NFAs to DFAs Examples

Lecture 8
Section 2.3

Robb T. Koether

Hampden-Sydney College

Fri, Sep 9, 2016

Outline

Outline

Example

Example (Even Number of **a**'s and **b**'s)

- Let $\Sigma = \{\mathbf{a}, \mathbf{b}\}$.
- Let $L_1 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{a}\text{'s}\}$.
- Let $L_2 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{b}\text{'s}\}$.
- Convert the NFA that accepts $L_1 \cup L_2$ to a DFA.

Example

Example (Even Number of **a**'s and **b**'s)

- Let $\Sigma = \{\mathbf{a}, \mathbf{b}\}$.
- Let $L_1 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{a}\text{'s}\}$.
- Let $L_2 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{b}\text{'s}\}$.
- Convert the NFA that accepts $L_1 \cup L_2$ to a DFA.
- Convert the NFA that accepts $L_1 L_2$ to a DFA.

Example (Even Number of **a**'s and **b**'s)

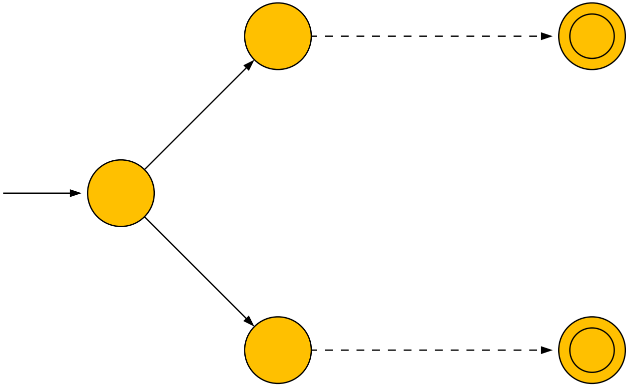
- Let $\Sigma = \{\mathbf{a}, \mathbf{b}\}$.
- Let $L_1 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{a}\text{'s}\}$.
- Let $L_2 = \{w \in \Sigma^* \mid w \text{ contains an even number of } \mathbf{b}\text{'s}\}$.
- Convert the NFA that accepts $L_1 \cup L_2$ to a DFA.
- Convert the NFA that accepts $L_1 L_2$ to a DFA.
- In the last example, process **ababb**, **abaabb**, and **aababb**.

Outline

Creating M^R from M

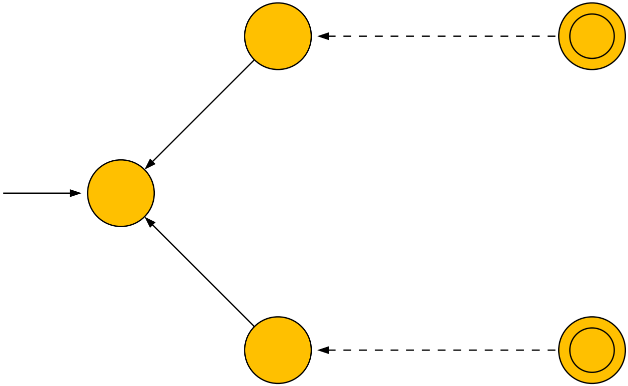
- Given a machine M that accepts a language L, we can construct a machine M^R that accepts the language L^R as follows.
  - Reverse all the arrows in the transition diagram for M.
  - Create a new start state q_0'.
  - Create λ-moves from q_0' to each of M's final states.
  - Make all of M's final states nonfinal.
  - Make M's start state final.

Creating M^R from M



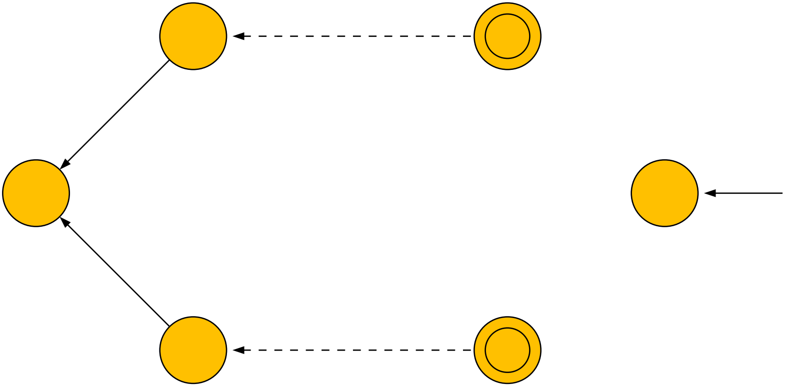
A generic DFA

Creating M^R from M



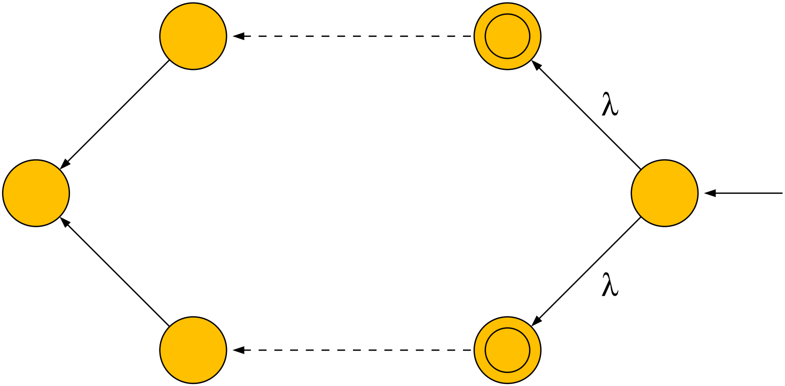
Reverse all the arrows

Creating M^R from M



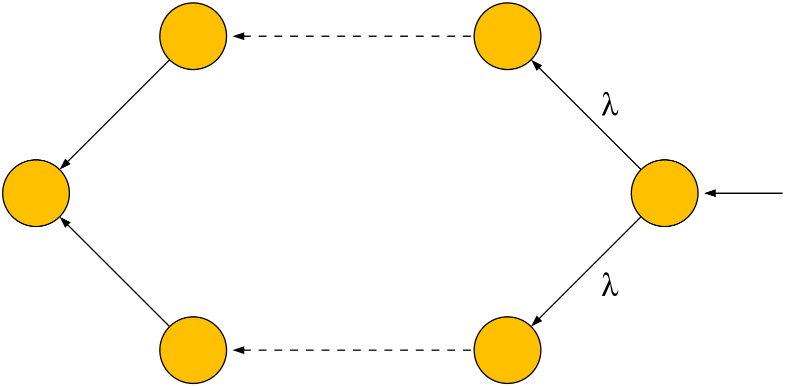
Create a new start state

Creating M^R from M



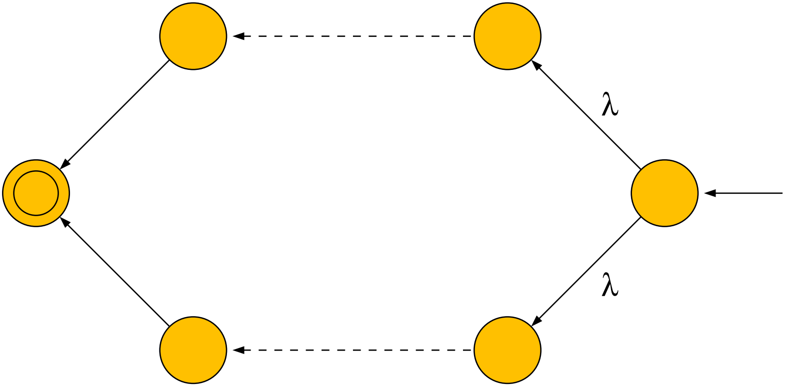
Add λ-moves to the final states

Creating M^R from M



Make the final states non-final

Creating M^R from M

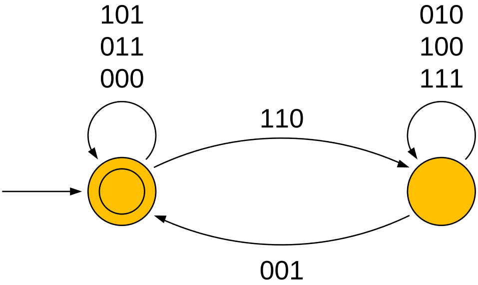


Make the original start state the final state

Example (Binary Adder)

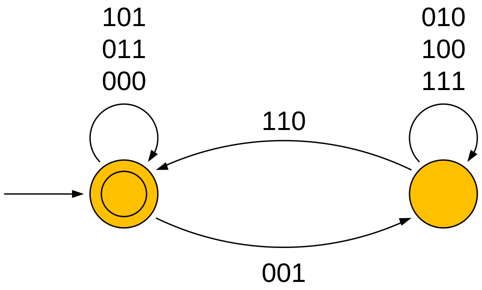
- Build a DFA that will recognize a correct binary addition by reading the columns from left to right.

Creating M^R from M



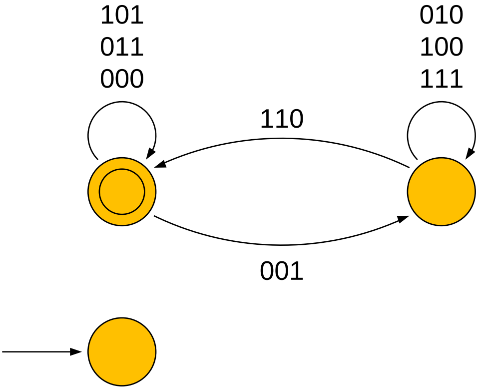
The original DFA

Creating M^R from M



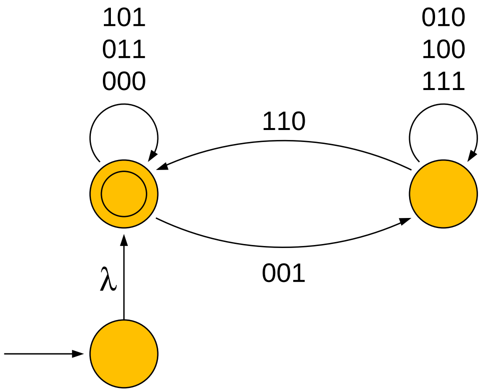
Reverse all the arrows

Creating M^R from M



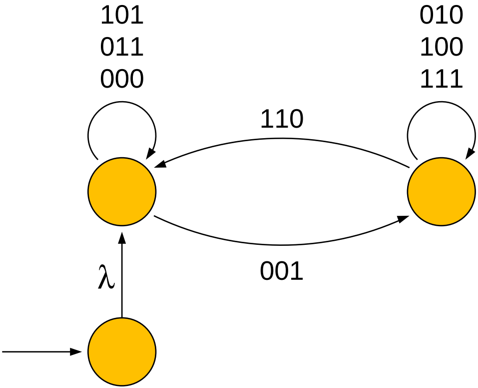
Create a new start state

Creating M^R from M



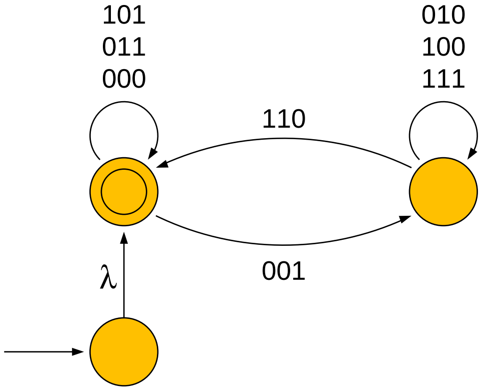
Add λ-moves to the final states

Creating M^R from M



Make the final states non-final

Creating M^R from M

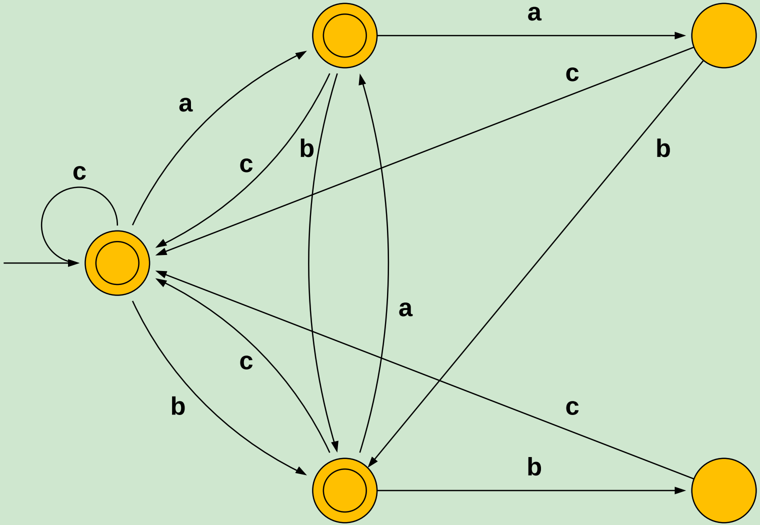


Make the original start state the final state

Example

- Let $\Sigma = \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$.
- Build a DFA that will recognize strings in which **aa** is always followed immediately by either **b** or **c** and **bb** is always followed immediately by **c**.

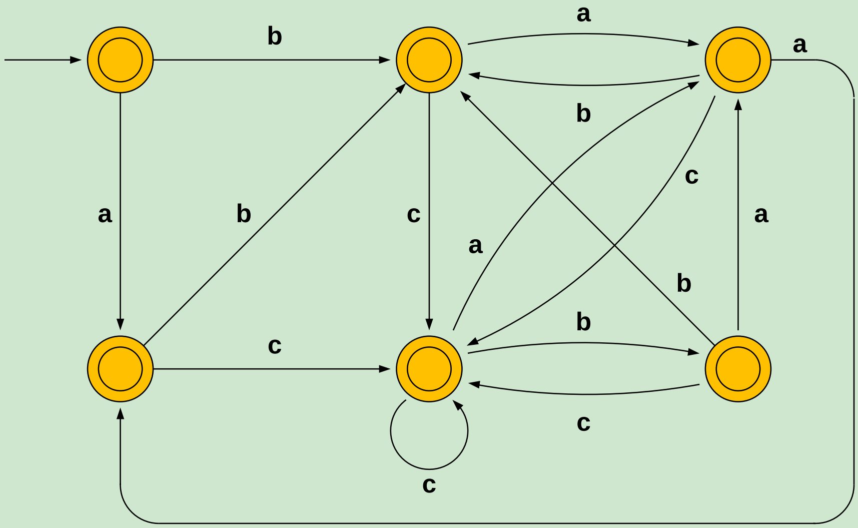
Example



Example

- Build a DFA that will accept the reverse of the language of the previous example.

Example



Outline

Programming Assignment

Programming Assignment

- To be collected Wednesday, September 14.
- Use JFLAP to build the following automata.
  (1) A DFA that will accept $L = \{\mathbf{a}^n \mid n \geq 1\} \cup \{\mathbf{b}^n\mathbf{a} \mid n \geq 1\}$.
  (2) An NFA that will accept L^*, where L is as in the previous problem.
  (3) A DFA that will validate base-3 addition problems, reading the columns from left to right.

Outline

Assignment

Assignment

- Section 2.2 Exercises 19, 23
- Section 2.3 Exercises 3, 4, 5, 7, 9, 10, 16.