CSE-217: Theory of Computation

REGULAR LANGUAGES

Md Jakaria

Lecturer
Department of Computer Science and Engineering
Military Institute of Science and Technology

July 18, 2019

DESIGNING FINITE AUTOMATA

Design is a creative process!

DESIGNING FINITE AUTOMATA

Put yourself in the place of the machine you are trying to design.

DESIGNING FINITE AUTOMATA

Put yourself in the place of the machine you are trying to design.

Figure out what you need to remember about the string.

Example - 1

Sipser, 1.1, p-41

Example 1

1. Suppose that the alphabet is {0,1} and that the language consists of all strings with an odd number of 1s.
2. You want to construct a finite automaton E_1 to recognize this language.

Example - 1

Sipser, 1.1, p-42

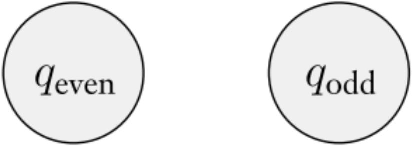


FIGURE 1.18
The two states q_{even} and q_{odd}

Example - 1

Sipser, 1.1, p-42

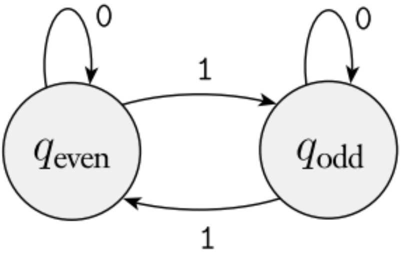


FIGURE 1.19
Transitions telling how the possibilities rearrange

Example - 1

Sipser, 1.1, p-43

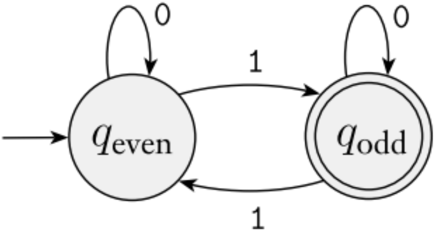


FIGURE 1.20
Adding the start and accept states

Example - 2

Sipser, 1.1, p-43

Example 2

1. Design a finite automaton E_2 to recognize the regular language of all strings that contain the string 001 as a substring.
2. For example, 0010, 1001, 001, and 11111110011111 are all in the language, but 11 and 0000 are not.

Example - 2

Sipser, 1.1, p-44

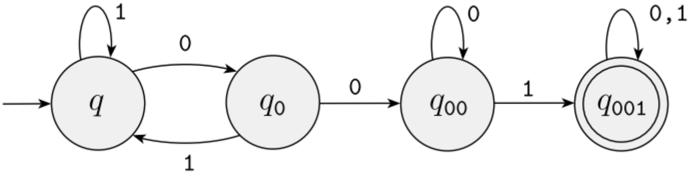


FIGURE 1.22
Accepts strings containing 001

Example - 3

Sipser, 1.1, p-43

Example 3

1. Let us formally specify a DFA that accepts all and only the strings of 0's and 1's that have the sequence 01 somewhere in the string.

Example - 3
Ullman, 2.1, Fig-2.4

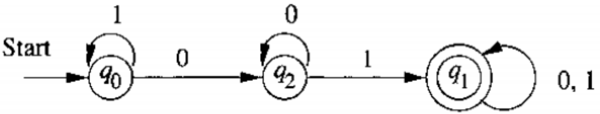


Figure 2.4: The transition diagram for the DFA accepting all strings with a substring 01

Example - 3

Sipser, 1.1, p-43

- We can write this language L as:
  {w | w is of the form x01y for some strings x and y consisting of 0's and 1's only.}
- Another equivalent description, using parameters x and y to the left of the vertical bar, is:
  {x01y | x and y are any strings of 0's and 1's}

Example - 4

Sipser, 1.1, p-43

Example 4

1. Design a DFA to accept the language L = {w | w has both an even number of 0's and an even number of 1's}

Example - 4

Ullman, 2.1, Fig-2.6

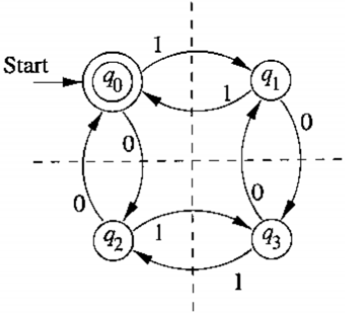
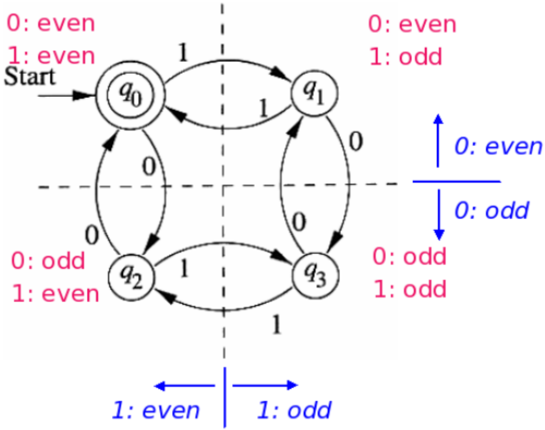


Figure 2.6: Transition diagram for the DFA of Example 2.4

Example - 4

Ullman, 2.1, Fig-2.6



Example - 5

Lewis and Papadimitriou, Example 2.1.2

Example 5

1. Design a deterministic finite automaton M that accepts the language
   $L(M) = \{ w \in \{a,b\}^* : w \text{ does not contain three consecutive b's}\}$

Example - 5

Ullman, 2.1, Fig-2.6

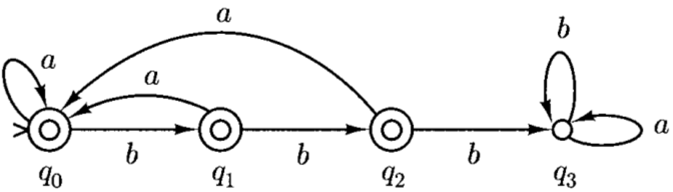


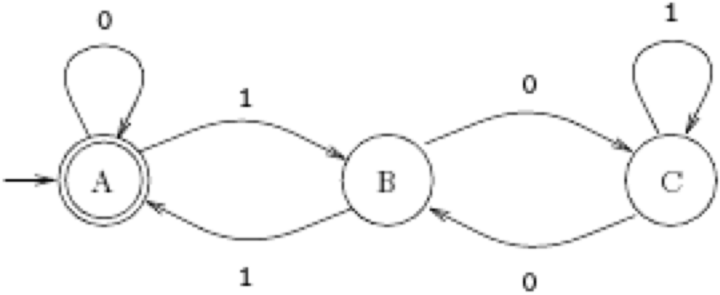
Figure 2-3

Example 6

1. Design a DFA that accepts binary numbers that are divisible by three.

Example - 6

Ullman, 2.1, Fig-2.6



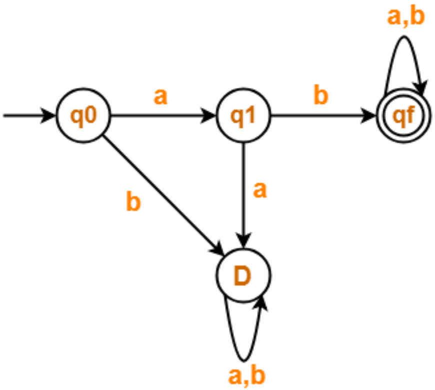
Example - 7

Example 7

1 Draw a DFA for the language accepting strings starting with ‘ab’ over input alphabets $\Sigma = \{a, b\}$

Example - 7

Ullman, 2.1, Fig-2.6



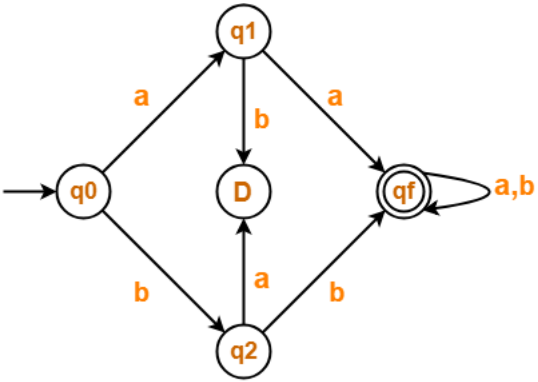
DFA

Example - 8

Example 8

1. Construct a DFA that accepts a language L over input alphabets$\Sigma = \{a, b\}$ such that L is the set of all strings starting with ‘aa’ or ‘bb’.

Example - 8



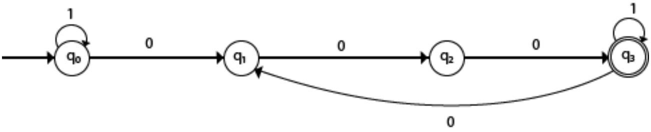
DFA

Example - 9

Example 9

1 Design FA with $\Sigma = \{0, 1\}$ accepts the set of all strings with three consecutive 0's.

Example - 9

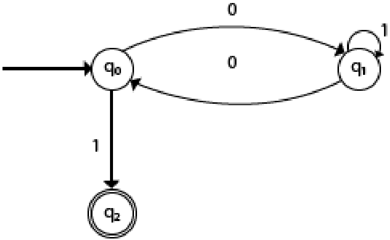


Example - 10

Example 10

1. Design a FA with $\Sigma = \{0, 1\}$ accepts the strings with an even number of 0's followed by single 1.

Example - 10

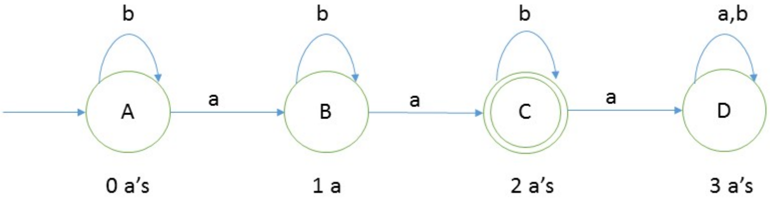


Example - 11

Example 11

1. Design a DFA over w $\in$ {a,b}* such that number of a = 2 and there is no restriction over length of b.

Example -11

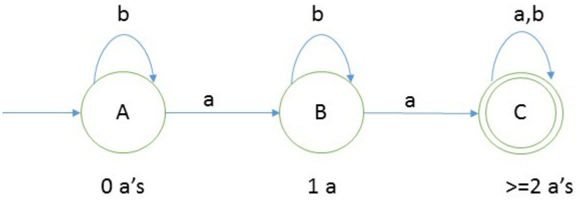


Example - 12

Example 12

1. Design a DFA over w $\in$ {a,b}* such that number of a is less or equals to 2 and there is no restriction over length of b.

Example -12

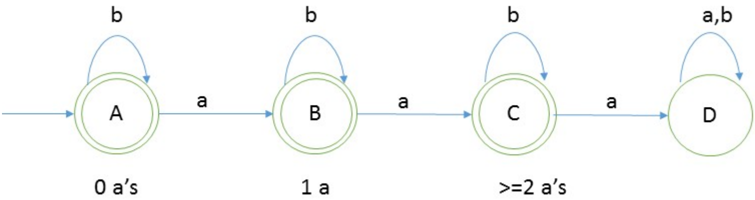


Example - 13

Example 13

1. Design a DFA over w $\in$ {a,b}* such that number of a is greater or equals to 2 and there is no restriction over length of b.

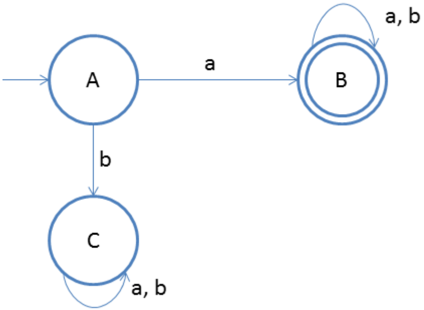
Example -13



Example 14

1. Design a DFA over w $\in$ {a,b}* in which set of all strings can be accepted which start with a.

Example -14

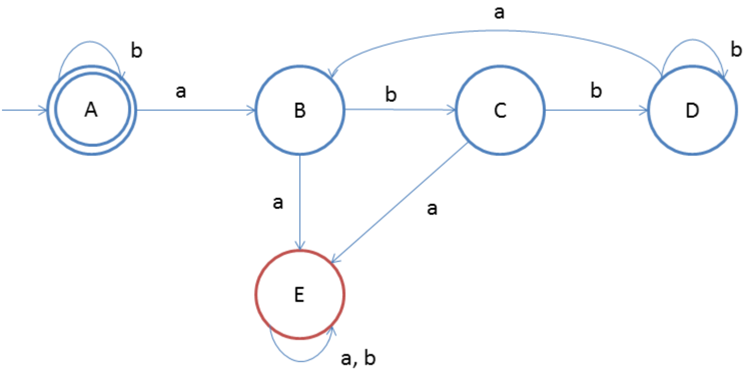


Example - 15

Example 15

1. Design a DFA over w $\in$ {a,b}* in which every 'a' should followed by 'bb'

Example -15

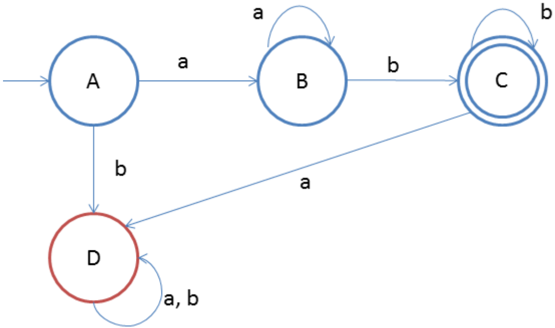


Example - 16

Example 16

1. Design a DFA such that:
   L = {$a^n b^m$ | n,m $\geq$ 1} Given: Input alphabet, Σ={a, b}
   Language L = {ab, aab, aaab, abbb, aabb, abbbb, ...}

Example -16



Ar Parina Boddo Hoyran Lage