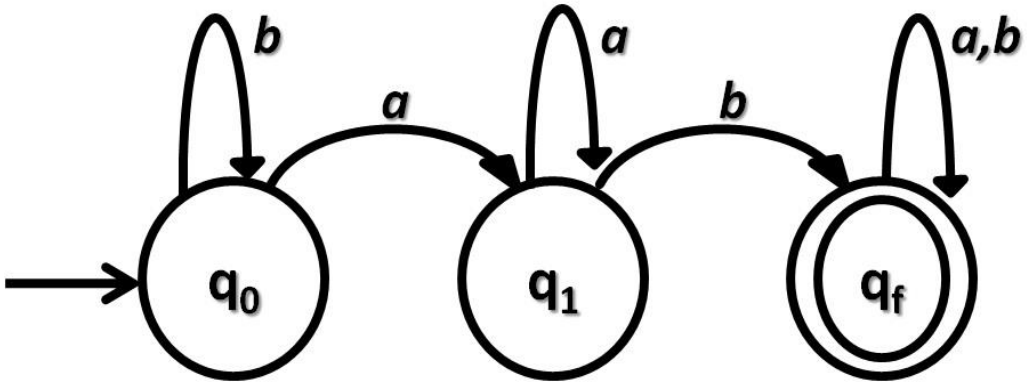
Types of Finite Automaton



Deterministic Finite Automata

In DFA, there must be exactly one transition from each state over each symbol of given input alphabet Σ.

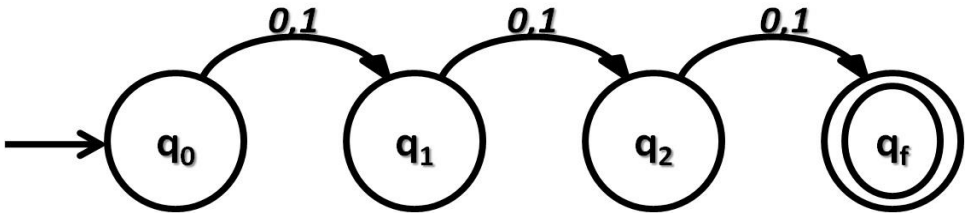
- Consider the following DFA over Σ = {a, b}



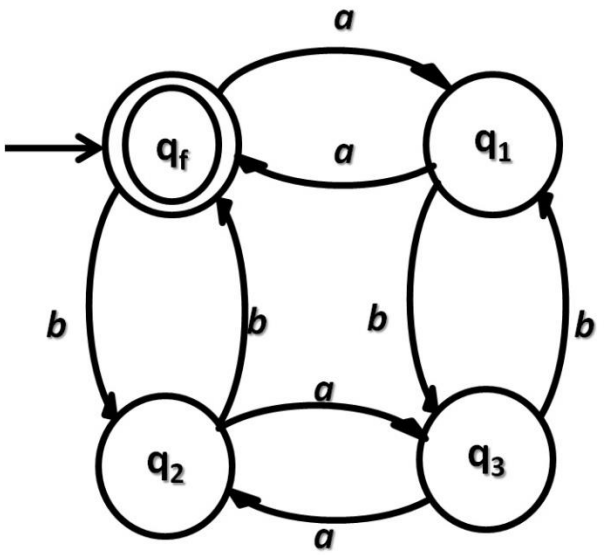
More Examples

Consider the following Finite Automata and identify whether they are DFA or not

FA over Σ ={0,1}



FA over Σ ={a,b}



Steps for Designing of DFA

Step 1

- Construct the language ‘L’ starting with the string of minimum length and gradually move towards other strings.

Step 2

- Design the FA for the minimal length string identified in Language L.

Step 3

- Finally, transform that FA into DFA for all the other additional strings specified in the Language ‘L’.

Designing Examples of DFA

Design a DFA which accepts all strings of a's and b's where each string ends with symbol 'b'.

DFA = ?

