University of Central Florida

School of Computer Science

COT 4210 Fall 2004

Prof. Rene Peralta

Homework 3 Solutions (by TA Robert Lee)

1.11. Give a counterexample to show that the following construction fails to prove Theorem 1.24, the closure of the class of regular languages under the star operation. Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1. Construct $N = (Q_1, \Sigma, \delta, q_1, F)$ as follows. N is supposed to recognize A_1^*.

a. The states of N are the states of N_1.
b. The start state of N is the same as the start state of N_1.
c. $F = \{q_1\} \cup F_1$. The accept states F are the old accept states plus its start state.
d. Define δ so that for any $q \in Q$ and any $a \in \Sigma_\epsilon$,

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & q \notin F_1 \text{ or } a \neq \epsilon \\ \delta_1(q,a) \cup \{q_1\} & q \in F_1 \text{ and } a = \epsilon. \end{cases}$$

Solution

See Figure 1.11.1 on the next page for a picture of this formal construction. The primary difference between this construction and the one in Theorem 1.24 is that the original start state becomes an accept state. To create a counterexample, we want to consider a language that does not already accept in its start state.

Choose the language $A_1 = 1(01)^*$. Language A_1 is recognized by NFA N_1:
$N_1 = (\{q_1, q_2\}, \{0, 1\}, \delta_1, q_1, \{q_2\})$, where δ_1 is given by

δ_1	0	1	ϵ
q_1	$\emptyset$	$\{q_2\}$	$\emptyset$
q_2	$\{q_1\}$	$\emptyset$	$\emptyset$

Using the procedure given in this problem, construct the NFA N to recognize A_1^*:
$N = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{q_1, q_2\})$, where δ is given by

δ_1	0	1	ϵ
q_1	$\emptyset$	$\{q_2\}$	$\emptyset$
q_2	$\{q_1\}$	$\emptyset$	$\{q_1\}$

Both NFA N_1 and NFA N are shown in Figure 1.11.2 on the next page. Notice that the constructed machine N accepts the string 10, which is not in the language $A_1^* = (1(01)^*)^*$. Thus N_1 is a valid counterexample to prove that this construction does not work.

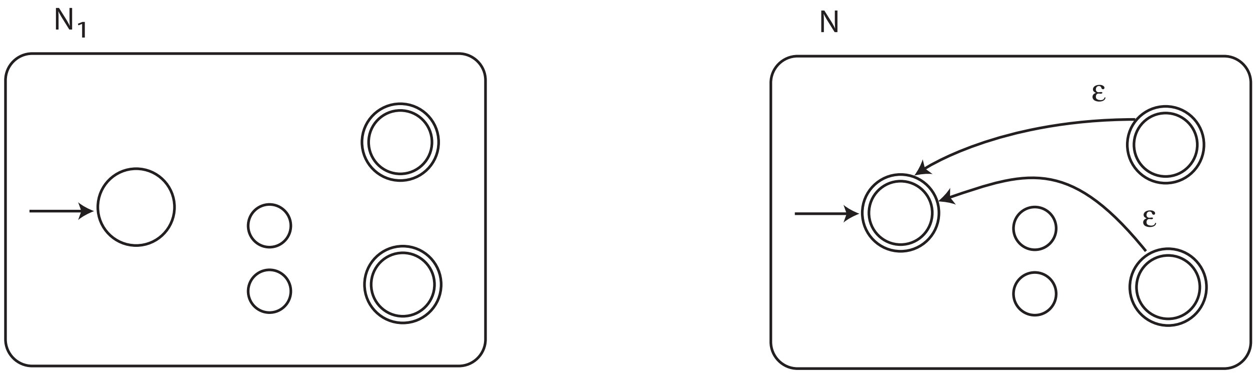


Fig. 1.11.1. The proposed construction to prove the closure of the regular languages under the star operation.

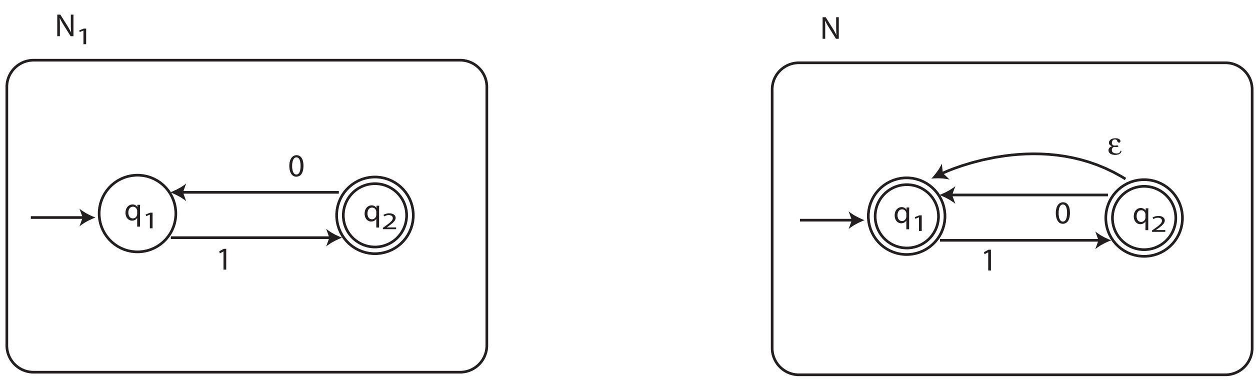


Fig. 1.11.2. The counterexample to show why this construction doesn't work. NFA N_1 accepts $1(01)^*$, but NFA N accepts 10, which is not in $(1(01)^*)^*$.

1.12 Use the construction given in Theorem 1.19 to convert the following two nondeterministic finite automata to equivalent deterministic finite automata.

Solution

a. This NFA is

$$(\{1,2\},\{a,b\},\delta_{NFA},1,\{1\}),$$

where δ_{NFA} is given by

δ_{NFA}	a	b	ϵ
1	$\{1,2\}$	$\{2\}$	$\emptyset$
2	$\emptyset$	$\{1\}$	$\emptyset$

Using the construction in Theorem 1.19, the constructed DFA is

$$(\mathcal{P}(\{1,2\}),\{a,b\},\delta_{DFA},\{1\},\{\{1\},\{1,2\}\}),$$

where δ_{DFA} is given by

δ_{DFA}	a	b
$\emptyset$	$\emptyset$	$\emptyset$
$\{1\}$	$\{1,2\}$	$\{2\}$
$\{2\}$	$\emptyset$	$\{1\}$
$\{1,2\}$	$\{1,2\}$	$\{1,2\}$

(See the figure on the page after part b.)

b. This NFA is $(\{1,2,3\},\{a,b\},\delta_{NFA},1,\{2\})$, where δ_{NFA} is given by

δ_{NFA}	a	b	ϵ
1	$\{3\}$	$\emptyset$	$\{2\}$
2	$\{1\}$	$\emptyset$	$\emptyset$
3	$\{2\}$	$\{2,3\}$	$\emptyset$

Using the construction in Theorem 1.19, the constructed DFA is

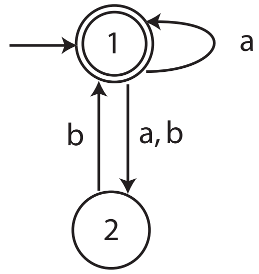
$$(\mathcal{P}(\{1,2,3\}),\{a,b\},\delta_{DFA},\{1,2\},\{\{2\},\{1,2\},\{2,3\},\{1,2,3\}\}),$$

where δ_{DFA} is given by

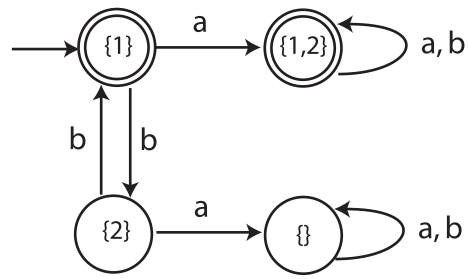
δ_{DFA}	a	b
$\emptyset$	$\emptyset$	$\emptyset$
$\{1\}$	$\{3\}$	$\emptyset$
$\{2\}$	$\{1\}$	$\emptyset$
$\{3\}$	$\{2\}$	$\{2,3\}$
$\{1,2\}$	$\{1,3\}$	$\emptyset$
$\{1,3\}$	$\{2,3\}$	$\{2,3\}$
$\{2,3\}$	$\{1,2\}$	$\{2,3\}$
$\{1,2,3\}$	$\{1,2,3\}$	$\{2,3\}$

(See the figure on the following page.)

