COSC 444/541: Automata

Lecture 3: Three Basic Concepts, DFAs

Today’s Agenda

- Review proof by induction
- Continue Three Basic Concepts (Chapter 1.2)
- Break at 1:50 PM
- Start Finite Automata:
  - DFA (Chapter 2.1)
- Exit Token

Review: Proof by Induction

Solution posted on Canvas:

[Textbook page 16, exercise 38]

Prove that for all $n \geq 4$ the inequality $2^n < n!$ holds.

Review: Proof that π is irrational

Out of scope of our course, but you can take a look at the multiple proofs:

Wiki: prove that the number π is irrational

Three Basic Concepts

Language

Grammar

Automata

What is a language?

Natural languages: English, French, …

Formal languages require precise definitions.

- Start with an alphabet Σ = finite, nonempty set of symbols
- Build strings that are finite sequences of symbols from Σ
- A language = any subset of Σ^*

Strings

Example: $\Sigma = \{a, b\}$

Strings: abab, aaabbba, λ (empty string)

Notion of strings: u, v, w

Operations:

- Concatenation: uv
- Reverse: w^R
- Length: $|w|$
- Empty string: λ, with $|\lambda|=0$
- w^n denotes the string obtained by repeating w n times
- The length of concatenation of strings is the sum of individual lengths (theorem 1.6 in textbook)
  - $|uv| = |u|+|v|$

Strings

Σ^* = all strings (including λ) from Σ

Languages = subsets of Σ^*

Standard set operations on languages: Union, intersection, difference, complement

Reverse: $L^R = \{w^R : w \in L\}$
Concatenation: $L_1L_2 = \{xy : x \in L_1, y \in L_2\}$
Complement: $\overline{L}$ or L^c

Strings

L^n is defined as L concatenated with itself n times.

Special cases:

- $L^0=\{\lambda\}$
- $L^1=LU$

Star-closure of L:

$$L^* = L^0 \cup L^1 \cup L^2 \ldots$$

If $L^1=\{a^nb^n: n\geq 0\}$, then $L^2=\{a^nb^na^mb^m: n\geq 0, m\geq 0\}$

Strings

For example, if L = {ab, baa}, so tokens are of lengths 2 and 3.

L^3 consists of concatenations of exactly 3 tokens.

Possible lengths: 2+2+2=6, 2+2+3=7, 2+3+3=8, 3+3+3=9

L^4 consists of concatenations of exactly 4 tokens.

Possible lengths: 8,9,10,11,12

Grammar

Informal descriptions (English) are ambiguous.

Grammar is a precise way to define languages.

A grammar $G = (V, T, S, P)$:

- V = variables (nonterminals)
- T = terminals (alphabet symbols)
- $S \in V$ = start symbol
- P = productions

Automata

An automata is an abstract model of computation

Components:

- Input mechanism (reads symbols)
- Control unit
- Optional storage (stack, etc.)
- Transition function

Output:

Accepter (yes/no)
Transducer (produces output string)

Three Basic Concepts

Example 1:

[Textbook page 28, exercise 1]

1. How many substrings aab are in $ww^R w$, where $w = aabbab$?

Three Basic Concepts

Example 2:

[Textbook page 28, exercise 2]

2. Use induction on n to show that $|u^n| = n\,|u|$ for all strings u and all n.

Three Basic Concepts

Example 3:

[Textbook page 28, exercise 6]

6. Let $\Sigma = \{a, b\}$ and $L = \{aa, bb\}$. Use set notation to describe $\overline{L}$.

Three Basic Concepts

In-class Exercise:

[Textbook page 28, exercise 5]

5. Let $L = \{ab, aa, baa\}$. Which of the following strings are in L^*: $abaabaaabaa$, $aaaabaaaa$, $baaaaabaaaab$, $baaaaabaa$? Which strings are in L^4?

Finite Automata

Simplest pattern recognizers.

Reads inputs from left to right, either accepts or rejects them.

DFA (deterministic) and NFA (nondeterministic):

- DFAs have one and only one option at any time.
- NFAs can have several options.

Finite Automata

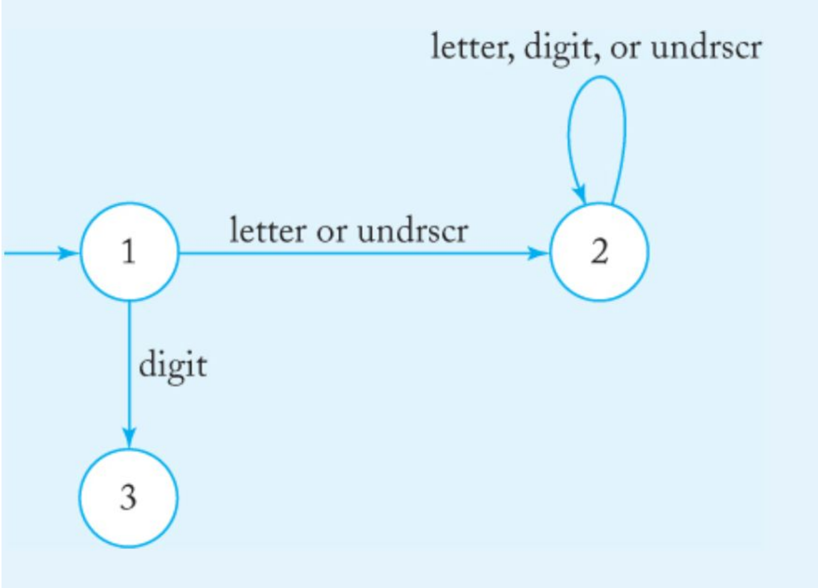


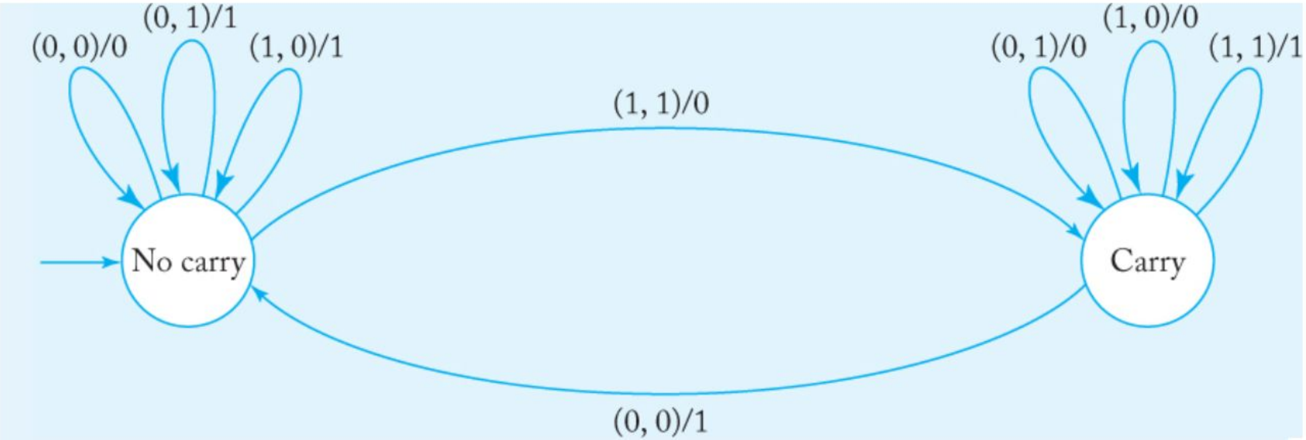
FIGURE 1.6

C identifiers:

Start with state 1 (arrow with no origin).

If first symbol is letter or undrscr, go to state 2 -> accepting state
If first symbol is digit, go to state 3 -> rejecting/trap state

Finite Automata



Binary adder:

Start with No carry state (arrow with no origin).

Example: add 101 and 011

Position 0 (rightmost): (1,1)/0, sum 0, carry 1, go to Carry state

Position 1: (0,1) + carry 1, sum 0, carry 1, stay in Carry state

Position 2: (1,0) + carry 1, sum 0, carry 1, stay in Carry state

Position 3: overflow, carry 1 -> sum 1

DFA Definition

DEFINITION 2.1

A **deterministic finite accepter** or **dfa** is defined by the quintuple $M = (Q, \Sigma, \delta, q_0, F)$,

where

Q is a finite set of **internal states**, Σ is a finite set of symbols called the **input alphabet**, $\delta : Q \times \Sigma \rightarrow Q$ is a total function called the **transition function**, $q_0 \in Q$ is the **initial state**, $F \subseteq Q$ is a set of **final states**.

Q is the set of states. There are only finitely many.

Σ is the input alphabet. The set of symbols the machine can read.

δ is the transition function.

q_0 is the initial state.

F is the set of final or accepting states.

DFA Example (Figure 2.1)

M = ({q0,q1,q2},{0,1}, δ, q0, {q1})

$$\begin{aligned} \delta(q_0,0) &= q_0, & \delta(q_0,1) &= q_1, \\ \delta(q_1,0) &= q_0, & \delta(q_1,1) &= q_2, \\ \delta(q_2,0) &= q_2, & \delta(q_2,1) &= q_1. \end{aligned}$$

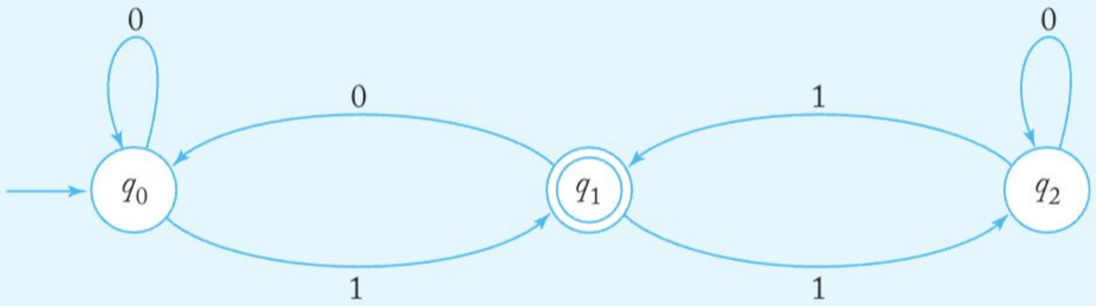


FIGURE 2.1

DFA Example (Figure 2.1)

M = ({q0,q1,q2},{0,1}, δ, q0, {q1})

$$\delta(q_0,0) = q_0, \qquad \delta(q_0,1) = q_1,$$
$$\delta(q_1,0) = q_0, \qquad \delta(q_1,1) = q_2,$$
$$\delta(q_2,0) = q_2, \qquad \delta(q_2,1) = q_1.$$

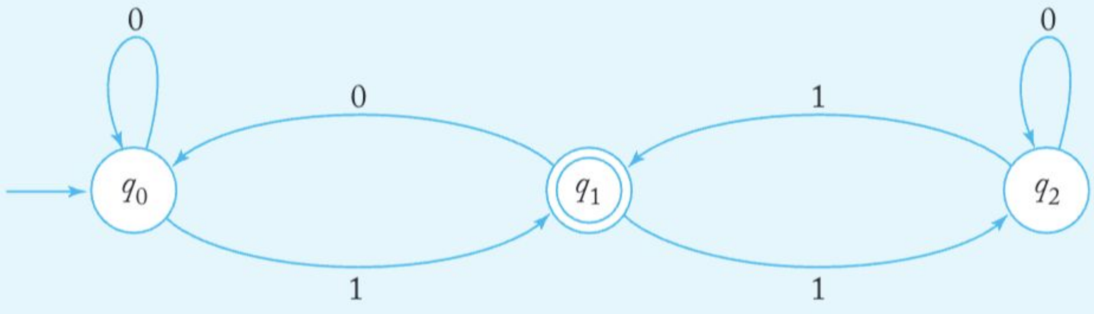


FIGURE 2.1

Example 1: (Textbook page 48, exercise 1)

1. Which of the strings 0001, 01101, 00001101 are accepted by the dfa in Figure 2.1?

DFA Example (Figure 2.5)

A DFA that accepts all the strings on {0,1}, except those containing the substring 001.

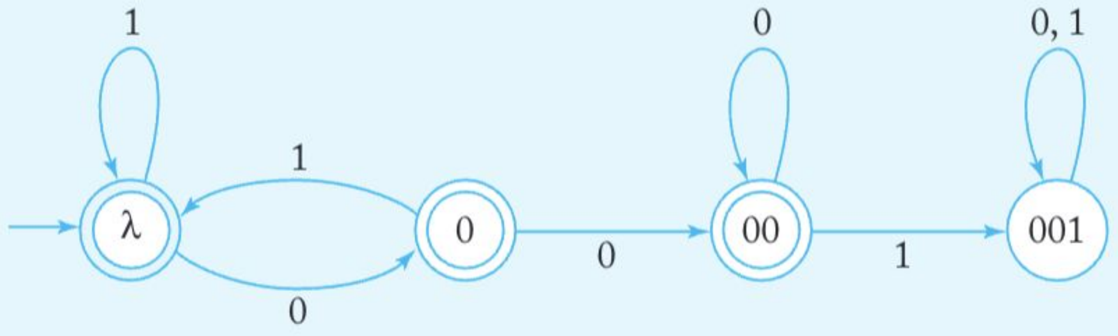


FIGURE 2.5

DFA Example (Figure 2.5)

A DFA that accepts all the strings on {0,1}, except those containing the substring 001.

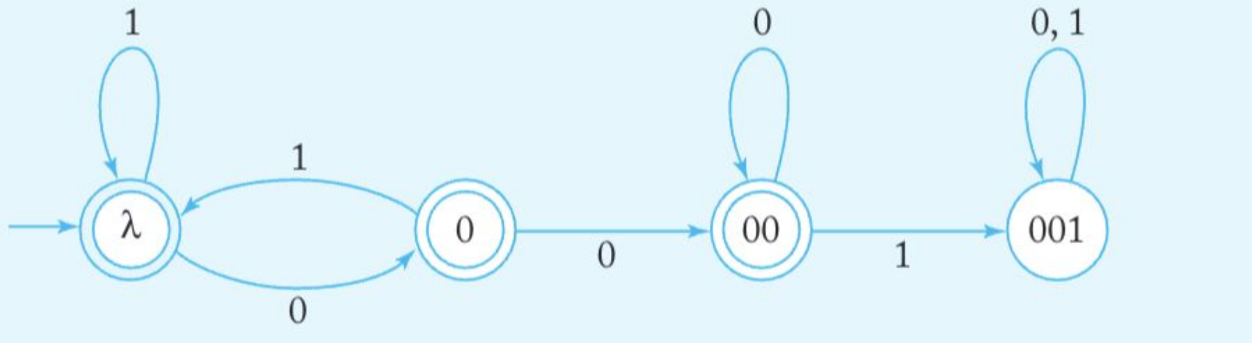


FIGURE 2.5

Example 1: (Textbook page 48, exercise 2)

2. Translate the graph in Figure 2.5 into δ- notation.

DFAs

In-class Exercise:

[Textbook page 48, exercise 3]

3. For $\Sigma = \{a, b\}$, construct dfa's that accept the sets consisting of

   (a) all strings of even length.

   (b) all strings of length greater than 5.

   (c) all strings with an even number of a's.

   (d) all strings with an even number of a's and an odd number of b's.

Today’s Takeaways

Three Basic Concepts

DFAs

Exit Token

- Write on a piece of paper:
  - Name
  - Time and location of CS Fall Meet & Greet