$\Sigma = \{a, b\}$

Example (Continued……)

Design a DFA which accepts all strings of a's and b's where each string ends with symbol 'b'.

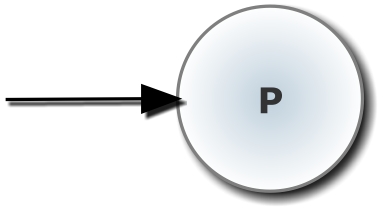
Step 1: Initially, construct the language 'L' over Σ = {a, b} starting with the string of minimum length (i.e. single 'b')

- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}

Example (Continued……)

Step 2: Secondly construct the FA for the minimal length string (i.e. ‘b’) from the language ‘L’.

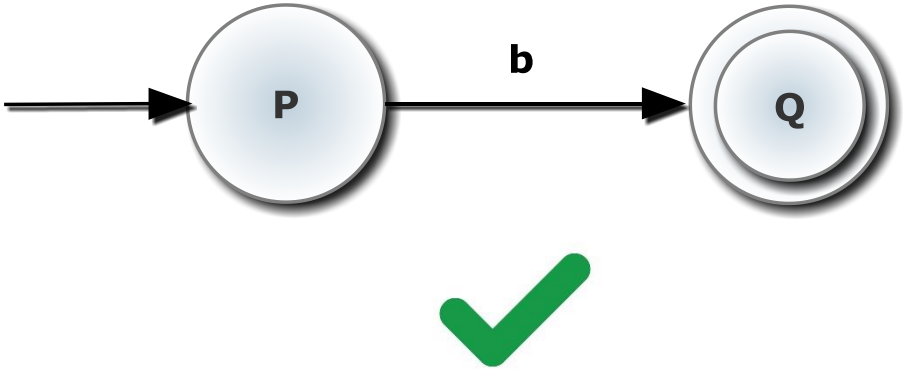
- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}

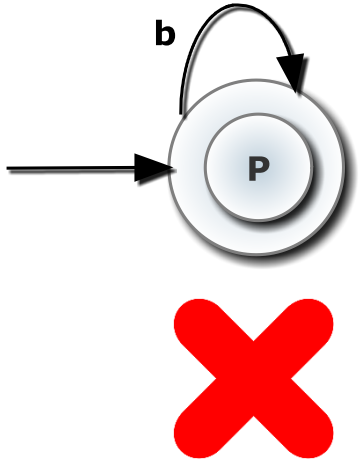


Example (Continued……)

Step 2: Secondly construct the FA for the minimal length string (i.e. ‘b’) from the given language ‘L’.

- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}

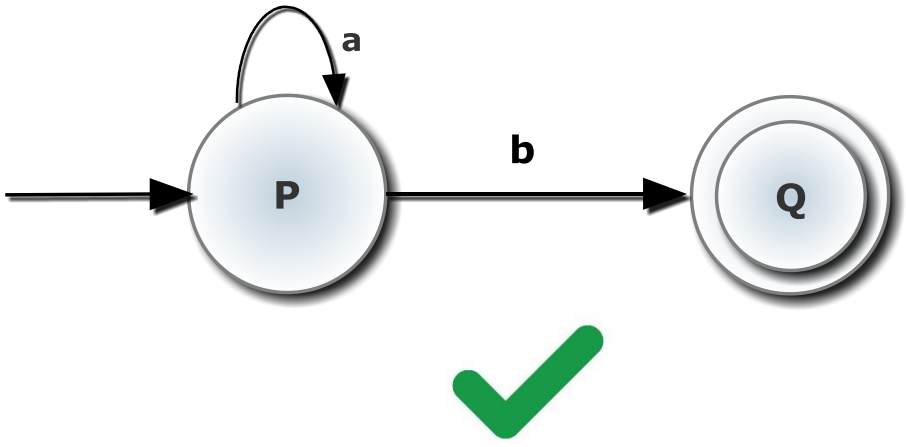


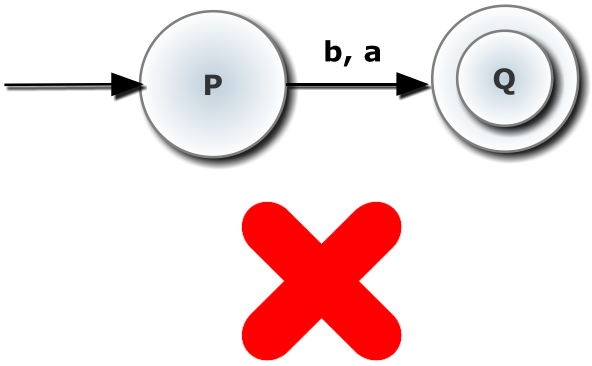


Example (Continued……)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}

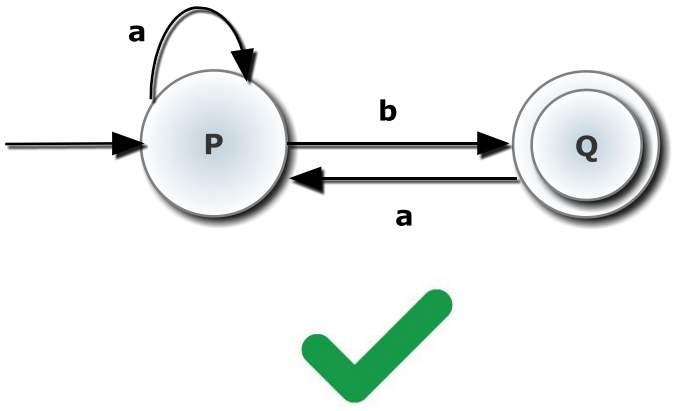


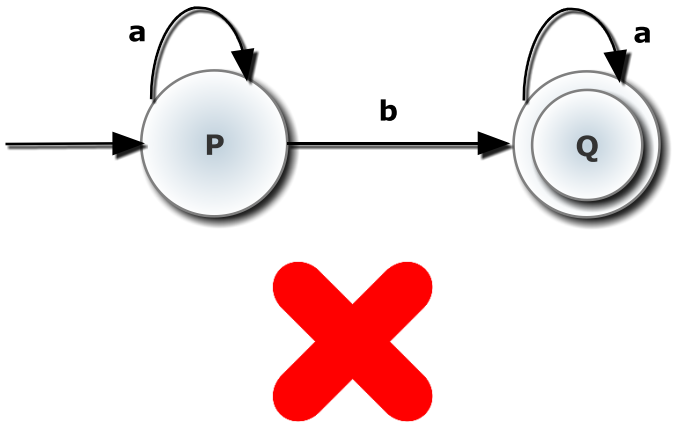


Example (Continued……)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}

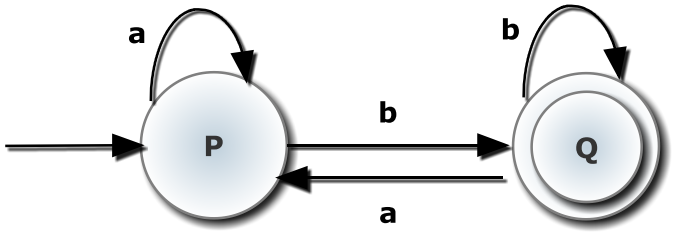


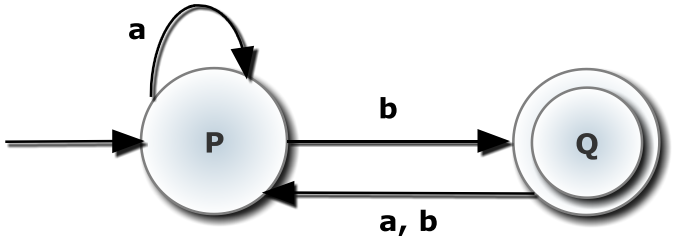


Example (Continued……)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {**b**, a**b**, b**b**, aa**b**, bba**b**, baba**b**, -------}





Formal Definition of DFA

- It is represented by 5 tuples $(Q, \Sigma, \delta, q_0, F)$ where
  - Q: Finite Set of States
  - Σ: Input Alphabet
  - δ: Transition Function i.e. $Q \times \Sigma \rightarrow Q$
  - $q_0 \in Q$ is the Start State
  - $F \subseteq Q$ is the Set of Accept States

  In DFA, from each and every state, there is exactly one transition over each symbol of input alphabet Σ.

  A DFA $(Q, \Sigma, \delta, q_0, F)$ accepts a string w by starting at a initial state q_0 and the DFA ends at an accept state by reading the string w.

Deterministic Finite Automata (DFA)

Now, according to the definition of the DFA i.e. M = {Q, ∑, $, q0, F}, here

-Q: {P, Q}

-∑: {a, b}

-$: $(P,a) → P

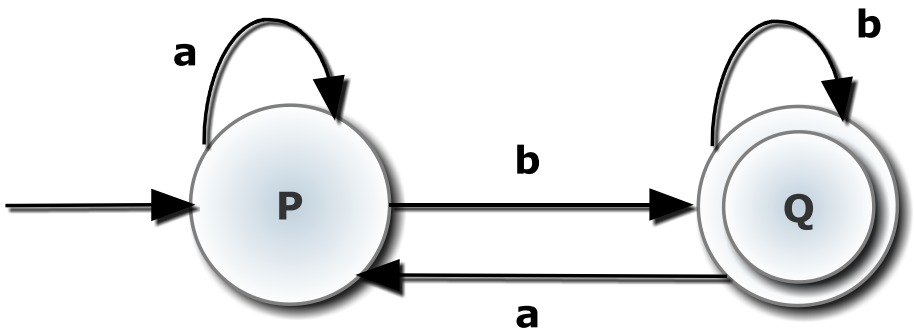
$(P,b) → Q

$(Q,a) → P

$(Q,b) → Q

-q0: P

-F: {Q}



Some More Examples on DFA

Construct a minimal DFA which accepts all strings of 0's and 1's where each string starts with symbol '0' and ends with the symbol '1'.

DFA = ?

$\Sigma = \{0, 1\}$

Example (Continued…..)

Construct a minimal DFA which accepts all strings of 0's and 1's where each string starts with symbol '0' and ends with the symbol '1'.

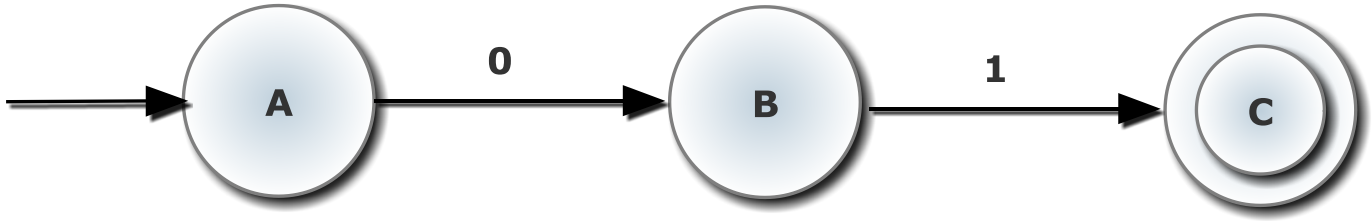
Step 1: Initially, construct the language 'L' over Σ = {0, 1} starting with the string of minimum length (i.e. "01")

- L = {**01**, **0**0**1**, **0**1**1**, **0**10101, **0**11101, ------}

Example (Continued…..)

Step 2: Secondly construct the FA for the minimal length string (i.e. ‘01’) from the language ‘L’.

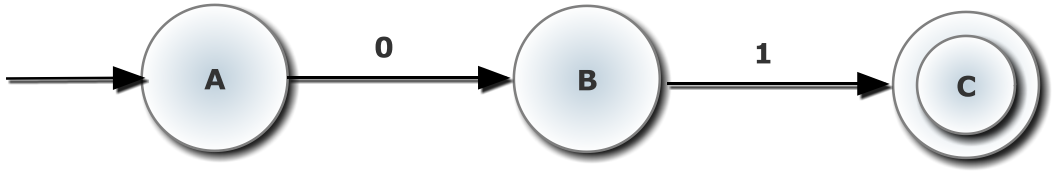
- L = {**01**, **0**0**1**, **0**1**1**, **0**1010**1**, **0**1110**1**, ------}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {**01**, **0**0**1**, **0**1**1**, **0**10101, **0**11101, ------}

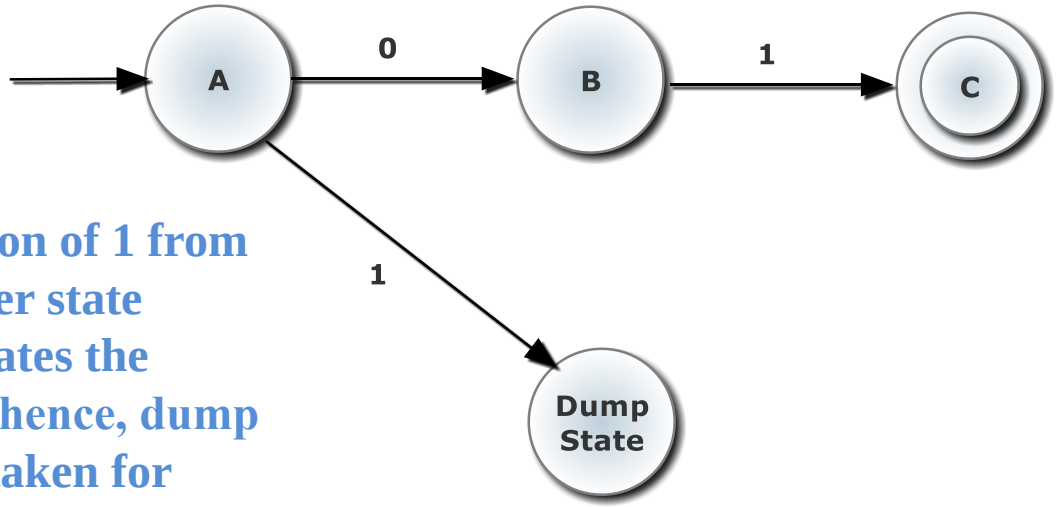


A Dump/Dead State (DS) is a state from where the automaton cannot reach to an accept state.

Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {**01**, **0**0**1**, **0**11, **0**10101, **0**11101, ------}

