Unit 2

Finite Automata

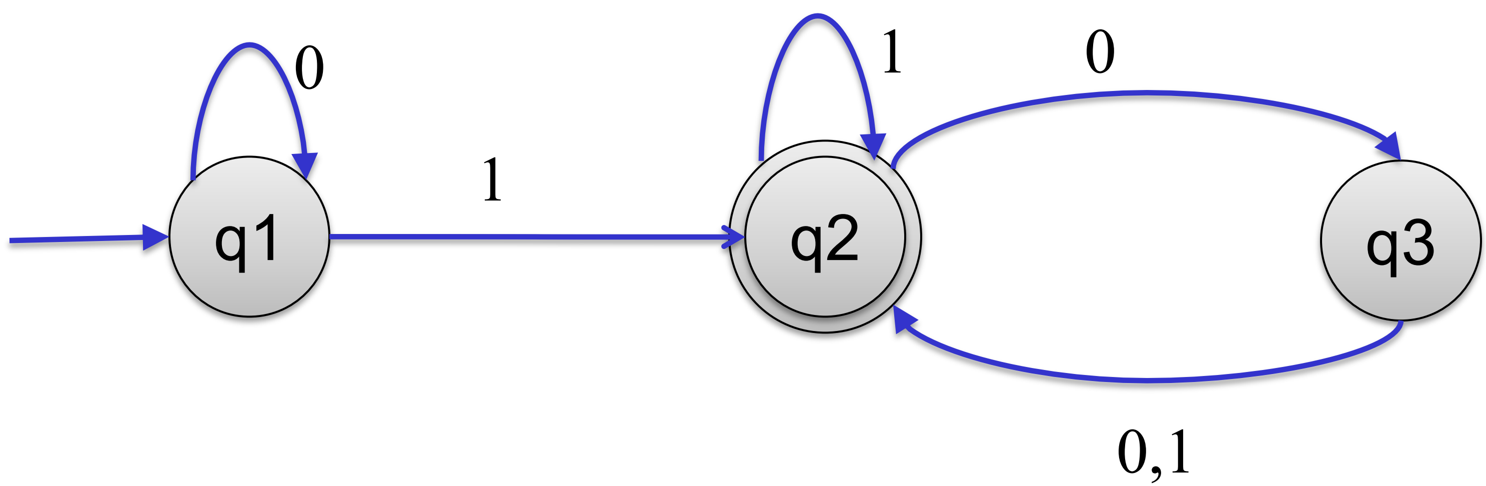
Regular Languages

Reading: Sipser, chapter 1

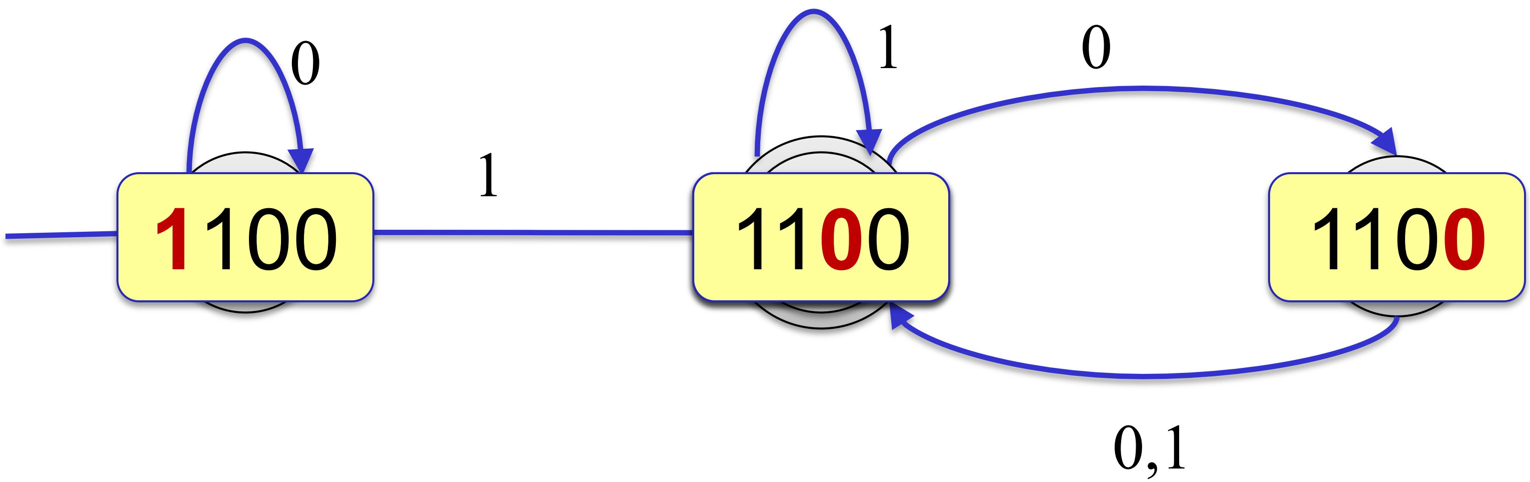
Please Welcome

our first computation model:

Finite Automata


0
1
q1
1
0
q2
q3
0,1

1100



The machine accepts a string if the process ends in a double circled state

Finite Automata (FA)

- Our first formal model of computation.
- Consists of *states* (nodes) and *moves* (edges).
- The *transition* between states is according to an input word. Each symbol dictates one move.
- Some states are ‘good’ (*accepting states*) and some are ‘bad’ (rejecting states).
- A word is accepted by the Automaton if the transitions it dictates end up in an accepting state.

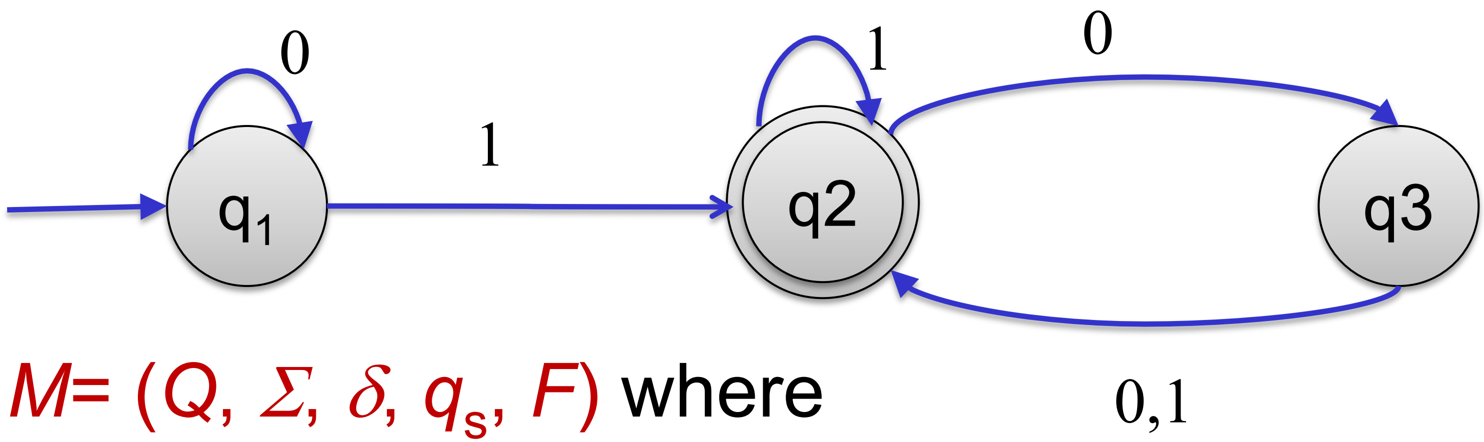
Formal Definition

A *Finite Automaton* (FA) is a 5-tuple $(Q, \Sigma, \delta, q_s, F)$:

- Q - a finite set called the *states.*
- Σ - a finite set called the *alphabet.*
- $\delta: Q\times\Sigma\rightarrow Q$ is the *transition function* $\delta(q_i,\sigma_j) = q_k$
- $q_s\in Q$ is the *starting state.*
- $F\subseteq Q$ is the set of *accepting states.*

FA is *deterministic (and complete)* if for every $(q_i, \sigma_j) \in Q\times\Sigma$, $\delta(q_i, \sigma_j)$ is uniquely defined.

Example:



- $M = (Q, \Sigma, \delta, q_s, F)$ where
  - $Q=\{q_1, q_2, q_3\}$
  - $\Sigma=\{0, 1\}$
  - $q_s=q_1$ is the starting state
  - $F=\{q_2\}$
  - δ is defined in the following table:

	0	**1**
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_2	q_2

Sometimes δ is defined as a set of 3-tupples where $(q_i, \sigma, q_j) \in \delta$ *means* $\delta(q_i, \sigma) = q_j$

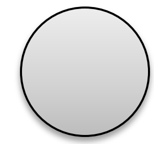
State Diagram

דיאגרמת מצבים

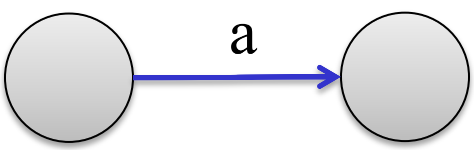
- A graphical representation of a finite automaton FA is also called a ***state diagram***.
- Given a formal definition of the FA one can draw its state diagram.
- Given a state diagram one can write a formal definition of the FA.

State Diagram

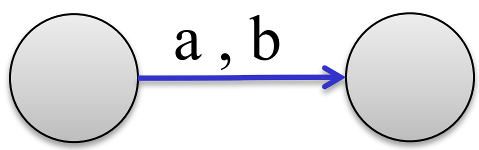
- A circle represents a *state* of the automaton.



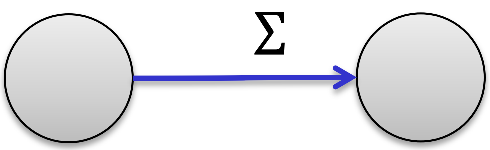
- The *transition function* is represented by directed and labeled edges between states.



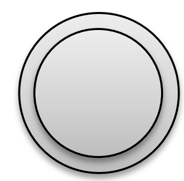
or



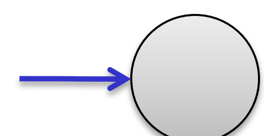
or



- *Accepting states* have a double circle.



- The *starting state* has an incoming arrow.



How does an Automaton work?

- Reads one symbol of the input word (i.e. one letter) in each time slot (left to right).
- In each time slot, it moves to a new state according to the current state and the symbol it reads (defined by the transition function).
- The automaton stops after the last move dictated by the last symbol in the input.
- If the state in which it stoped is accepting - the automaton ***accepts*** the input word.

Formal Definition of Computation

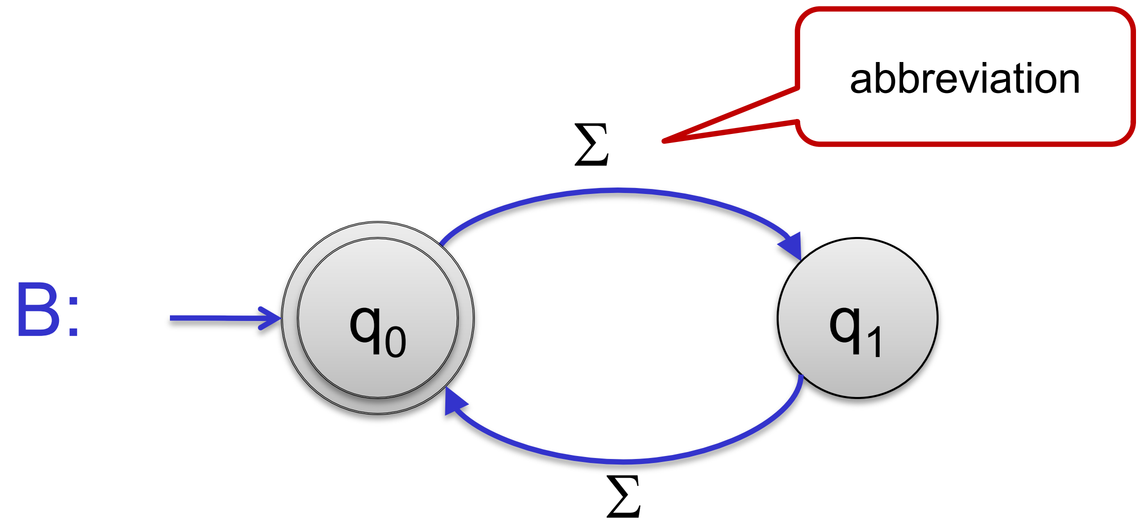
A finite automaton $M = (Q,\Sigma,\delta,q_s,F)$ ***accepts*** a string/word $\mathbf{w} = \sigma_1\ldots\sigma_n$, $\sigma_i \in \Sigma$ if and only if there is a sequence $r_0\ldots r_n$ of states $r_i \in Q$, such that:

1) $r_0 = q_s$
2) $\delta(r_i,\sigma_{i+1}) = r_{i+1}$ for all $i = 0,\ldots,n-1$
3) $r_n \in F$

The Language of an Automaton

- The *language* of an automaton *M* is denoted *L(M)*
- *L(M)* consists of ***all and only words*** that *M* accepts, i.e. when reading them it stops in an accepting state.

Recognizing the Language of an Automaton

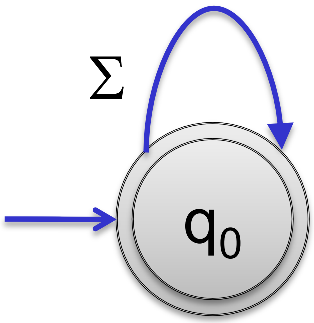


The language of B:

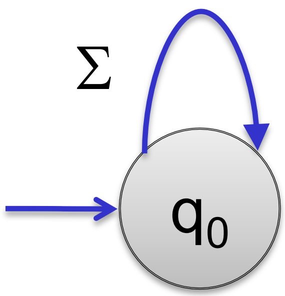
$$L(B) = \{ w \in \Sigma^* \mid |w| \text{ is even} \}$$

More Examples

- $L=\Sigma^*$

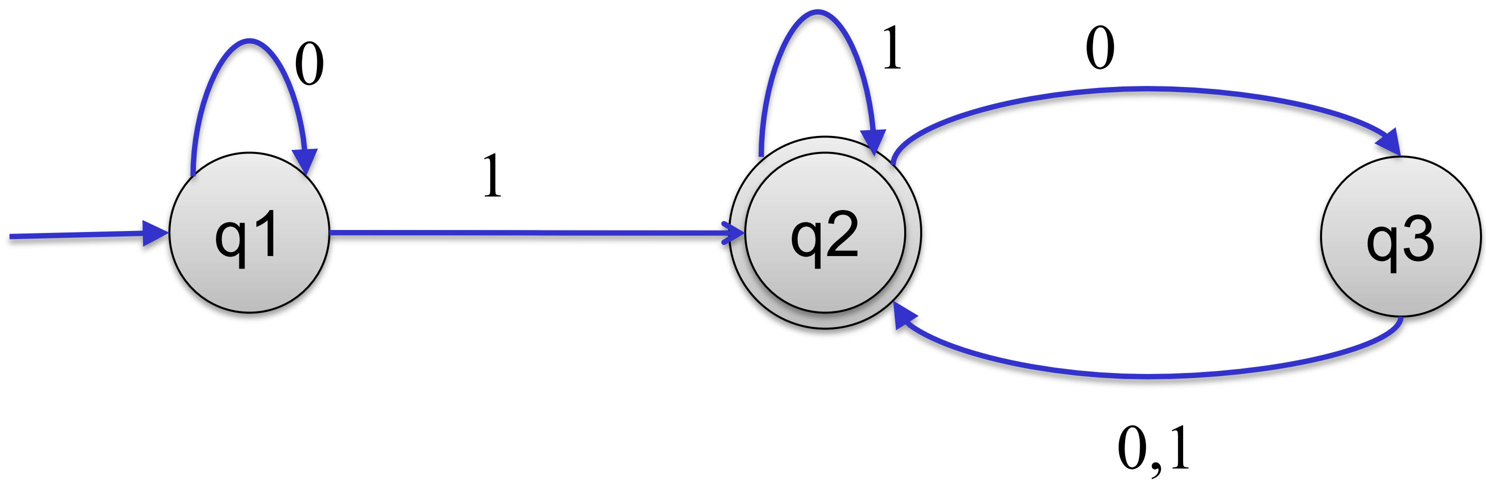


- $L=\varnothing$



Previous Example

M:



The language of *M*:

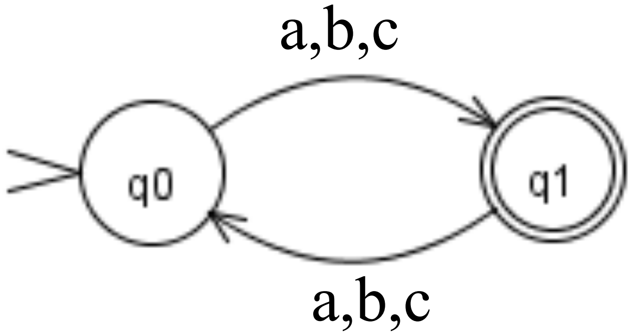
$L(M) = \{w \in \{0,1\}^* \mid w$ contains at least one 1 and even number of 0 follows the last 1$\}$

Designing an Automaton

Construct an automaton B accepting the following language:

L(B)= $\{w \in \{a, b, c\}^* \mid |w| \text{ is odd}\}$

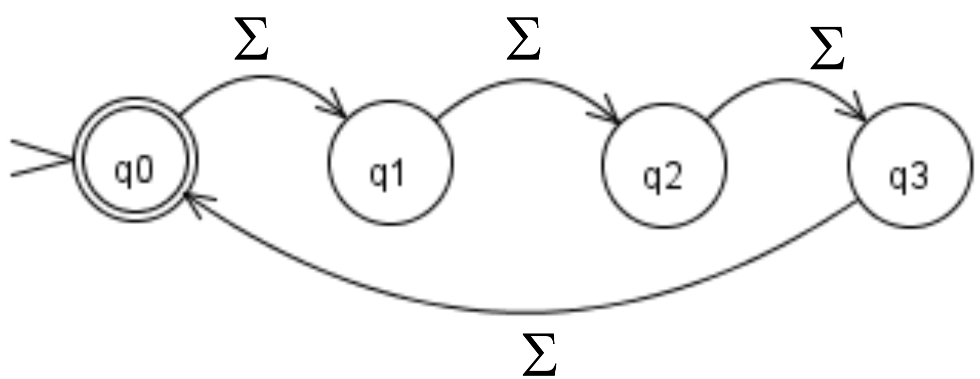
B:



Another Example

- Construct an automaton accepting the following language:

$$L = \{w \in \{0,1\}^* \mid |w| \; mod \; 4 = 0\}$$

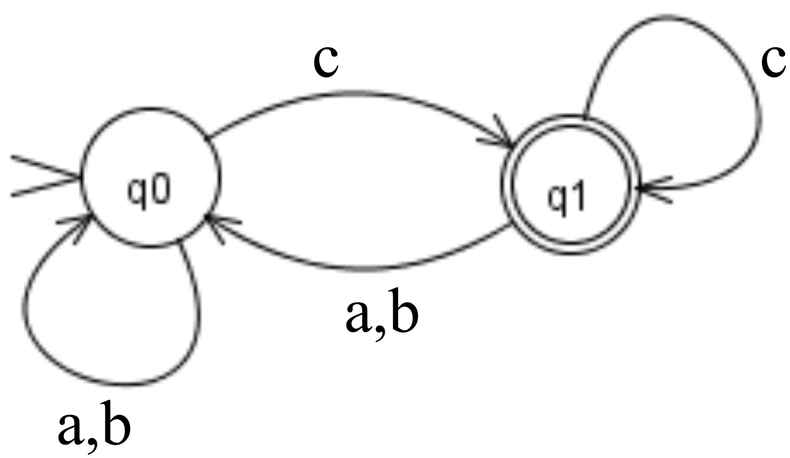


Another Example

- Construct an automaton C accepting the following language:

$$L(C)=\{\, w \mid w = uc,\ u \in \{a,b,c\}^* \}$$

C:



Exercise

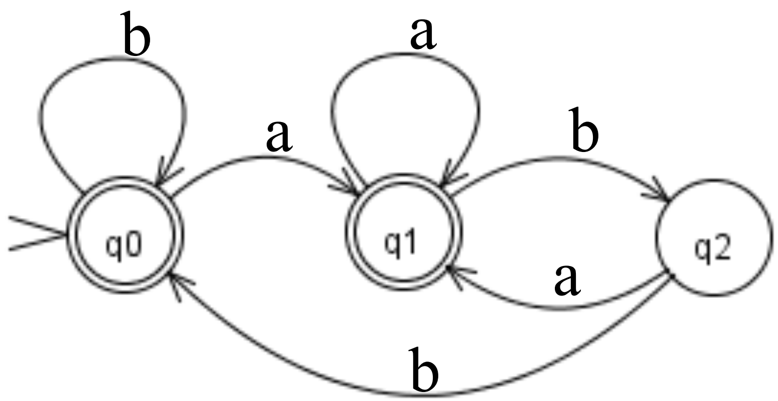
Construct an automaton accepting the language

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

Answer: in class

From state diagram to a formal description

- Consider the finite-state automaton ***A*** defined by the state diagram shown below:



$A = (Q_A, \Sigma_A, \delta_A, q_0, F_A)$ where

$Q_A=\{q_0,q_1,q_2\}$, $\Sigma_A=\{a,b\}$,

δ_A – see table

$F_A=\{q_0,q_1\}$,

What is L(A)? What is ~L(A)

δ_A	a	b
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_0

From formal description to a state diagram

- Draw a state diagram according to $B=(Q, \Sigma, \delta, q_0, F)$.

$Q=\{q_0,q_1,q_2,q_3\}$,

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

q_0 initial state,

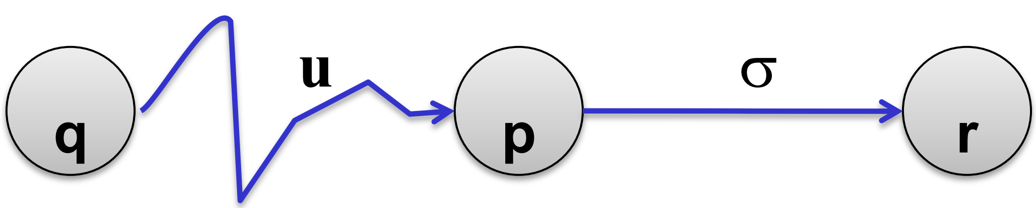
$F=\{q_1\}$.

δ	0	1
q_0	q_1	q_3
q_1	q_1	q_2
q_2	q_1	q_2
q_3	q_3	q_3

- Answer: In class

Extended transition function

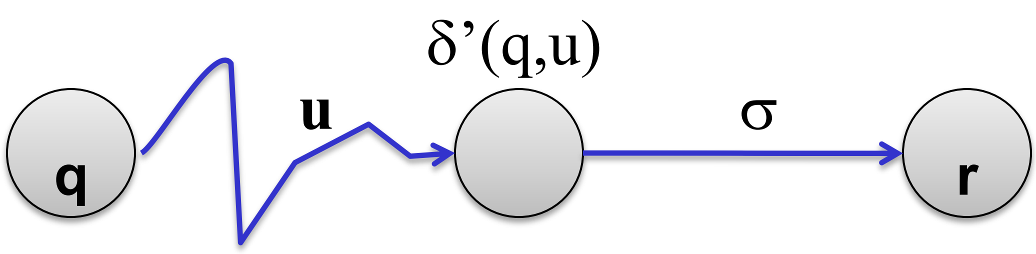
- Let $\delta: Q \times \Sigma \rightarrow Q$ be a transition function
- We define an *extended transition function* $\delta': Q \times \Sigma^* \rightarrow Q$
- The extended transition function δ' defines the movement of an automaton on words.
- Let $w = u\sigma$ (w,u- words, σ- symbol)
- $\delta'(q,w) = \delta'(q,u\sigma) = \delta(\delta'(q,u), \sigma) = \delta(p, \sigma) = r$



Extended transition function

Formal definition $\delta': Q \times \Sigma^* \rightarrow Q$

- $\delta'(q,\varepsilon)=q$, for all $q \in Q$,
- $\delta'(q,u\sigma)= \delta\ (\delta'(q,u),\ \sigma)$



The Language of an Automaton

- **Informally**: *L*(*A*) is a set of the words and only the words that *A* accepts.

- **Formally**:

Let $A=(Q,\Sigma,\delta,q_0,F)$,

$L(A)=\{w\in\Sigma^* \mid \delta'(q_0,w)\in F\}$

- We say that A ***recognizes*** *L*(*A*)

Regular Languages

A language is called a *regular language* iff some finite automaton recognizes it.

FA Questions

- Given a language - is it regular?
- In general, what kind of languages can be recognized by FA? (What are the regular languages?)
- What kind of languages a FA cannot recognize?

Exercises

Give DFA state diagrams for the following
languages over Σ={0,1}:

1. $\{ w \in \Sigma^* \mid$ w contains substring 110}
2. $\{w \in \Sigma^* \mid$ w does not contain substring 110}
3. $\{w \in \Sigma^* \mid$ w contains 00 but doesn't contain 11}
4. $\{w \in \{0,1,2\}^* \mid$ w is a number is basis 3, whose value is divisible by 2}

The language of an automaton

In-class exercise:

1. Describe formally and graphically an automaton, A, for the language

$$L=\{w\in\{0,1\}^* \mid \#_1(w) \text{ is even}\}$$

2. Prove that $L(A)=L$.