CSE-233 : Week 1

Summer 2020

Introduction to Finite Automata

Reference:

Book1 Chapter 1.5,

Book2 Chapter 1.1

Md. Saidul Hoque Anik

anik@cse.uiu.ac.bd

Key Concepts

1. Alphabet Σ
2. String, w
3. Language, L

Alphabet

Alphabet of English Language $\Sigma_{\text{Eng}} = \{a, b, c, \ldots\}$

Alphabet of Bengali Language $\Sigma_{\text{Beng}} =$ {অ, আ, ক,...}

Alphabet of Arabic Language $\Sigma_{\text{Arabic}} =$ {ج, ب,أ...}

Strings

Alphabet of English Language $\Sigma_{Eng} = \{a, b, c, \ldots\}$

Strings $= \{\varepsilon$, a, bd, cat, pq, ...$\}$

Alphabet of Bengali Language $\Sigma_{Beng} = \{$অ, আ, ক,...$\}$

Strings = $\{\varepsilon$, কখ, আ, গাড়ি,...$\}$

Empty String is ε

Do you remember what is Σ^* ?

What is $\Sigma_{eng}{}^*$? What is $\Sigma_{Beng}{}^*$?

Language

Are all strings meaningful?

Alphabet of English Language $\Sigma_{Eng} = \{a, b, c, \ldots\}$

Strings = {a, bd, cat, pq, ...}

Language = set of strings from Σ^* that are meaningful

= {cat, dog, car, ... }

Mapping Problems as Language

Languages can be used to describe problems with "yes/no" answers, for example:

L_1 =	The set of all strings over S_1 that contain the substring "food"
L_2 =	The set of all strings over S_2 that are divisible by 7
=	{7, 14, 21, …}
L_3 =	The set of all strings of the form w#w where w is any string over {a, b, …, z}
L_4 =	The set of all strings over S_4 where every (can be matched with a subsequent)

More Examples of Language

1. The language of all strings consisting of n 0s followed by n 1s ($n \geq 0$):

$$\{\epsilon, 01, 0011, 000111, \ldots\}$$

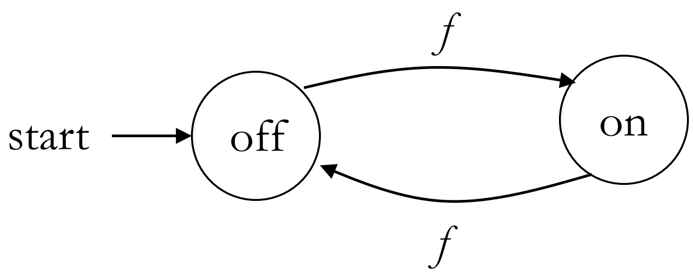
2. The set of strings of 0s and 1s with an equal number of each:

$$\{\epsilon, 01, 10, 0011, 0101, 1001, \ldots\}$$

3. Σ^* is a language for any alphabet Σ
4. $\emptyset$, the empty language, is a language over any alphabet
5. $\{\epsilon\}$, the language consisting of only the empty string, is also a language over any alphabet
   NOTE: $\emptyset \neq \{\epsilon\}$ since $\emptyset$ has no strings and $\{\epsilon\}$ has one
6. $\{w \mid w$ consists of an equal number of 0 and $1\}$
7. $\{0^n 1^n \mid n \geq 1\}$
8. $\{0^i 1^j \mid 0 \leq i \leq j\}$

Finite Automata

Small program that can recognize language (Meaningful Inputs)

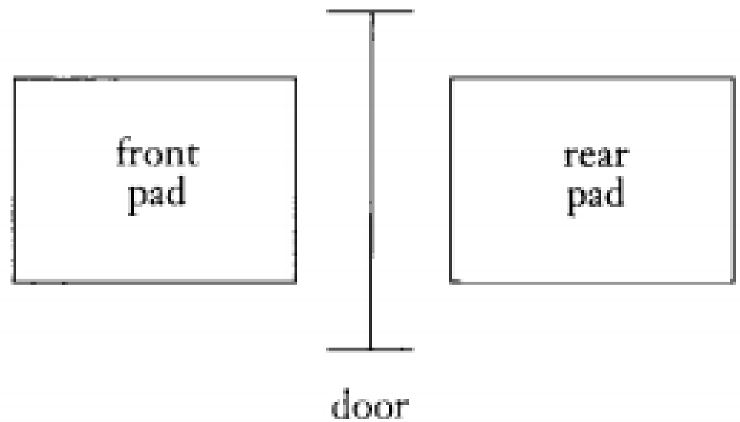


bulb is on if and only if
there was an odd number
of flips

Deterministic Finite Automata

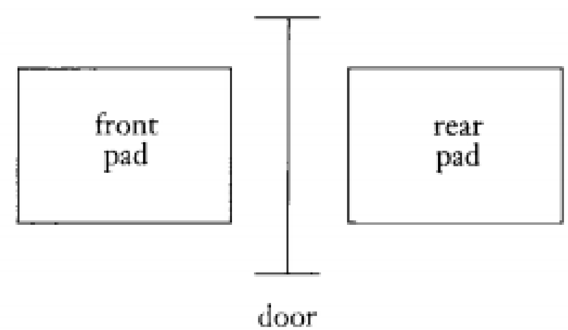
Small program that can recognize language (Meaningful Inputs) and takes one step at a time

Example: Shopping Mall Glass Door



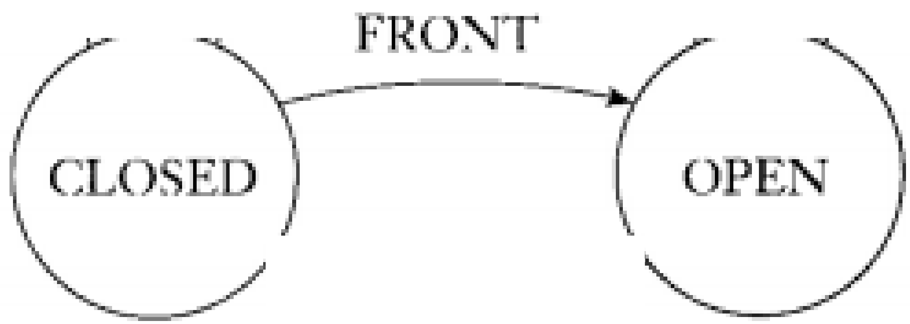
DFA Example

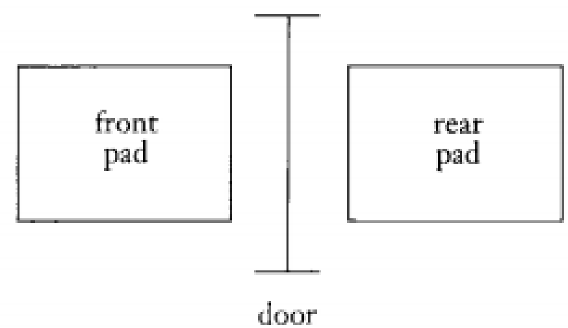
Shopping Mall Glass Door



DFA Example

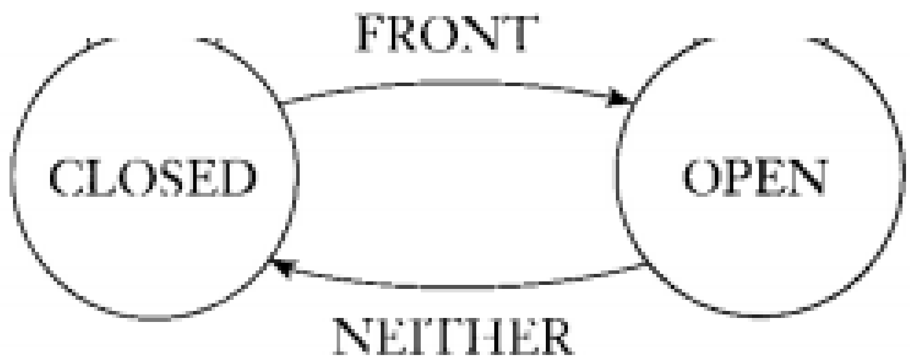
Shopping Mall Glass Door

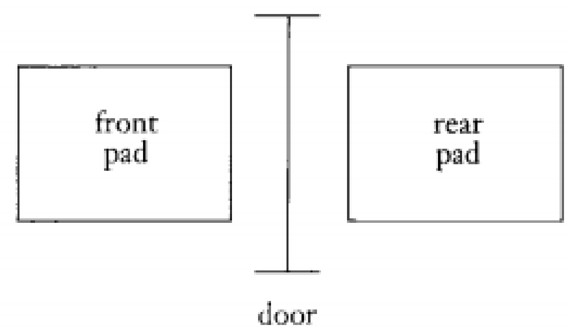




DFA Example

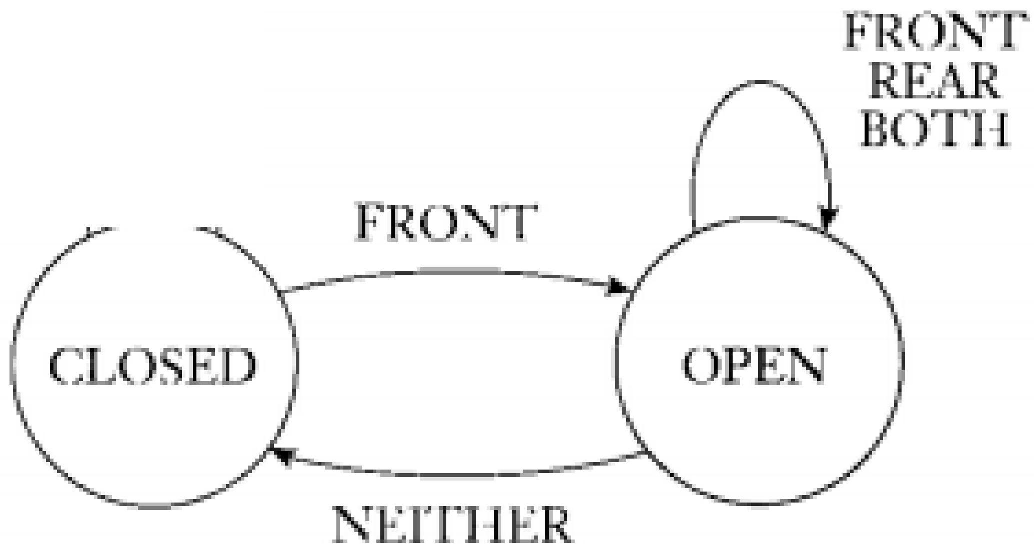
Shopping Mall Glass Door

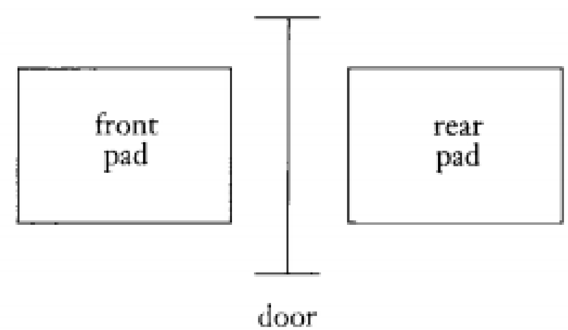




DFA Example

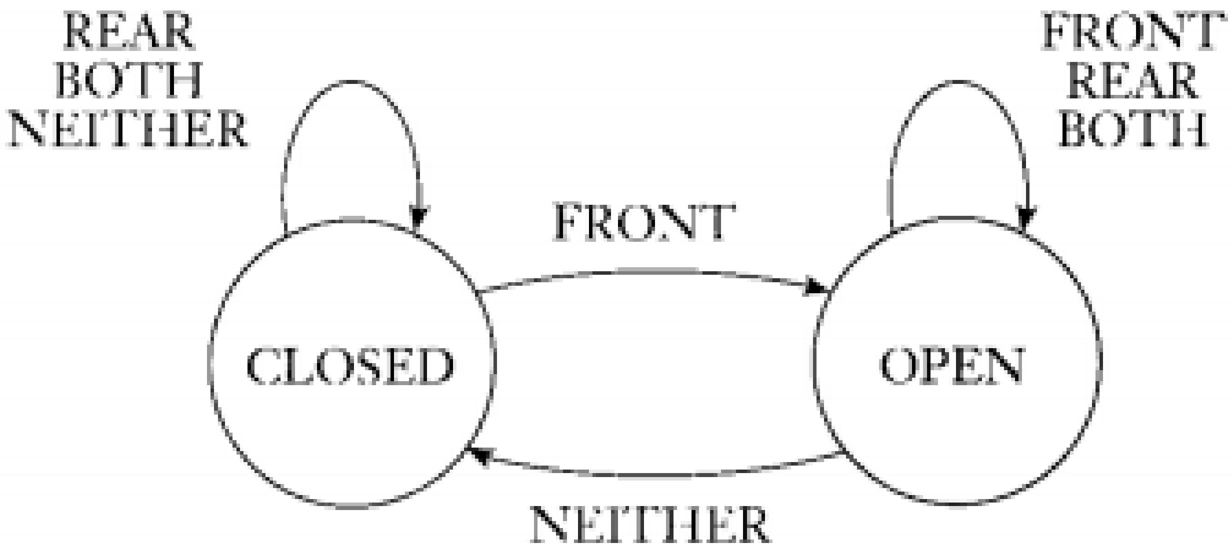
Shopping Mall Glass Door

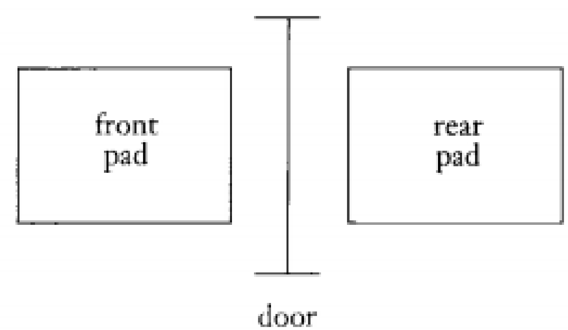




DFA Example

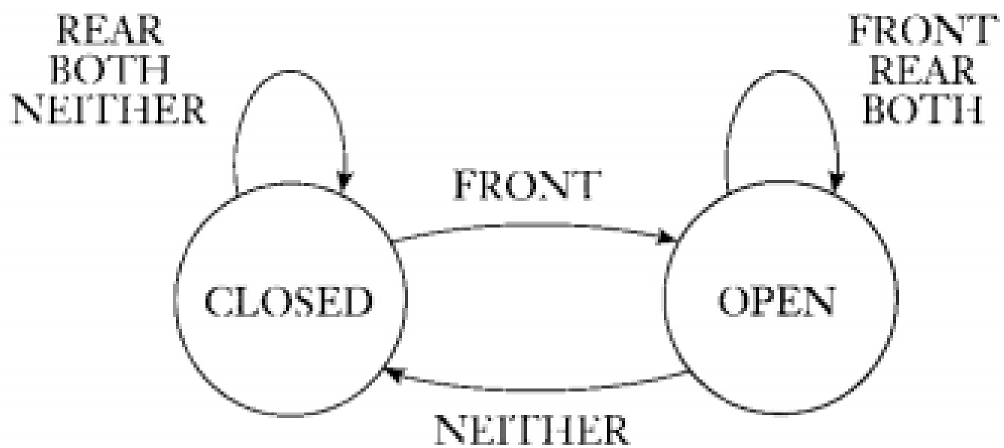
Shopping Mall Glass Door





Transition Function

Shopping Mall Glass Door



input signal

		NEITHER	FRONT	REAR	BOTH
state	CLOSED	CLOSED	OPEN	CLOSED	CLOSED
	OPEN	CLOSED	OPEN	OPEN	OPEN

Another Example

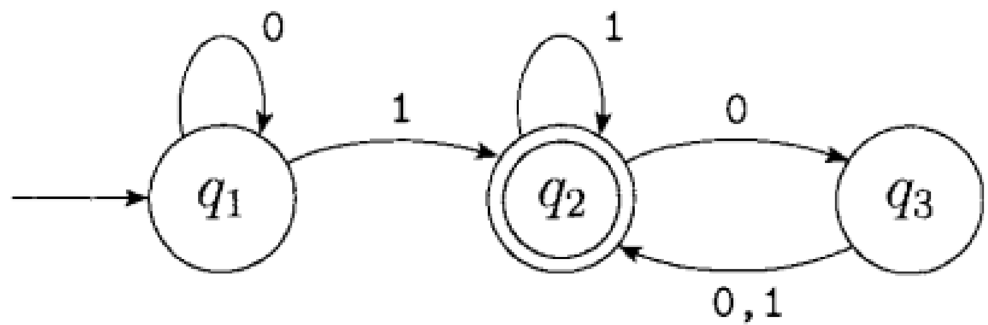


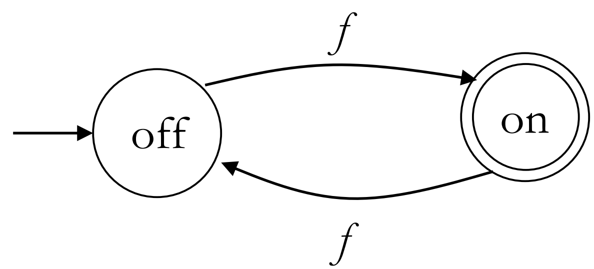
FIGURE 1.4
A finite automaton called M_1 that has three states

1. States = ?
2. Accepted Inputs (Alphabets)?
3. Initial/Starting State = ?
4. Transition Function = ?
5. Accept State = ?

Language of DFA

All inputs that are accepted by the FDA

M:



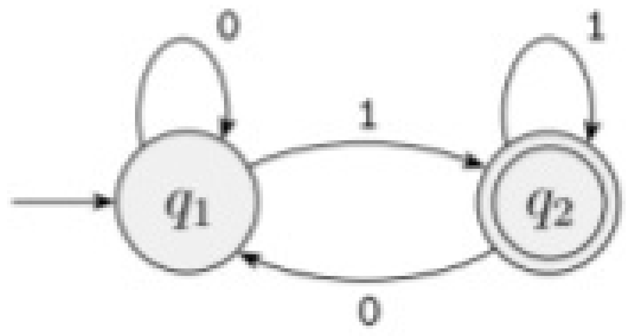
- Language of M is $\{f, fff, fffff, \ldots\} = \{f^n\text{: } n \text{ is odd}\}$

Formal Definition of DFA

A ***finite automaton*** is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where

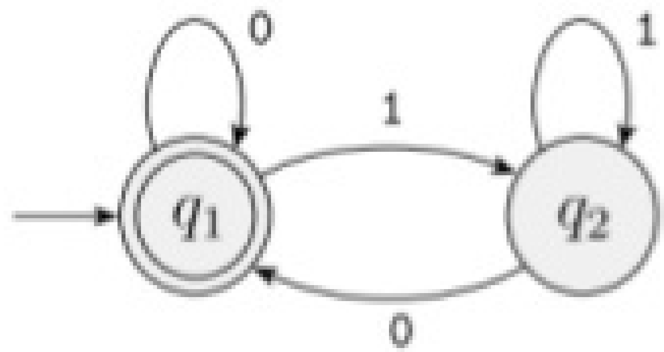
1. Q is a finite set called the ***states***,
2. Σ is a finite set called the ***alphabet***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ is the ***transition function***,[1]
4. $q_0 \in Q$ is the ***start state***, and
5. $F \subseteq Q$ is the ***set of accept states***.[2]

More Examples



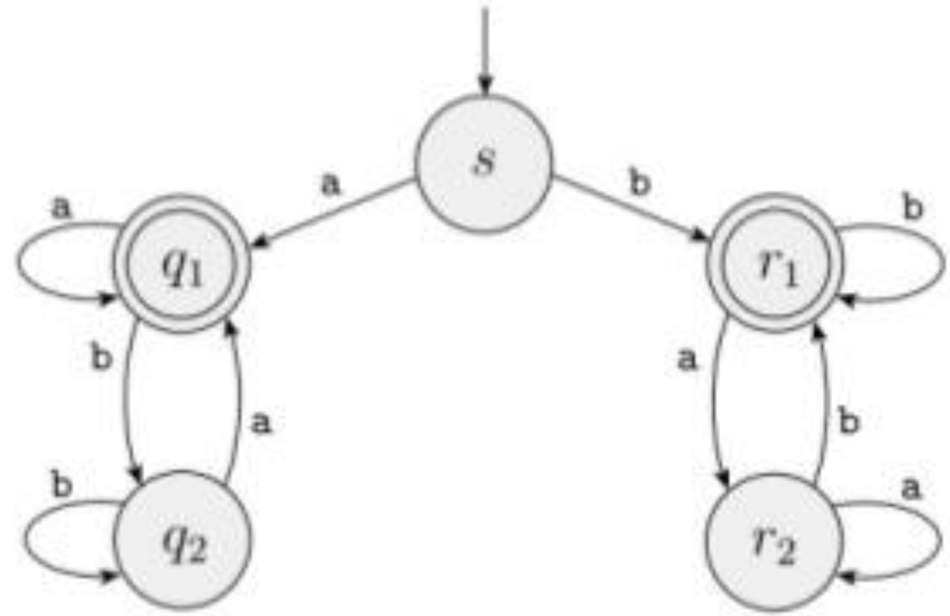
1. States = ?
2. Accepted Inputs (Alphabets)?
3. Initial/Starting State = ?
4. Transition Function = ?
5. Final State = ?
6. Language = ?

More Examples



1. States = ?
2. Accepted Inputs (Alphabets)?
3. Initial/Starting State = ?
4. Transition Function = ?
5. Final State = ?
6. Language = ?

More Examples



1. States = ?
2. Accepted Inputs (Alphabets)?
3. Initial/Starting State = ?
4. Transition Function = ?
5. Final State = ?
6. Language = ?