Lecture four

Topics that must be covered in this lecture:

- ***Deterministic finite state automata DFSA***
- ***Nondeterministic finite state automata NDFSA***
- ***Comparison between DFA and NDFA***
- ***The equivalence of DFAS and NDFSA***

Deterministic finite state automata DFSA*:* it is an acceptor for any state and input character has at most one transition state that the acceptor changes to. If no transition state is specified the input string is rejected.

A DFSA is a 5-tuple (Q,Σ, δ, q0, F), where:

1. Q is a finite set called the states,
2. Σ is a finite set called the alphabet,
3. δ : Q × Σ → Q is the transition function where given state and an input alphabet delivers the next state to be visited.
4. q0 ∈ Q is the start state, and
5. F ⊆ Q is the set of accept states (final state).

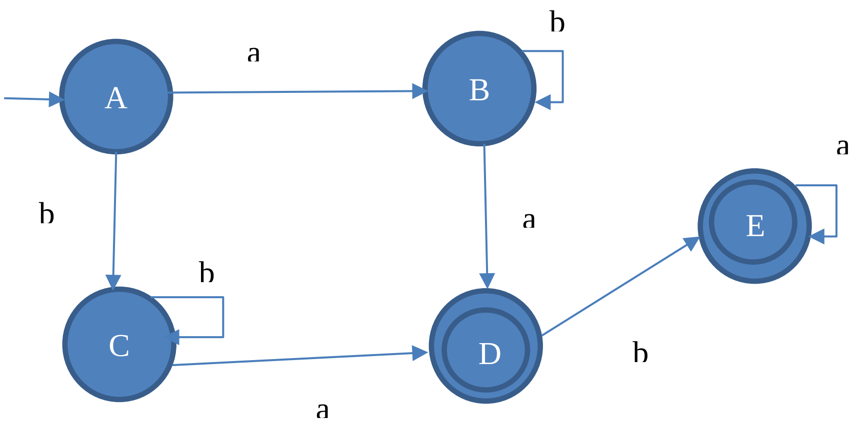
- accept string when $\delta(\mathrm{q0},\mathrm{w}) = p, p \in F$

The language accepted by M, designated L(M), language of automata:

L(M): $\{\mathrm{w} |\ \delta(q0, w) = p, p \in F\}$

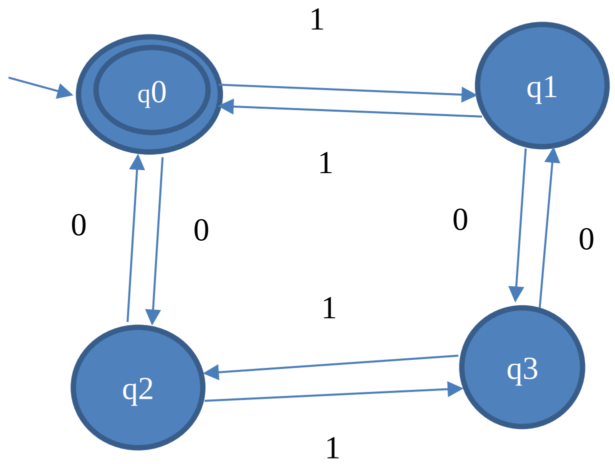
Example1: consider the DFSA, M=({A,B, C, D, E}, {a,b}, δ , A, {D,E}) where $\delta(\mathrm{A},\mathrm{a}) = \mathrm{B}, \delta(\mathrm{A},\mathrm{b}) = \mathrm{C}, \delta(\mathrm{B},\mathrm{a}) = \mathrm{D}, \delta(\mathrm{B},\mathrm{b}) = \mathrm{B}, \delta(\mathrm{C},\mathrm{a}) = \mathrm{D}, \delta(\mathrm{C},\mathrm{b}) = \mathrm{C}, \delta(\mathrm{D},\mathrm{b}) = \mathrm{E}, \delta(E, a) = E$.

Draw state digraph for this example.



Example2: consider the DFSA, M=({q0,q1, q2, q3}, {0,1}, δ , q0, {q0}) where $\delta(q0,0) = q2, \delta(q0,1) = q1, \delta(q1,0) = q3, \delta(q1,1) = q0, \delta(q2,0) = q0, \delta(q2,1) = q3, \delta(q3,0) = q1, \delta(q3,1) = q2$.
Draw state digraph and transition table for this example, and then check input string **110101** accepted by M?

1-



2- Transition table:

Σ / **Q**	**0**	**1**
q0	q2	q1
q1	q3	q0
q2	q0	q3
q3	q1	q2

3-Suppose input string 110101 entered to M, check string is accepted?
Start machine in state q1:

$q0 \xrightarrow{1} q1 \xrightarrow{1} q0 \xrightarrow{0} q2 \xrightarrow{1} q3 \xrightarrow{0} q1 \xrightarrow{1} q0 \in F$
the string is accepted

Nondeterministic finite state automata NDFSA:

NDFSA it is FA that allow one or more transition from a state on the same input symbol.

NDFSA is a 5-tuple M=(Q,Σ, δ, q0, F) where Q,Σ, q0, F are the same as in DFSA and δ can be defined as total transition function $\delta : Q \times \Sigma \rightarrow 2^{Q}$ (2^{Q} : is the power set of Q, the set of all subset of Q).

- Transition function δ({q0,q1},a)= $\delta(q0, a) \cup \delta(q1, a)$

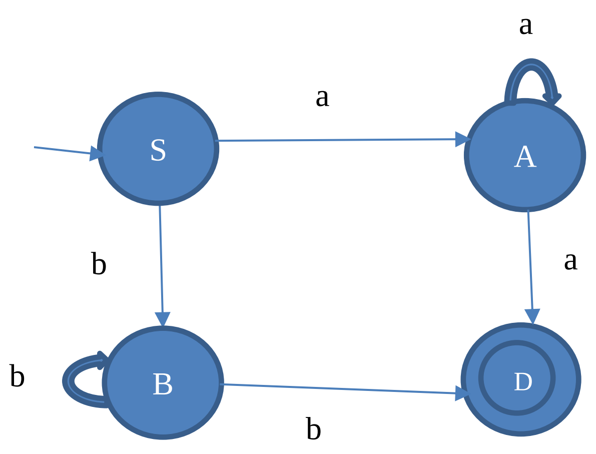
Example1: consider the FSA:

M=({A,B,S,D},{a,b}, δ, S ,{D}) where δ is given by:

$\delta(S, a) = \{A\}, \delta(S, b) = \{B\}, \delta(A, a) = \{A, D\}, \delta(A, b) = \emptyset,$

$\delta(B, a) = \emptyset, \delta(B, b) = \{B, D\}, \delta(D, a) = \emptyset , \delta(D, b) = \emptyset$

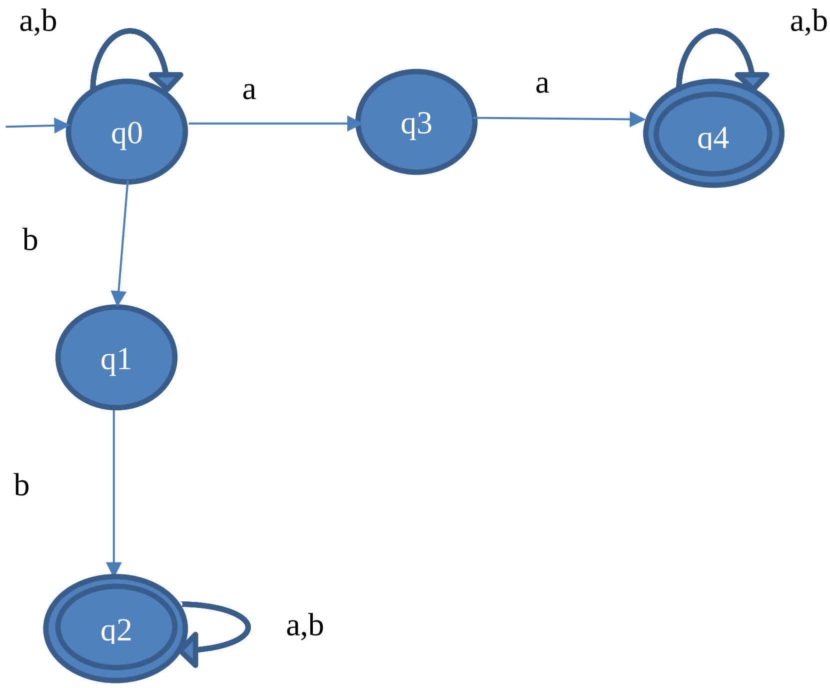
draw state diagram and transition table.



Note: an input string is accepted by NDFSA if there exist a sequence of transition for the given string that leads from initial state to some final states.

$\{w \in \Sigma^* | \ \delta(Q, w) \cap F \neq \emptyset\}$.

Example2: consider the NDFSA, M=({q0,q1,q2,q3,q4}, {a,b}, δ, q0, {q2, q4}) suppose you have the state digraph mentioned below, find transition table for this example, and check is the input **abaab** accepted or not ?



Transition table:

Σ / Q	a	b
q0	{q0,q3}	{q0,q1}
q1	Ø	{q2}
q2	{q2}	{q2}
q3	{q4}	Ø
q4	{q4}	{q4}

$\delta(q0, abaab) = \delta(\delta(q0, a), baab) = \delta(\{q0, q3\}, baab) =$

$= \delta\big(\delta(q0, b) \cup \delta(q3, b), aab)\big) = \delta(\{q0, q1\} \cup \emptyset, aab)$

$= \delta(\{q0, q1\}, aab) = \ \delta(\delta(q0, a) \cup \ \delta\big((q1, a), ab\big) = \delta(\{q0, q3\} \cup \emptyset, ab)$

$\delta(\{q0, q3\}, ab) = \delta(\delta(q0, a) \cup \delta(q3, a), b) = \delta(\{q0, q3\} \cup \{q4\}, b) =$

$\delta(\{q0, q3, q4\}, b) = \delta\big(\delta(q0, b) \cup \delta(q3, b) \cup \delta(q4, b)\big) = \{q0, q1, q4\}$

$\{q0,q1,q4\} \cap \{q2, q4\} = \{q4\} \in F$

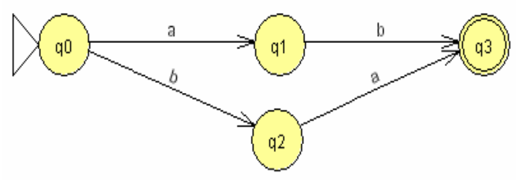
The input string **abaab accepted**

Comparison between Deterministic finite state automata and Nondeterministic finite state automata

Deterministic Finite Automata DFA

DFA: different input from state to different states

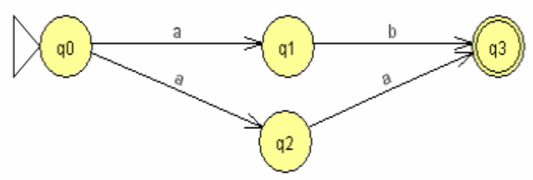
DFA: This DFA has not an arrow with the label ϵ.



Non Deterministic Finite Automata NFA

NFA: one input from state to different states

NFA: This NFA has an arrow with the labelϵ.



The Equivalence of DFSA and NDFSA:

For every NDFSA can constructed an equivalent DFSA (one which accepts the same language).

Language L can be accepted by a DFSA if L can be accepted by NDFSA .

To convert NFA to DFA we can carry out the following algorithm:

1- begin with {q0}, start state, and calculate δ ({q0},a) for all a in Σ this gives a number of new states Q^-.

2- For each new states Q^-, we again calculate δ (Q^-,a) for all a in Σ and introduce new states if necessary.

3- Repeat step2 until there are no new states.

4- Final states of new DFA are the states that contain any final state of the previous NFA.

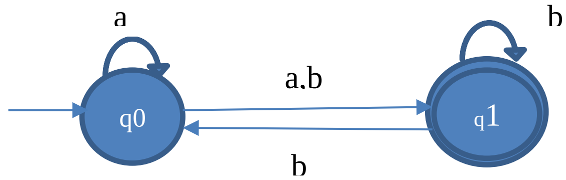
Example1: let M=({q0,q1},{a,b}, $\delta, q0, \{q1\}$) be an NFSA where :

$$\delta(q0, a) = \{q0, q1\}, \delta(q1, a) = \emptyset, \delta(q0, b) = \{q1\},$$
$$\delta(q1, b) = \{q0, q1\}$$

draw state diagram for this NFA, then convert this NFA to DFA.

<u>Solution:</u>

NFA- state diagram:



NFA- transition table:

Σ / Q	a	b
q0	{q0,q1}	{q1}
q1	∅	{q0,q1}

DFA- transition table:

Σ / Q	a	b
→ **q0**	{q0,q1}	{q1}
+q1	*C*	{q0,q1}
+q0q1	q0q1	q0q1
C	C	C

DFA state diagram: