بسم الله الرحمن الرحیم

نظریه زبان‌ها و ماشین‌ها

جلسه ۹

مجتبی خلیلی
دانشکده برق و کامپیوتر
دانشگاه صنعتی اصفهان

زبان‌های متناظر با عبارت‌های منظم

DEFINITION 3.2

The language $L(r)$ denoted by any regular expression r is defined by the following rules.

1. $\varnothing$ is a regular expression denoting the empty set,

2. λ is a regular expression denoting $\{\lambda\}$,

3. For every $a \in \Sigma$, a is a regular expression denoting $\{a\}$.

If r_1 and r_2 are regular expressions, then

4. $L(r_1 + r_2) = L(r_1) \cup L(r_2)$,

5. $L(r_1 \cdot r_2) = L(r_1) L(r_2)$,

6. $L((r_1)) = L(r_1)$,

7. $L(r_1^*) = (L(r_1))^*$.

زبان‌های متناظر با عبارت‌های منظم

IUT-ECE

عبارت منظم	زبان منظم
ϕ	$\{\}$
ϵ	$\{\epsilon\}$
a	$\{a\}$
$a \cup b$	$\{a, b\}$
ab	$\{a\}\{b\}$
a^*	$\{a\}^* = \{\epsilon, a, aa, aaa, \ldots\}$
$(aab)^*(a \cup ab)$	$\{aab\}^*\{a, ab\}$
$(aa \cup bb \cup (a \cup b)(aa)^*(ab \cup ba))^*$	$(\{aa, bb\} \cup \{a, b\}\{aa\}^*\{ab, ba\})^*$

مثال

EXAMPLE 3.2

Exhibit the language $L(a^* \cdot (a+b))$ in set notation.

$$
\begin{aligned}
L(a^* \cdot (a+b)) &= L(a^*)\, L(a+b) \\
&= (L(a))^* (L(a) \cup L(b)) \\
&= \{\lambda, a, aa, aaa, \ldots\} \{a, b\} \\
&= \{a, aa, aaa, \ldots, b, ab, aab, \ldots\}.
\end{aligned}
$$

مثال

EXAMPLE 3.3

For $\Sigma = \{a, b\}$, the expression

$$r = (a + b)^* (a + bb)$$

is regular. It denotes the language

$$L(r) = \{a, bb, aa, abb, ba, bbb, ...\}.$$

We can see this by considering the various parts of r. The first part, $(a + b)^*$, stands for any string of a's and b's. The second part, $(a + bb)$ represents either an a or a double b. Consequently, $L(r)$ is the set of all strings on $\{a, b\}$, terminated by either an a or a bb.

مثال

EXAMPLE 3.4

The expression

$$r = (aa)^* (bb)^* b$$

denotes the set of all strings with an even number of a's followed by an odd number of b's; that is,

$$L(r) = \{a^{2n} b^{2m+1} : n \geq 0, \ m \geq 0\}.$$

عبارت‌های معادل/هم ارز

- دو عبارت منظم را معادل گوییم اگر هر دو یک زبان را توصیف کنند. مثال:

$$(a^*b^*)^* = (a+b)^* = \Sigma^*$$

چند قاعده

- فرض کنید R_1 و R_2 و R_3 عبارت‌های منظم باشند؛ آنگاه:

$$R_1\phi = \phi R_1 = \phi$$
$$R_1\epsilon = \epsilon R_1 = R_1 \cup \phi = \phi \cup R_1 = R_1$$
$$R_1 \cup R_1 = R_1$$
$$R_1 \cup R_2 = R_2 \cup R_1$$
$$R_1(R_2 \cup R_3) = R_1R_2 \cup R_1R_3$$
$$(R_1 \cup R_2)R_3 = R_1R_3 \cup R_2R_3$$
$$R_1(R_2R_3) = (R_1R_2)R_3$$
$$\phi^* = \epsilon$$

چند قاعده

$R \cup \varepsilon$ may not equal R.
For example, if $R = 0$, then $L(R) = \{0\}$ but $L(R \cup \varepsilon) = \{0, \varepsilon\}$.

چند قاعده

$R \circ \emptyset$ may not equal R.
For example, if $R = 0$, then $L(R) = \{0\}$ but $L(R \circ \emptyset) = \emptyset$.