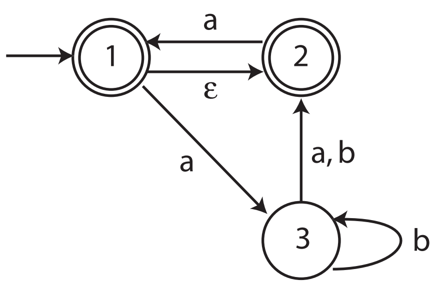
1.12a. Original NFA



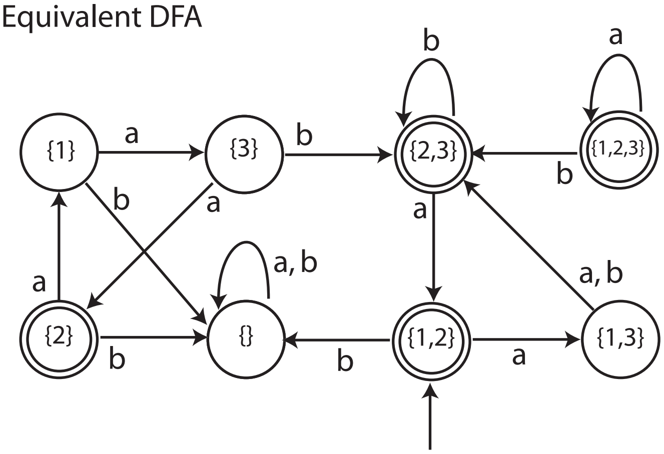
Equivalent DFA



1.12b. Original NFA



Equivalent DFA



1.13 Give regular expressions generating the languages of Exercise 1.4.

Solution One way to approach this problem is to start with the DFAs you have already drawn (in HW2) to answer Exercise 1.4. A regular expression can be derived by following each accepting path, that is, each path from the start state to an accept state, and writing a regular subexpression for each path. After all paths have been considered, take the union of the regular subexpressions to obtain the final regular expression. Because the alphabet $\Sigma = \{0, 1\}$ for each of these languages, we will write Σ as shorthand for $(0 \cup 1)$.

a. $\{w \mid w$ begins with a 1 and ends with a 0$\} = 11^*0(0 \cup 11^*0)^*$.

b. $\{w \mid w$ contains at least three 1s$\} = 0^*10^*10^*1\Sigma^*$.

c. $\{w \mid w$ contains the substring 0101$\} = 1^*00^*1(11^*00^*1)^*0(00^*1(0 \cup 11^*00^*10))^*1\Sigma^*$.

d. $\{w \mid w$ has length at least 3 and its third symbol is a 0$\} = \Sigma\Sigma0\Sigma^*$.

e. $\{w \mid w$ starts with 0 and has odd length, or starts with 1 and has even length$\} = (0 \cup 1\Sigma)(\Sigma\Sigma)^*$.

f. $\{w \mid w$ doesn't contain the substring 110$\} = 0^*(10)^*0^* \cup 0^*1(00^*1)^* \cup 0^*1(00^*1)^*11^*$.

g. $\{w \mid$ the length of w is at most 5$\} = \epsilon \cup \Sigma \cup \Sigma\Sigma \cup \Sigma\Sigma\Sigma \cup \Sigma\Sigma\Sigma\Sigma \cup \Sigma\Sigma\Sigma\Sigma\Sigma$.

h. $\{w \mid w$ is any string except 11 and 111$\} = \epsilon \cup 1 \cup 0\Sigma^* \cup 10\Sigma^* \cup 110\Sigma^* \cup 111\Sigma\Sigma^*$.

i. $\{w \mid$ every odd position of w is a 1$\} = (1\Sigma)^* \cup 1(\Sigma1)^*$.

j. $\{w \mid w$ contains at least two 0s and at most one 1$\} = 000^* \cup 000^*10^* \cup 0100^* \cup 1000^*$.

k. $\{\epsilon, 0\} = \epsilon \cup 0$.

l. $\{w \mid w$ contains an even number of 0s, or exactly two 1s$\}$ is generated by the following regular expression. Note that the subexpressions are grouped by accept state in the order q_2, q_4, q_6, q_8, q_5, based on the DFA in the HW2 solution key for 5(l).

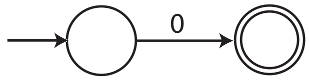
$$
\begin{array}{ll}
 & (00)^* \\
\cup & 1(00)^* \cup 0(00)^*10(00)^* \\
\cup & 1(00)^*1(00)^* \cup 0(00)^*10(00)^*1(00)^* \\
 & \cup 0(00)^*1(00)^*10(00)^* \cup 10(00)^*10(00)^* \\
\cup & 1(00)^*1(00)^*1(01^*0 \cup 1)^* \cup 1(00)^*10(00)^*11^*0(01^*0 \cup 1)^* \\
 & \cup 10(00)^*1(00)^*11^*0(01^*0 \cup 1)^* \cup 10(00)^*10(00)^*1(01^*0 \cup 1)^* \\
 & \cup 0(00)^*1(00)^*1(00)^*11^*0(01^*0 \cup 1)^* \cup 0(00)^*1(00)^*10(00)^*1(01^*0 \cup 1)^* \\
 & \cup 0(00)^*10(00)^*1(00)^*1(01^*0 \cup 1)^* \cup 0(00)^*10(00)^*10(00)^*11^*0(01^*0 \cup 1)^* \\
\cup & 1(00)^*10(00)^* \cup 10(00)^*1(00)^* \cup 0(00)^*1(00)^*1(00)^* \cup 0(00)^*10(00)^*10(00)^*
\end{array}
$$

m. The empty set is generated by $\emptyset$.

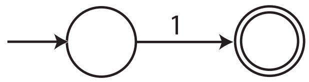
n. The language containing all strings except the empty string is generated by $\Sigma\Sigma^*$.

1.14a. (0 U 1)*000(0 U 1)*

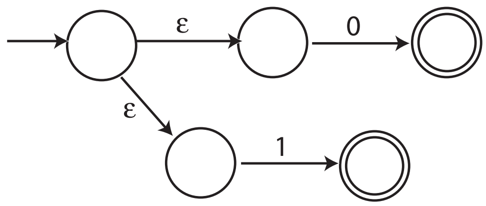
0



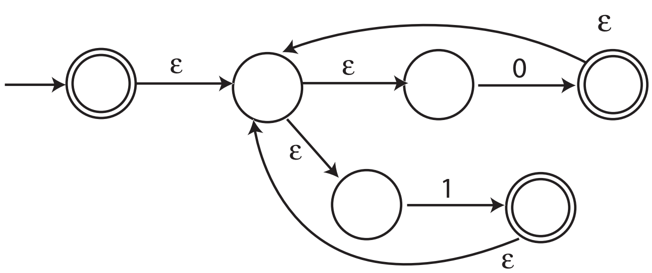
1



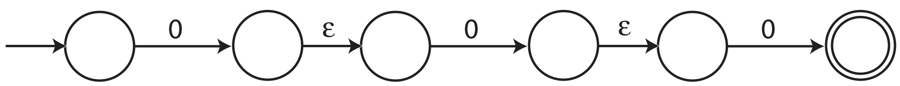
0 U 1



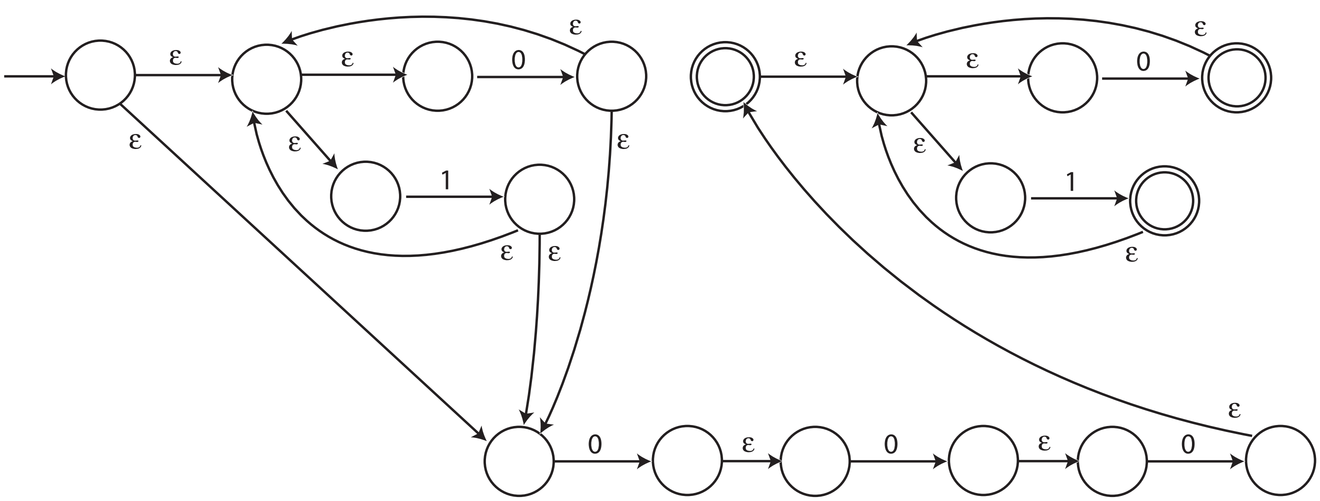
(0 U 1)*



000

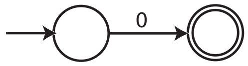


(0 U 1)*000(0 U 1)*

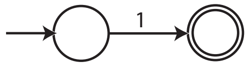


1.14b. (((00)*(11)) U 01)*

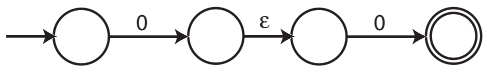
0



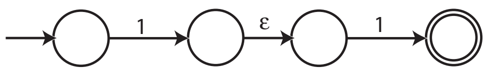
1



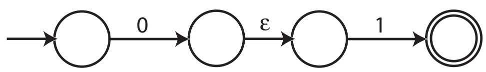
00



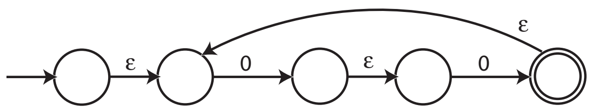
11



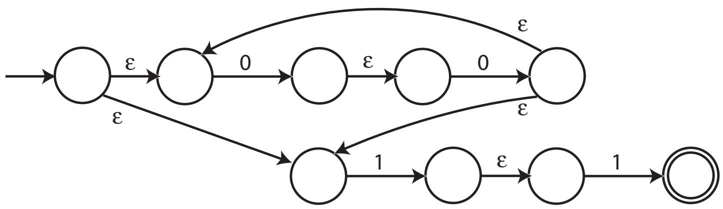
01



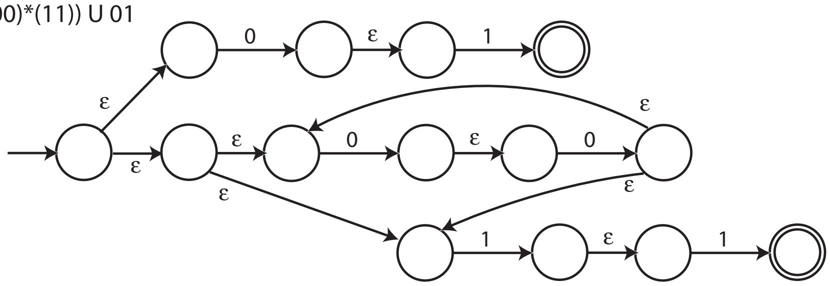
(00)*



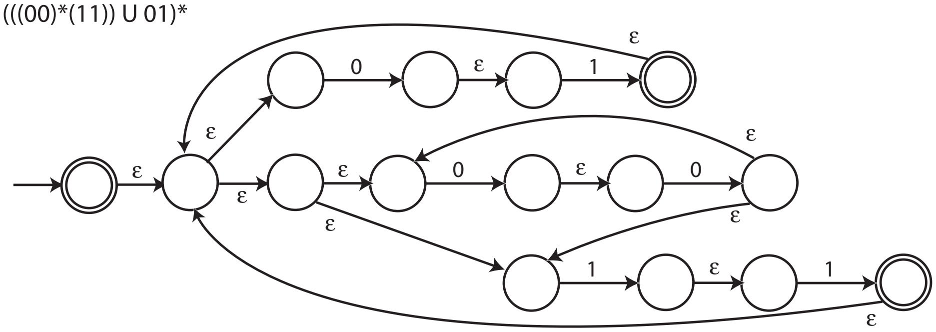
(00)*(11)



((00)*(11)) U 01

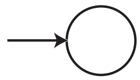


(((00)*(11)) U 01)*

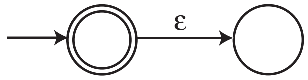


1.14c. θ*

θ



θ*



1.15 For each of the following languages, give two strings that are members and two strings that are *not* members—a total of four strings for each part. Assume the alphabet $\Sigma = \{a, b\}$ in all parts.

Solution

a. a^*b^*. Members: ϵ, a. Non-members: ba, aba.

b. $a(ba)^*b$. Members: $ab, abab$. Non-members: ϵ, aa.

c. $a^* \cup b^*$. Members: ϵ, a. Non-members: ab, ba.

d. $(aaa)^*$. Members: ϵ, aaa. Non-members: a, aa.

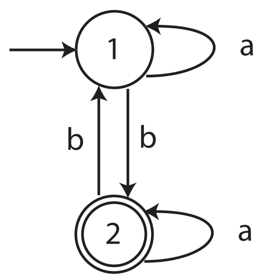
e. $\Sigma^*a\Sigma^*b\Sigma^*a\Sigma^*$. Members: $aba, aaba$. Non-members: ϵ, a.

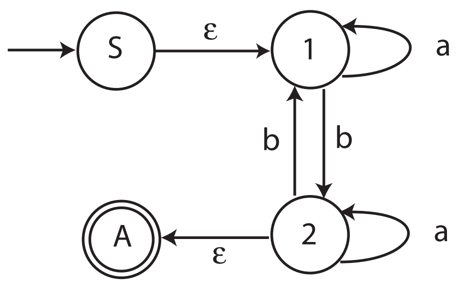
f. $aba \cup bab$. Members: aba, bab. Non-members: ϵ, a.

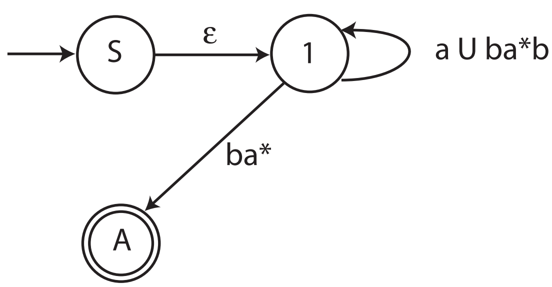
g. $(\epsilon \cup a)b$. Members: b, ab. Non-members: ϵ, a.

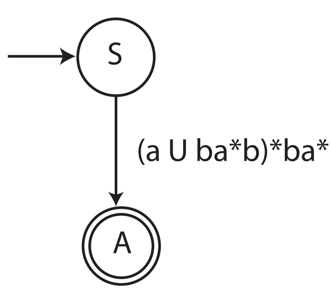
h. $(a \cup ba \cup bb)\Sigma^*$. Members: a, ba. Non-members: ϵ, b.

1.16a.

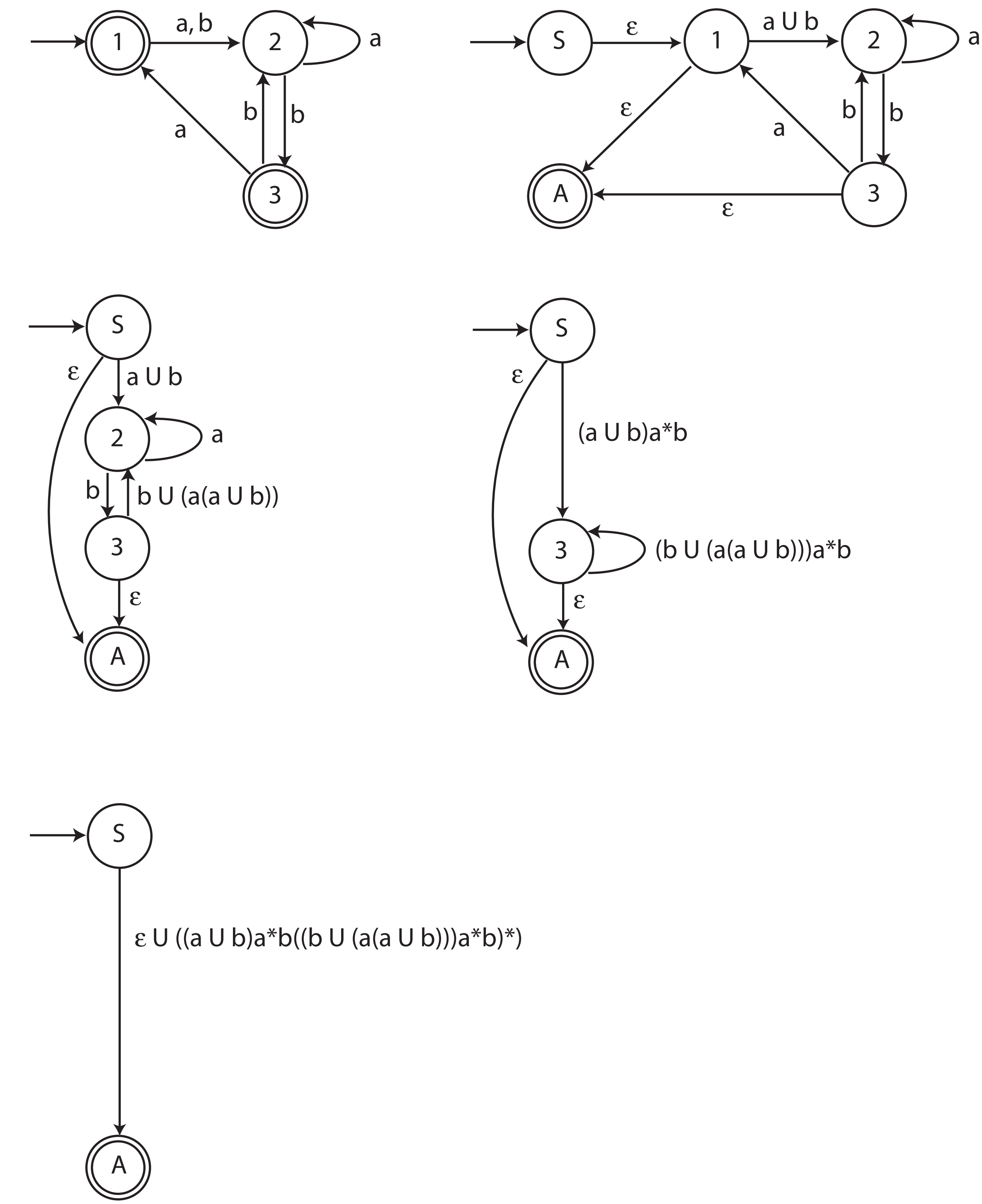








1.16b.


1
a, b
2
a
b
b
a
3
S
ε
1
a U b
2
a
ε
b
b
a
A
3
ε
S
ε
a U b
2
a
b
b U (a(a U b))
3
ε
A
S
ε
(a U b)a*b
3
(b U (a(a U b)))a*b
ε
A
S
ε U ((a U b)a*b((b U (a(a U b)))a*b)*)
A