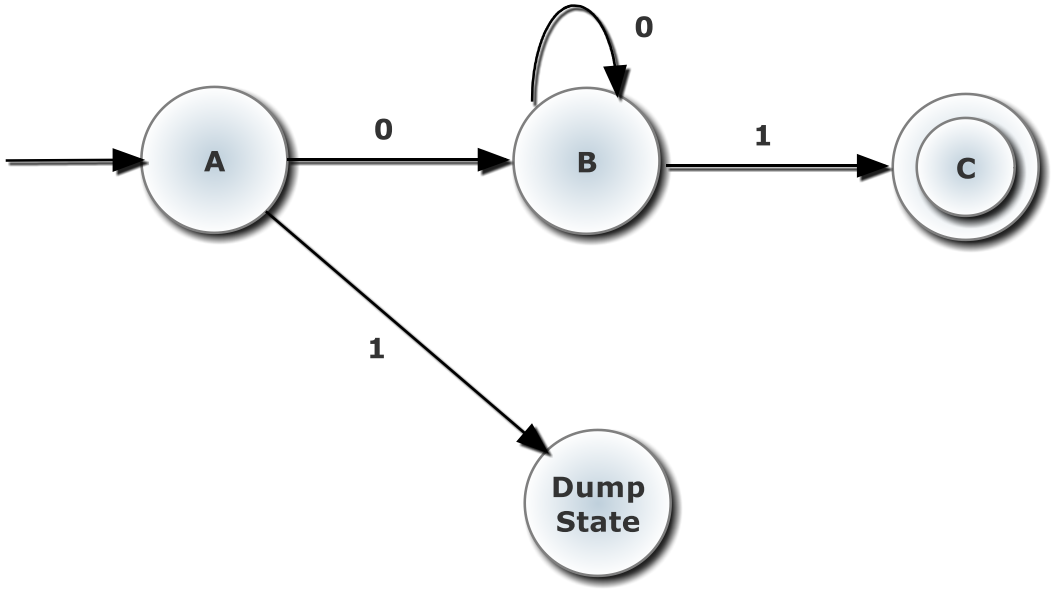


Since, the transition of 1 from state A to any other state (including A) violates the condition of “L”, hence, dump state needs to be taken for including the transition of 1 from A to dump state.

Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

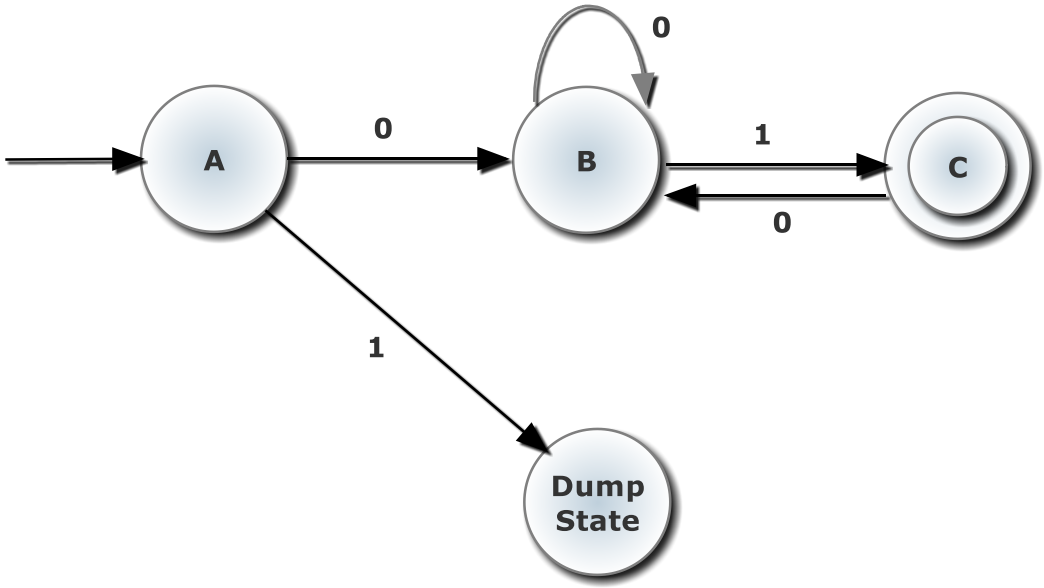
- L = {**01**, **0**0**1**, **0**1**1**, **0**10**1**0**1**, **0**111**01**, ------}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

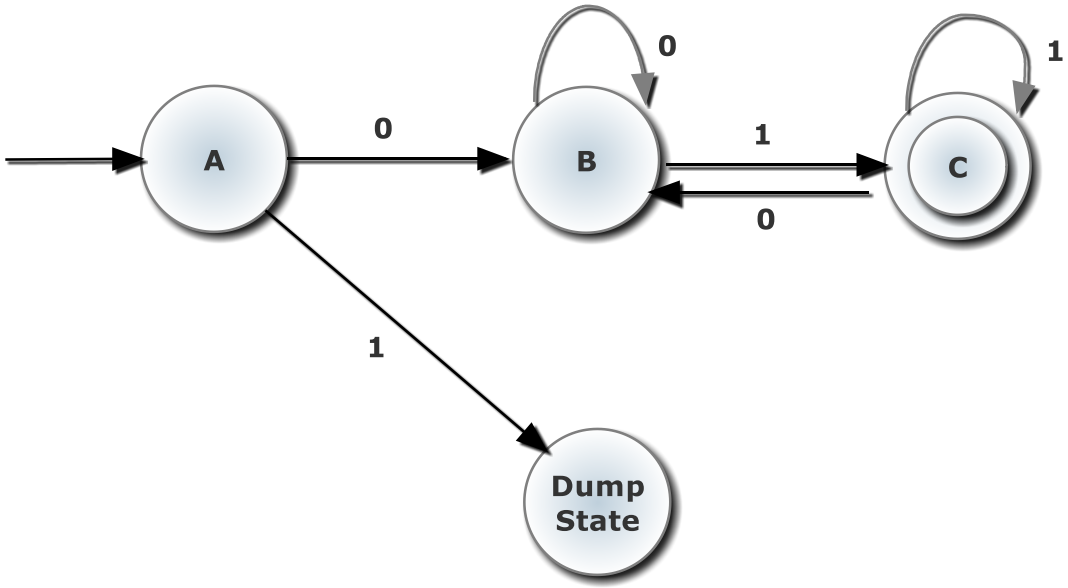
- L = {**01**, **0**0**1**, **0**1**1**, **0**10101, **0**11101, ------}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

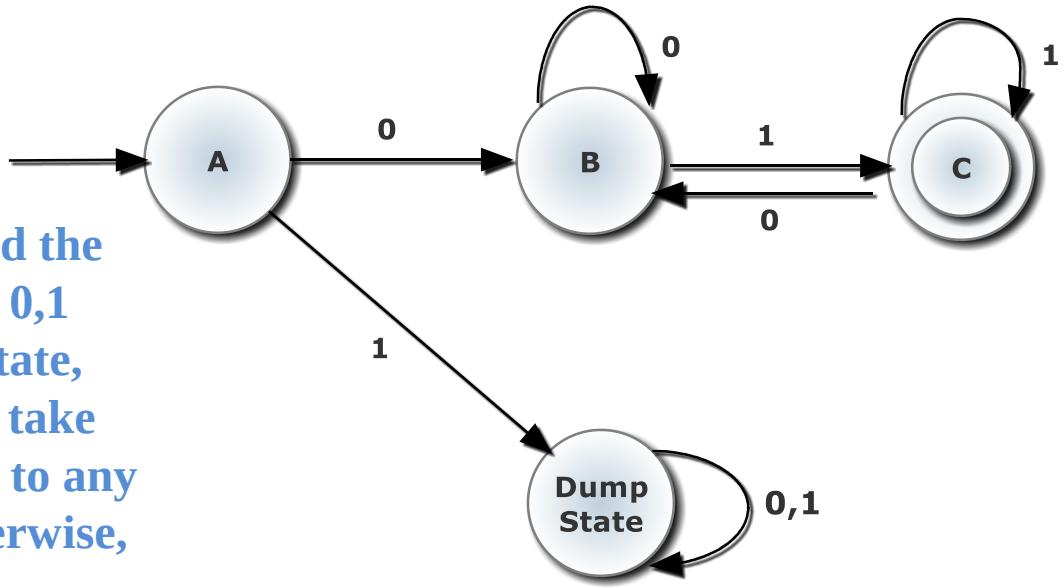
- L = {**01**, **0**0**1**, **0**1**1**, **0**10101, **0**11101, ------}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language 'L'.

- L = {**01**, **0**0**1**, **0**1**1**, **0**10**1**0**1**, **0**111**01**, ------}



We have included the self transition of 0,1 onto the dump state, since, we cannot take these transitions to any other state. Otherwise, this violates the meaning of dump state.

Home work Assignment

Construct a minimal DFA which accepts all strings of 0's and 1's where each string does not start with symbol '0' and does not end with the symbol '1'.

Thank You