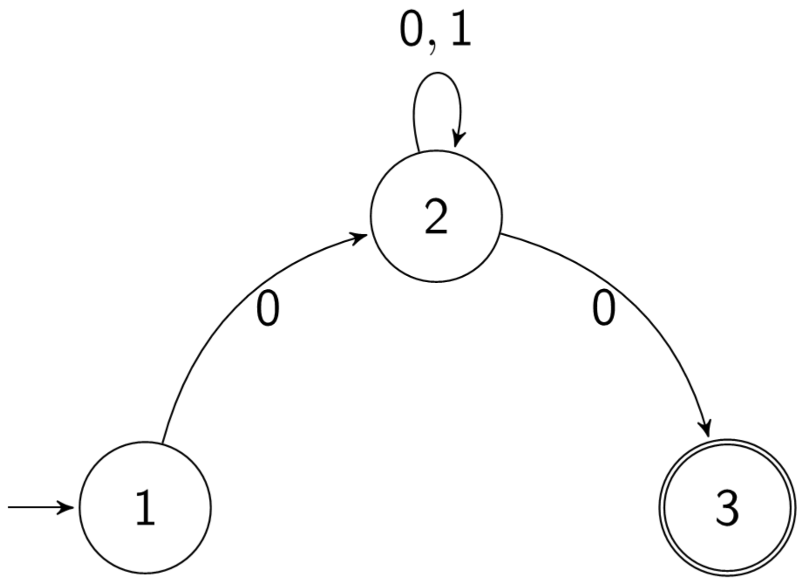
مثال

- عبارت منظم برای رشته‌هایی که شامل تعداد فرد 0 باشد (الفبای باینری).

عبارت منظم/اتوماتا

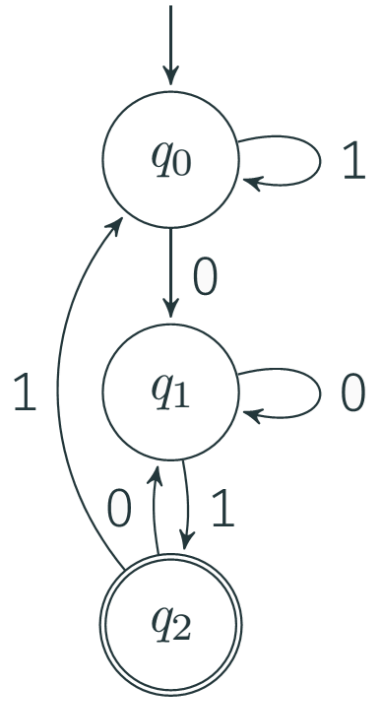
- ارتباط بین RE و DFA/NFA چیست؟

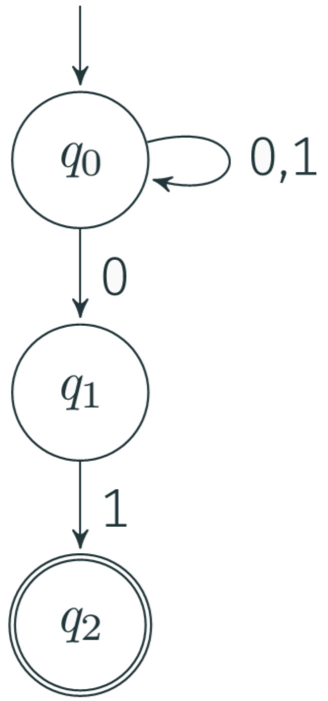
$0(0 \cup 1)^*0$:



عبارت منظم/اتوماتا

- زبانی شامل همه رشته‌های ختم به 01





$(0 + 1)^*01$

عبارت منظم/اتوماتا

- ارتباط بین RE و DFA/NFA چیست؟
  - آیا همه RE ها توسط DFA/NFA قابل نمایش هستند؟
  - آیا همه DFA/NFA ها توسط RE قابل توصیف هستند؟

عبارت‌های منظم/زبان منظم/اتوماتای متناهی

Regular expressions and finite automata are equivalent in their descriptive power. This fact is surprising because finite automata and regular expressions superficially appear to be rather different. However, any regular expression can be converted into a finite automaton that recognizes the language it describes, and vice versa. Recall that a regular language is one that is recognized by some finite automaton.

عبارت‌های منظم/زبان منظم

THEOREM 1.54

A language is regular if and only if some regular expression describes it.

اثبات (ادامه)

LEMMA 1.55

If a language is described by a regular expression, then it is regular.

- اثبات:

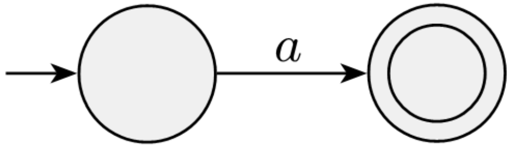
عبارت منظم R $\longrightarrow$ NFA N

$L(R) = A$ $\qquad$ $L(N) = A$

اثبات (ادامه)

PROOF Let's convert R into an NFA N. We consider the six cases in the formal definition of regular expressions.

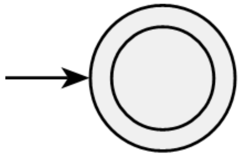
1. $R = a$ for some $a \in \Sigma$. Then $L(R) = \{a\}$, and the following NFA recognizes $L(R)$.



Formally, $N = (\{q_1, q_2\}, \Sigma, \delta, q_1, \{q_2\})$, where we describe δ by saying that $\delta(q_1, a) = \{q_2\}$ and that $\delta(r, b) = \emptyset$ for $r \neq q_1$ or $b \neq a$.

اثبات (ادامه)

2. $R = \varepsilon$. Then $L(R) = \{\varepsilon\}$, and the following NFA recognizes $L(R)$.

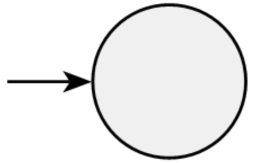


Formally, $N = \big(\{q_1\}, \Sigma, \delta, q_1, \{q_1\}\big)$, where $\delta(r, b) = \emptyset$ for any r and b.

اثبات (ادامه)

IUT-ECE

3. $R = \emptyset$. Then $L(R) = \emptyset$, and the following NFA recognizes $L(R)$.



Formally, $N = \big(\{q\}, \Sigma, \delta, q, \emptyset\big)$, where $\delta(r, b) = \emptyset$ for any r and b.

اثبات (ادامه)

4. $R = R_1 \cup R_2$.

5. $R = R_1 \circ R_2$.

6. $R = R_1^*$.

For the last three cases, we use the constructions given in the proofs that the class of regular languages is closed under the regular operations. In other words, we construct the NFA for R from the NFAs for R_1 and R_2 (or just R_1 in case 6) and the appropriate closure construction.

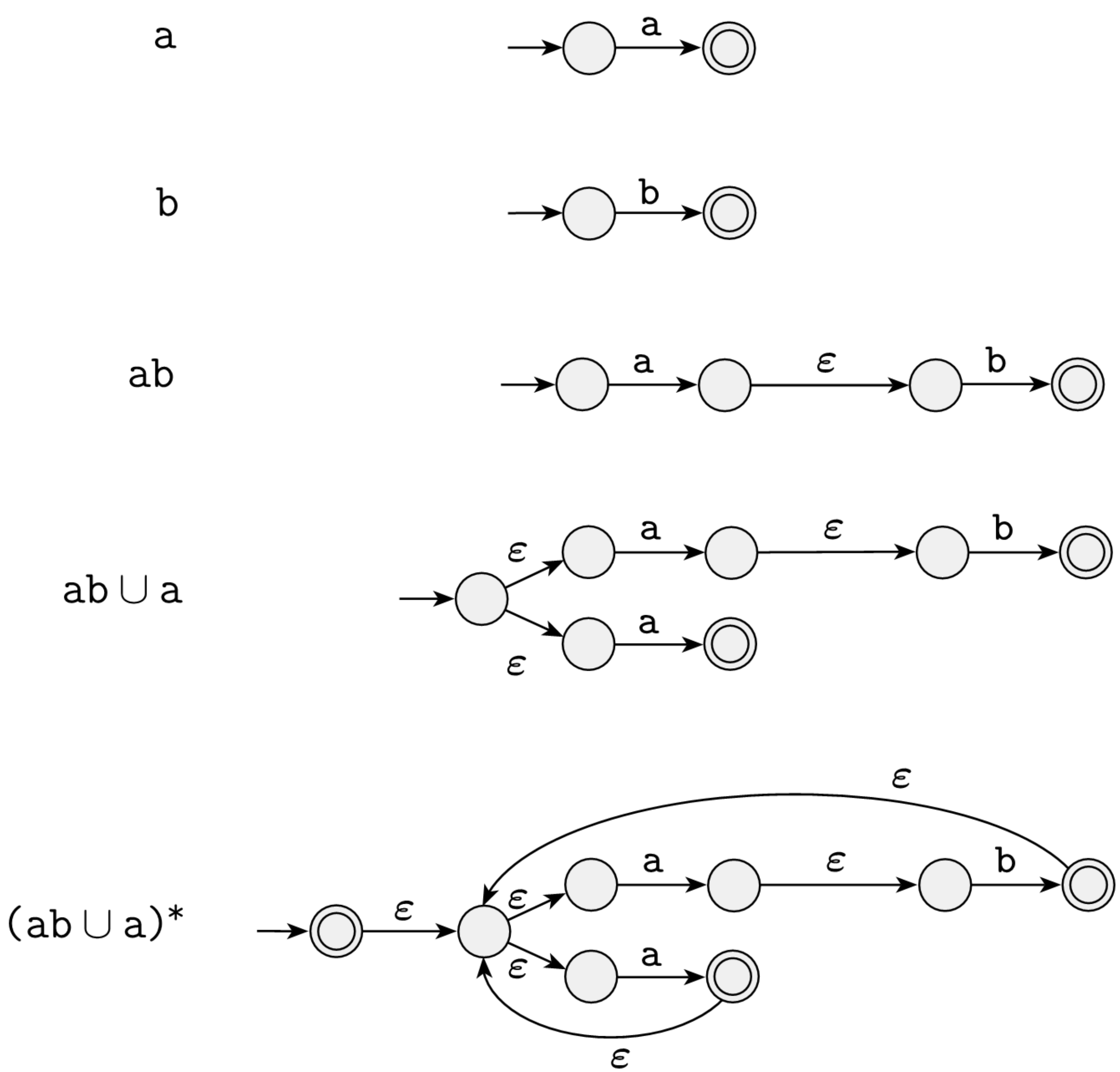
مثال

EXAMPLE 1.56

We convert the regular expression $(\mathtt{ab} \cup \mathtt{a})^*$ to an NFA in a sequence of stages.

مثال

$(\mathtt{ab} \cup \mathtt{a})^*$

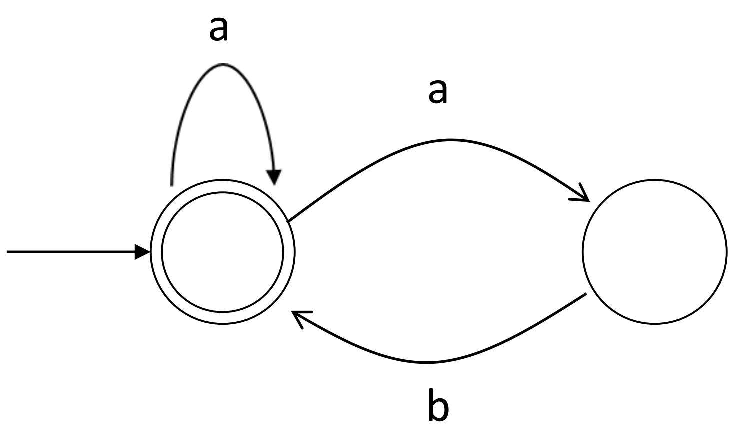


مثال

EXAMPLE 1.56 ....

We convert the regular expression $(\texttt{ab} \cup \texttt{a})^*$ to an NFA in a sequence of stages.

- به طور مستقیم؟

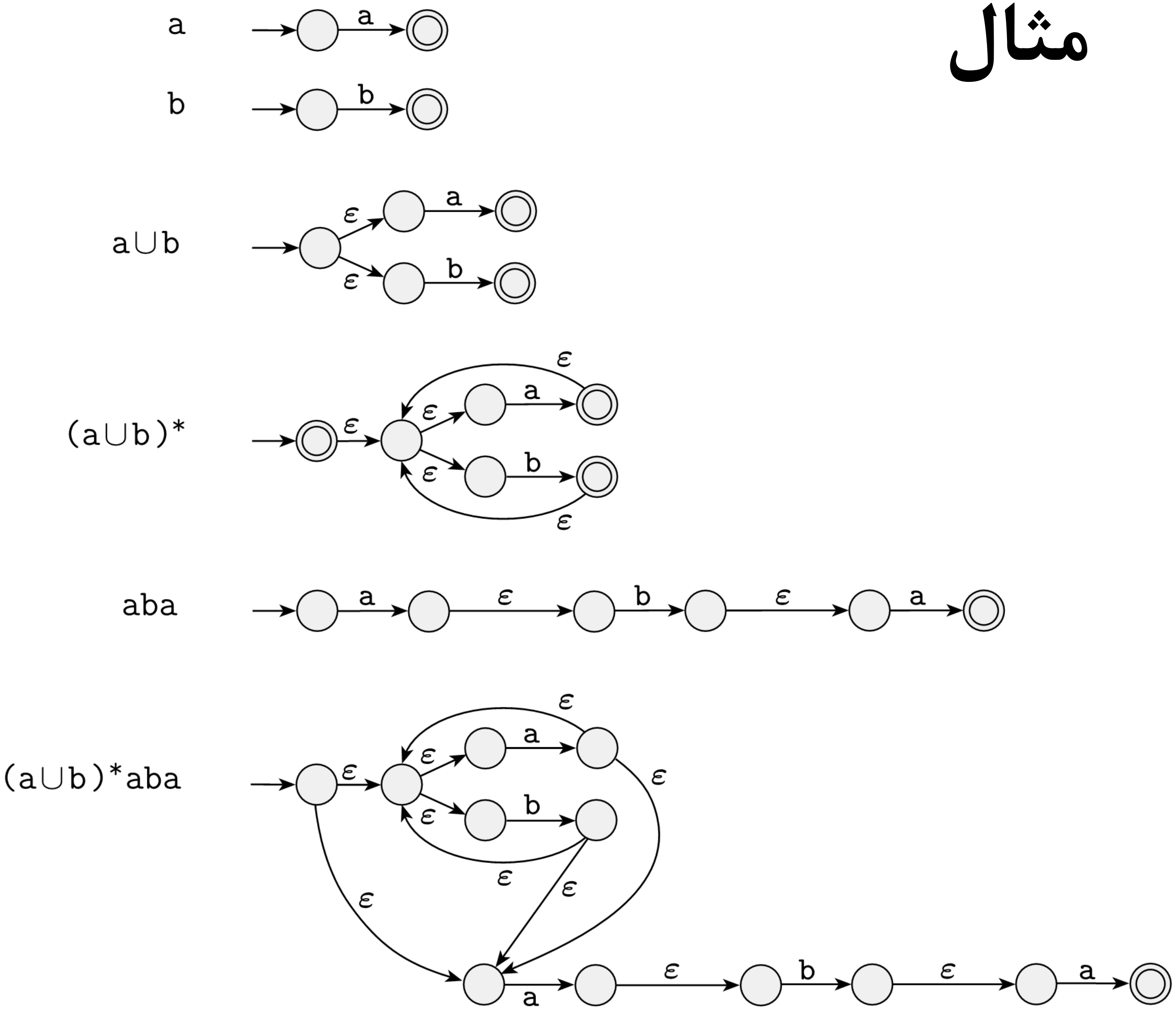


مثال

EXAMPLE 1.58

In Figure 1.59, we convert the regular expression $(\mathtt{a} \cup \mathtt{b})^*\mathtt{aba}$ to an NFA. A few of the minor steps are not shown.

مثال



$(\mathtt{a} \cup \mathtt{b})^*\mathtt{aba}$

اثبات (طرف دوم)

IUT-ECE

LEMMA 1.60

If a language is regular, then it is described by a regular expression.

- اثبات: