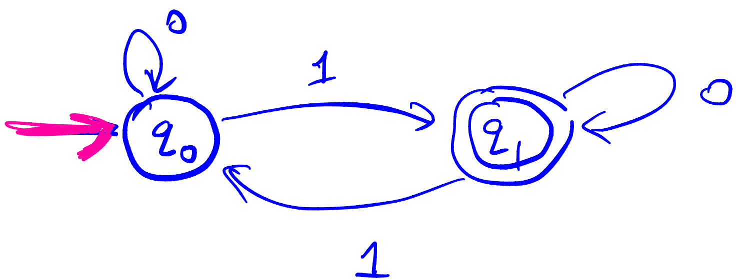
Lecture 2

<u>Last class</u>: admin stuff, coure resources, intro
Started: Regular Languages + DFA

<u>Today</u>: More about Regular Languages + DFAS

Regular Languages and Finite Automata

Example 1:



$w = 01101$ accepted
$w = 1100$ rejected
$w = 1110$

δ:

	1	0
q_0	q_1	q_0
q_1	q_0	q_1

$\mathcal{L}(M) = \{ w \in \{0,1\}^* \mid w \text{ contains an odd number of } 1\text{'s} \}$

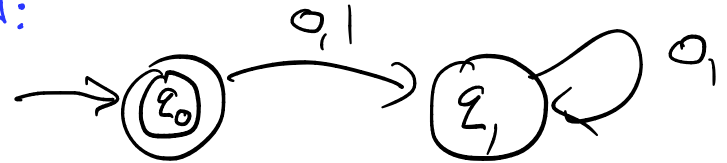
$w = 01011$

r_0	r_1	r_2	r_3	r_4	r_5
q_0	q_0	q_1	q_1	q_0	q_1
	0	1	0	1	1

Example 2

$L = \{\varepsilon\}$

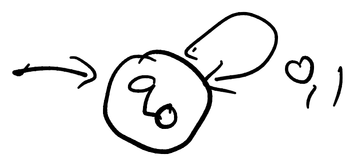
DFA:



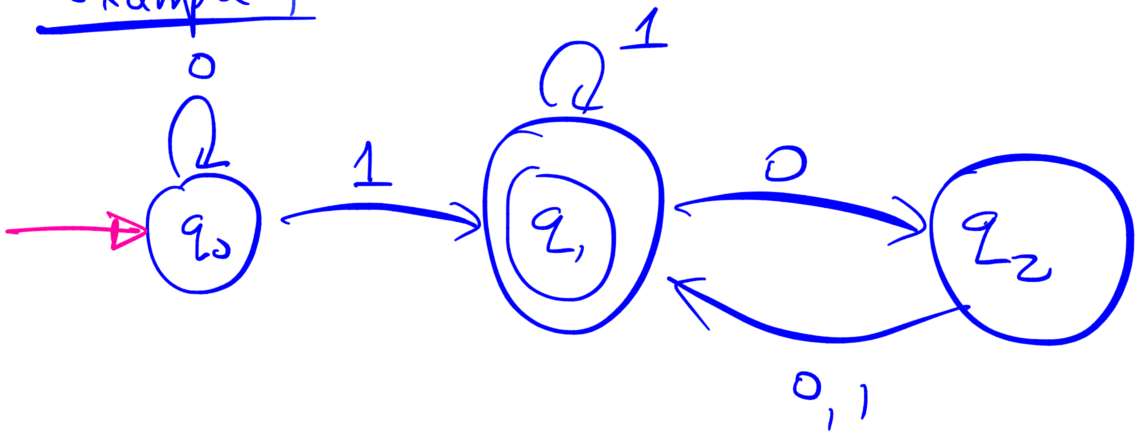
Example 3

$L = \phi$

DFA:



Example 4



Formal Defn of a DFA:

$$M = \{ \underbrace{\Sigma = \{0,1\}}_{\text{alphabet}},\ \underbrace{Q = \{q_0, q_1\}}_{\text{set of states}},\ \underset{\text{start state}}{q_0},\ \underset{\text{accept states}}{F = \{q_1\}},\ \underbrace{\delta : Q \times \Sigma \to Q}_{\text{transition function}} \}$$

- A DFA accepts a string $w = w_1 w_2 \dots w_n$ over Σ if there exists a sequence of states $r_0 r_1 \dots r_n$, $r_i \in Q$ such that:

