Deterministic Finite Automata

A (Real-Life?) Example

- Example: Kitchen scale
- P = Power button (ON / OFF)
- U = Units button (cycles through g / oz / lb)

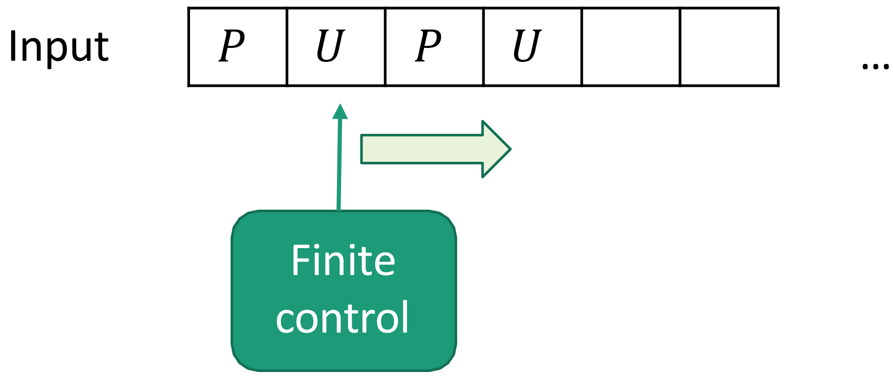
  Only works when scale is ON, but units remembered when scale is OFF

- Starts OFF in g mode

- A computational problem: Does a sequence of button presses in $\{P,U\}^*$ leave the scale ON in oz mode?

Machine Models

- Finite Automata (FAs): Machine with a finite amount of unstructured memory



Control scans left-to-right

A DFA for the Kitchen Scale Problem

A DFA Recognizing Parity

The language recognized by a DFA is the set of inputs on which it ends in an "accept" state

Parity: Given a string consisting of a's and b's, does it contain an even number of a's?

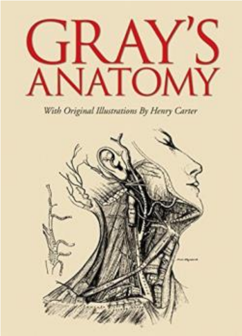
$\Sigma = \{a,b\}$ $L = \{w \mid w$ contains an even number of a's$\}$

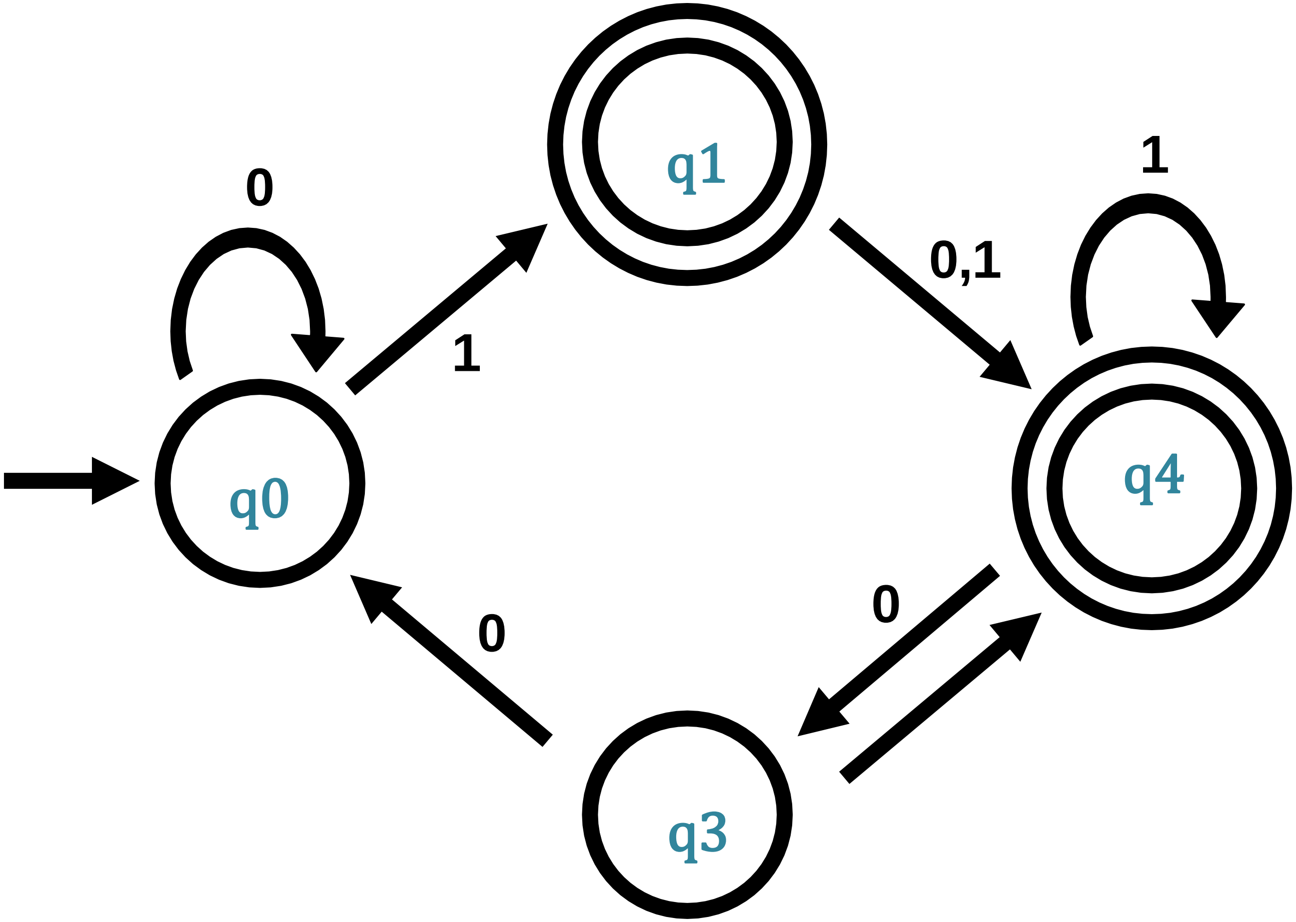
Which state is reached by the parity DFA on input aabab?

a) "even"

b) "odd"

Anatomy of a DFA





Some Tips for Thinking about DFAs

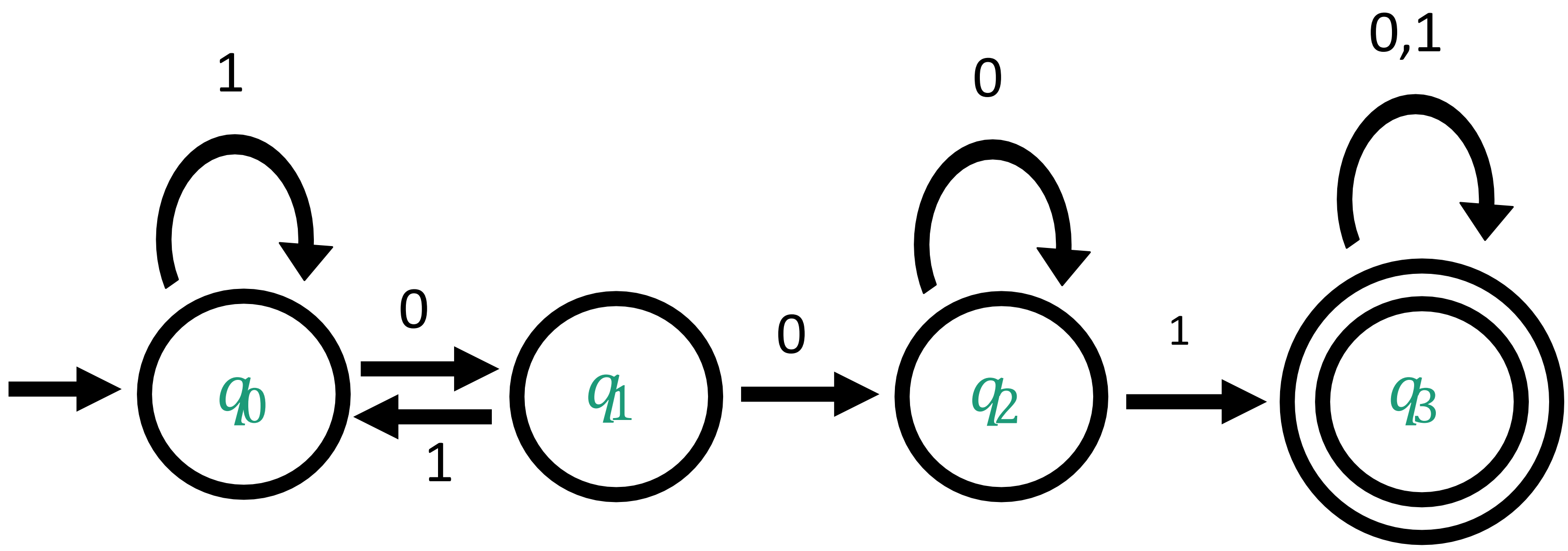
Given a DFA, what language does it recognize?

- Try experimenting with it on short strings. Do you notice any patterns?
- What kinds of inputs cause the DFA to get trapped in a state?

Given a language, construct a DFA recognizing it

- Imagine you are a machine, reading one symbol at a time, always prepared with an answer
- What is the essential information that you need to remember? Determines set of states.

What language does this DFA recognize?



Practice!

- Lots of worked out examples in Sipser
- Automata Tutor: https://automata-tutor.model.in.tum.de/

Formal Definition of a DFA

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

Q is the set of states

Σ is the alphabet

$\delta : Q \times \Sigma \rightarrow Q$ is the transition function

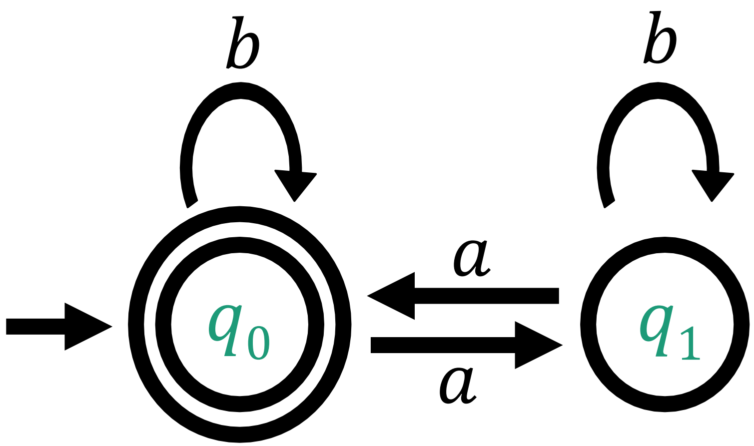
$q_0 \in Q$ is the start state

$F \subseteq Q$ is the set of accept states

A DFA for Parity

Parity: Given a string consisting of a's and b's, does it contain an even number of a's?

$\Sigma = \{a, b\}$ $L = \{w \mid w \text{ contains an even number of } a\text{'s}\}$



State set Q =

Alphabet Σ =

Transition function δ

δ	a	b
q_0		
q_1		

Start state q_0

Set of accept states F =

Formal Definition of DFA Computation

A DFA $M = (Q, \Sigma, \delta, q_0, F)$ accepts a string $w = w_1 w_2 \cdots w_n \in \Sigma^*$ (where each $w_i \in \Sigma$) if there exist $r_0, \ldots, r_n \in Q$ such that

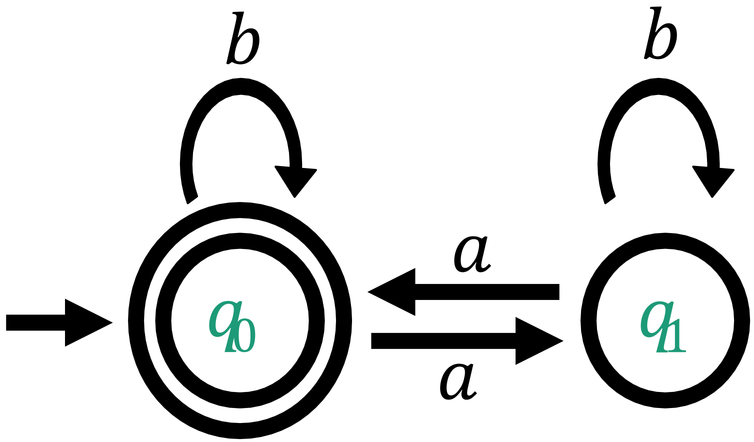
1. $r_0 = q_0$
2. $\delta(r_i, w_{i+1}) = r_{i+1}$ for each $i = 0, \ldots, n-1$, and
3. $r_n \in F$

$L(M)$ = the language of machine M

= set of all strings machine M accepts

M recognizes the language $L(M)$

Example: Computing with the Parity DFA



Let $w = \mathrm{abba}$

Does M accept w?

What is $\delta(r2, w3)$?

a) $q0$

b) $q1$

A DFA $M = (Q, \Sigma, \delta, q_0, F)$ accepts a string $w = w_1, w_2 \cdots w_n \in \Sigma^*$ (where each $w_i \in \Sigma$) if there exist $r_0, \ldots, r_n \in Q$ such that

1. $r_0 = q_0$
2. $\delta(r_i, w_{i+1}) = r_{i+1}$ for each $i = 0, \ldots, n-1$, and
3. $r_n \in F$

Regular Languages

Definition: **A language is regular if it is recognized by a DFA**

$L = \{ w \in \{a,b\}^* \mid w$ has an even number of as$\}$ is regular

$L = \{ w \in \{0,1\}^* \mid$ w contains 001 $\}$ is regular

Many interesting problems are captured by regular languages

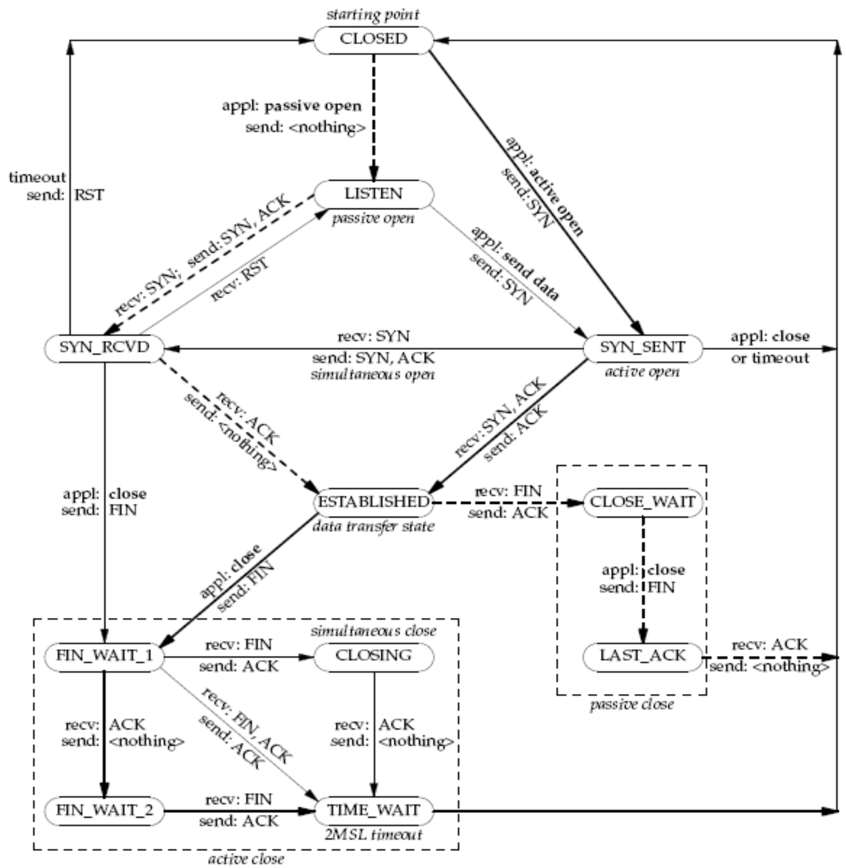
NETWORK PROTOCOLS

COMPILERS

GENETIC TESTING

ARITHMETIC

Internet Transmission Control Protocol



Let TCPS = { w| w is a complete TCP Session}

Theorem. TCPS is regular

Compilers

Comments :

Are delimited by /* */

Cannot have nested /* */

Must be closed by */

*/ is illegal outside a comment

COMMENTS = {strings over {0,1, /, *} with legal comments}

Theorem. COMMENTS is regular.

Genetic Testing

DNA sequences are strings over the alphabet $\{A, C, G, T\}$.

A **gene** g is a special substring over this alphabet.

A **genetic test** searches a DNA sequence for a gene.

GENETICTEST$_g$ = {strings over $\{A, C, G, T\}$ containing g as a substring}

Theorem. GENETICTEST$_g$ is regular for every gene g.

Arithmetic

LET $\Sigma = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}$

- A string over Σ has three ROWS (ROW$_1$, ROW$_2$, ROW$_3$)
- Each ROW $b_0 b_1 b_2 \ldots b_N$ represents the integer

$$b_0 + 2b_1 + \ldots + 2^N b_N.$$

- Let ADD = $\{S \in \Sigma^* \mid$ ROW$_1$ + ROW$_2$ = ROW$_3$ $\}$

Theorem. ADD is regular.

Readings

Sipser Ch 1.1-1.2