  (1) $r_0 = q_0$

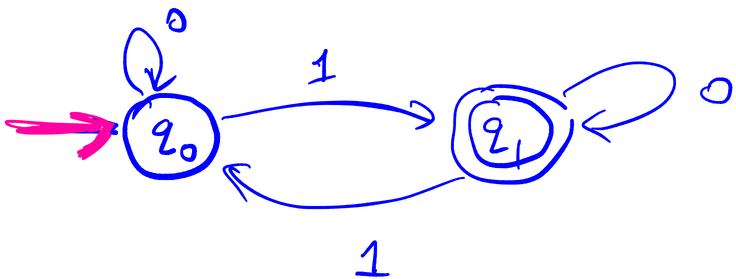
  (2) $\delta(r_i, w_{i+1}) = r_{i+1} \quad \forall i \in \{0, \dots, n-1\}$

  (3) $r_n \in F$

- The language accepted by a DFA M, denoted by $\mathcal{L}(M)$, is the set of all $w \in \Sigma^*$ that are accepted by M

- A language L is regular if some DFA accepts L

Example 1:



$w = 01101$ accepted
$w = 1100$ rejected
$w = 1110$

$\mathcal{L}(M) = \{ w \in \{0,1\}^* \mid w \text{ contains an odd number of } 1\text{'s} \}$

$w = 01011$ is accepted. Let

$$
\begin{array}{cccccc}
r_0 & r_1 & r_2 & r_3 & r_4 & r_5 \\
\shortparallel & \shortparallel & \shortparallel & \shortparallel & \shortparallel & \shortparallel \\
q_0 & q_0 & q_1 & q_1 & q_0 & q_1
\end{array}
$$

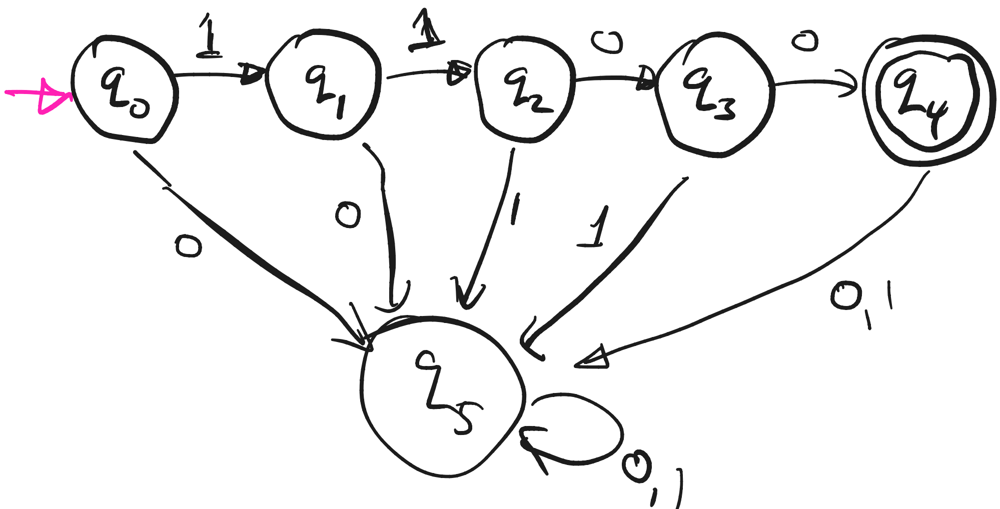
$$q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{0} q_1 \xrightarrow{1} q_0 \xrightarrow{1} q_1$$

Then conditions (1)-(3) are satisfied

More Examples

① $L \subseteq \{0,1\}^*$ such that $|L| = 1$

For example $L = \{1100\}$



Notation

For a string $w \in \Sigma^*$

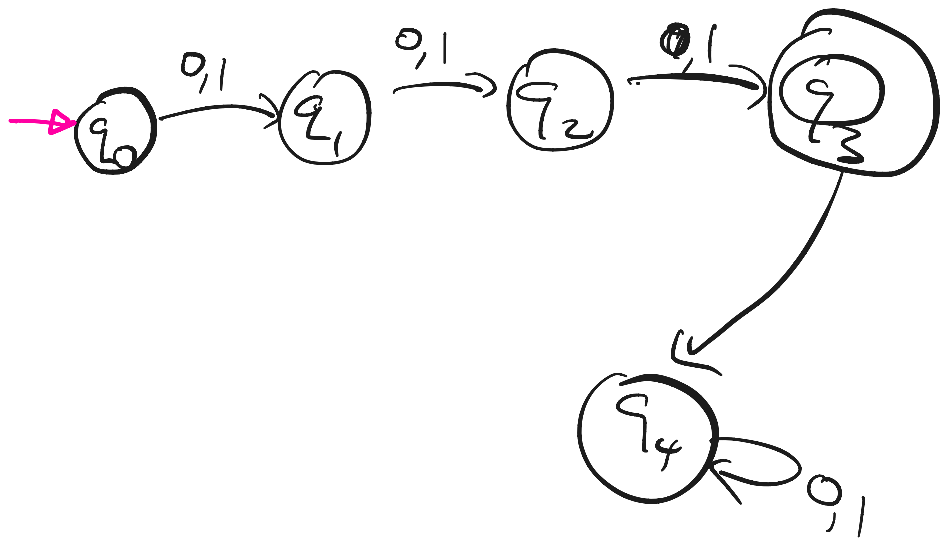
$|w|$ = # of symbols in w

For a set L

$|L|$ = number of elements in the set

= number of strings in L

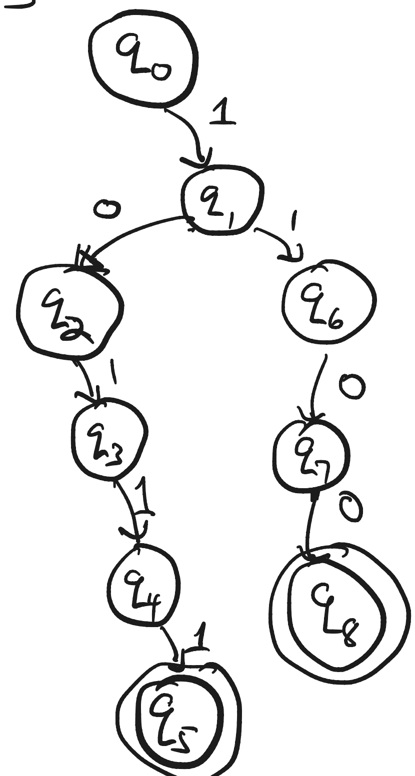
$$L = \{ w \in \{0,1\}^* \mid |w| = 3 \}$$



More Examples

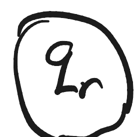
② $L = \{w_1, w_2, \ldots, w_k\}$ (L conists of exactly k strings over $\{0,1\}$)

Example $k=2$ $L = \{1100, 10111\}$

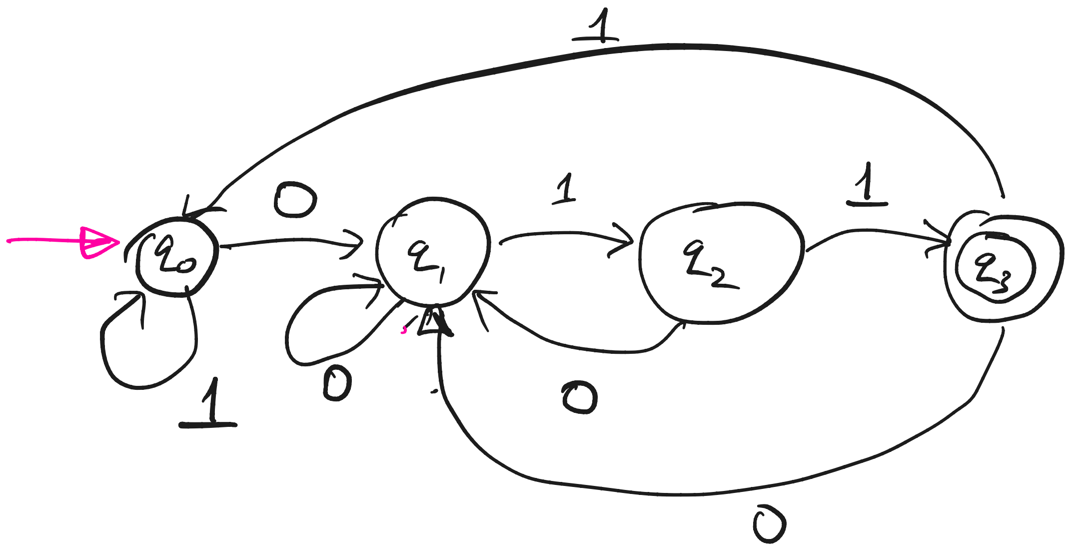


* All edges not drawn in diagram point to q_r

Example $\delta(q_2, 1) = q_r$



③ $L = \{w \in \{0,1\}^* \mid w \text{ ends in } 011\}$ ← L_2 from Lecture 1



q_3: last 3 symbols read are 011

q_2: last 2 symbols read are 01

q_1: last symbol read is 0

Nondeterministic Finite Automata (NFA's)

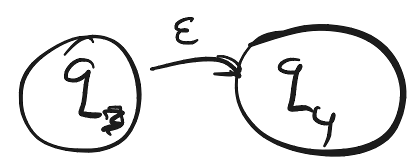
We will now define a more general type of finite automata

Key difference between DFA's & NFA's:

DFA: every state has exactly one outedge for every $a \in \Sigma$

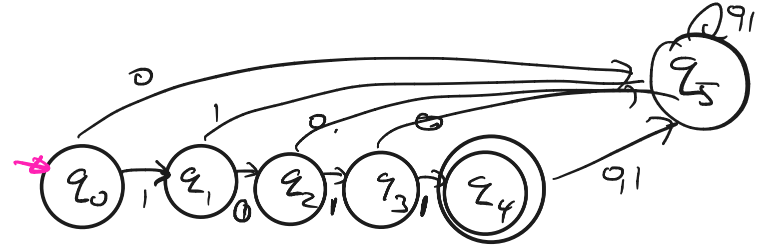
NFA: an NFA can have zero, one or more outedges labelled by $a \in \Sigma$

NFAs can also have edges labelled by ε which allow transitions from $q_i \to q_j$ without reading any input symbol

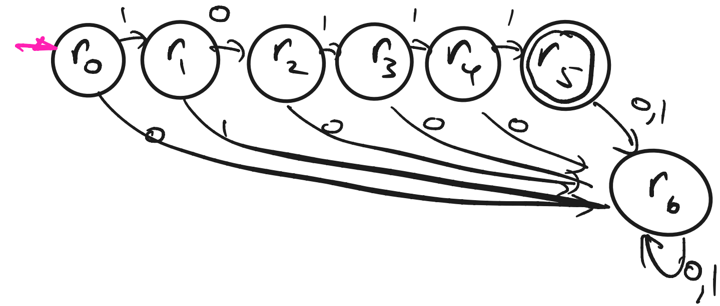


Example

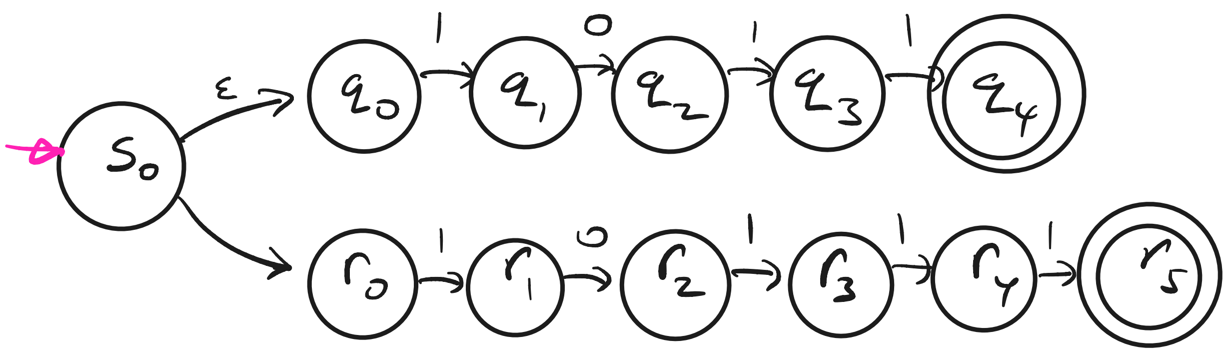
$L_1 = \{1011\}$ DFA for L_1:



$L_2 = \{10111\}$ DFA for L_2:

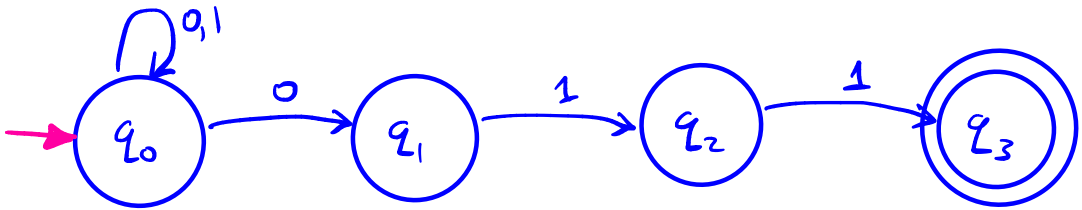


NFA for $L = L_1 \cup L_2 = \{1011, 10111\}$:



Nondeterministic Finite Automata (NFA's)

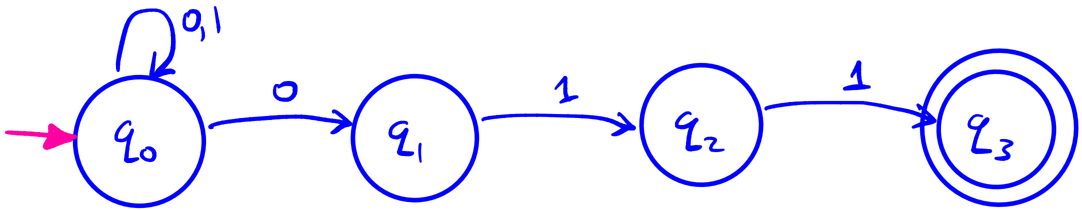
Example of an NFA M (over $\Sigma = \{0,1\}$):



$L = \{ w \in \{0,1\}^* \mid w \text{ ends in } 011 \}$

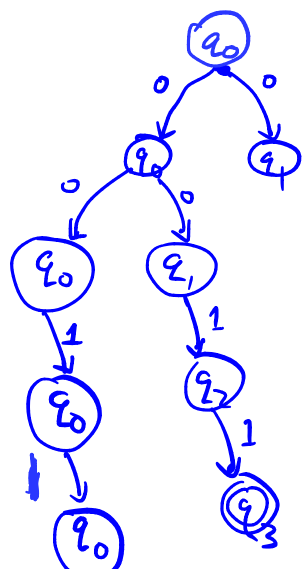
Nondeterministic Finite Automata (NFA's)

Example of an NFA M (over $\Sigma = \{0,1\}$):



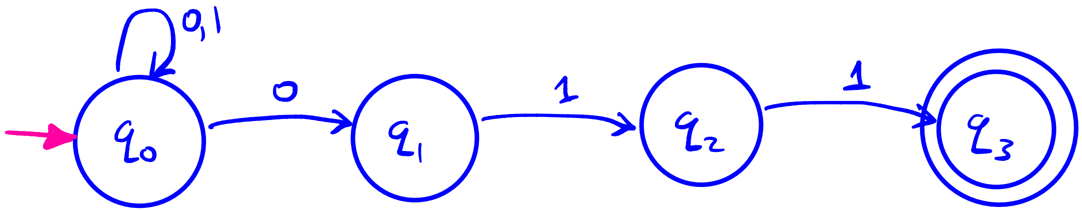
$w = 0011$

is accepted since there exists a sequence of transitions where w is accepted



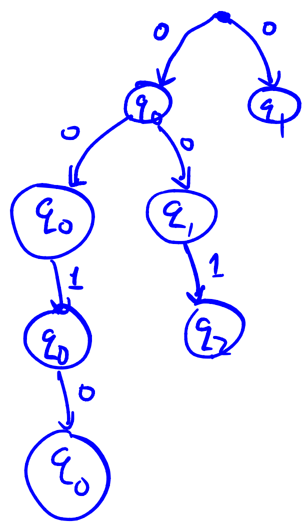
Nondeterministic Finite Automata (NFA's)

Example of an NFA M (over $\Sigma = \{0,1\}$):



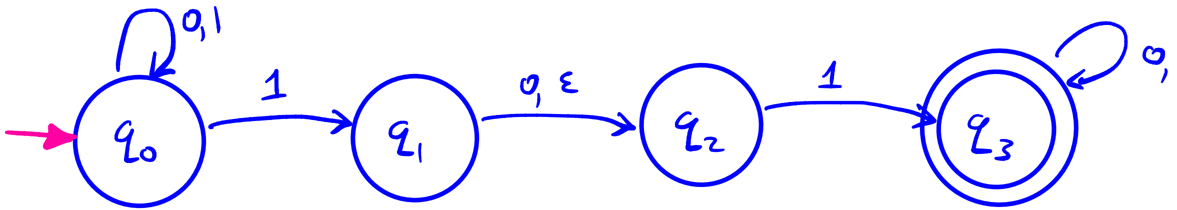
$w = 0010$

is rejected



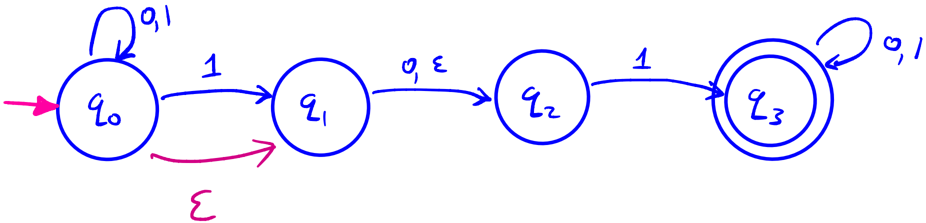
Nondeterministic Finite Automata (NFA's)

Example 2 :



Nondeterministic Finite Automata (NFA's)

Example 2 :



$w = 01011$: