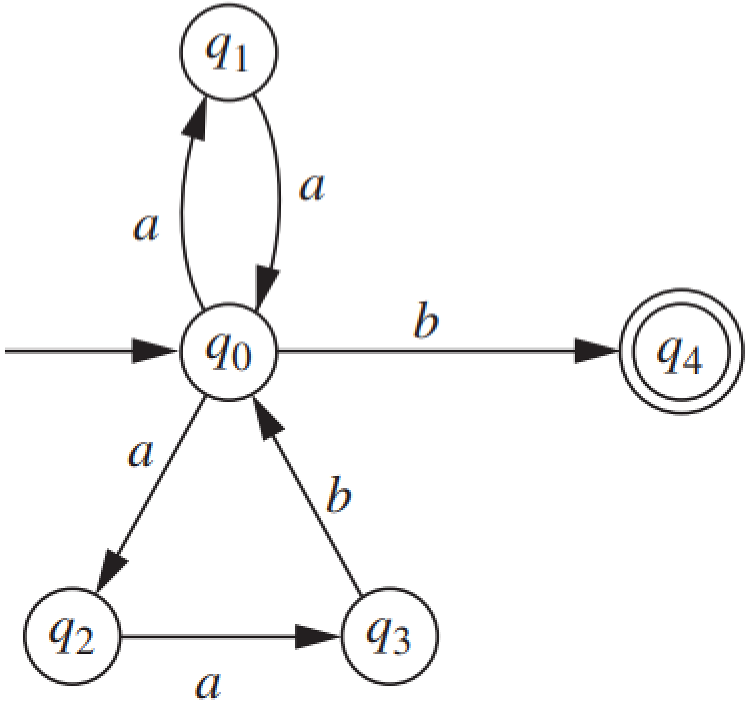
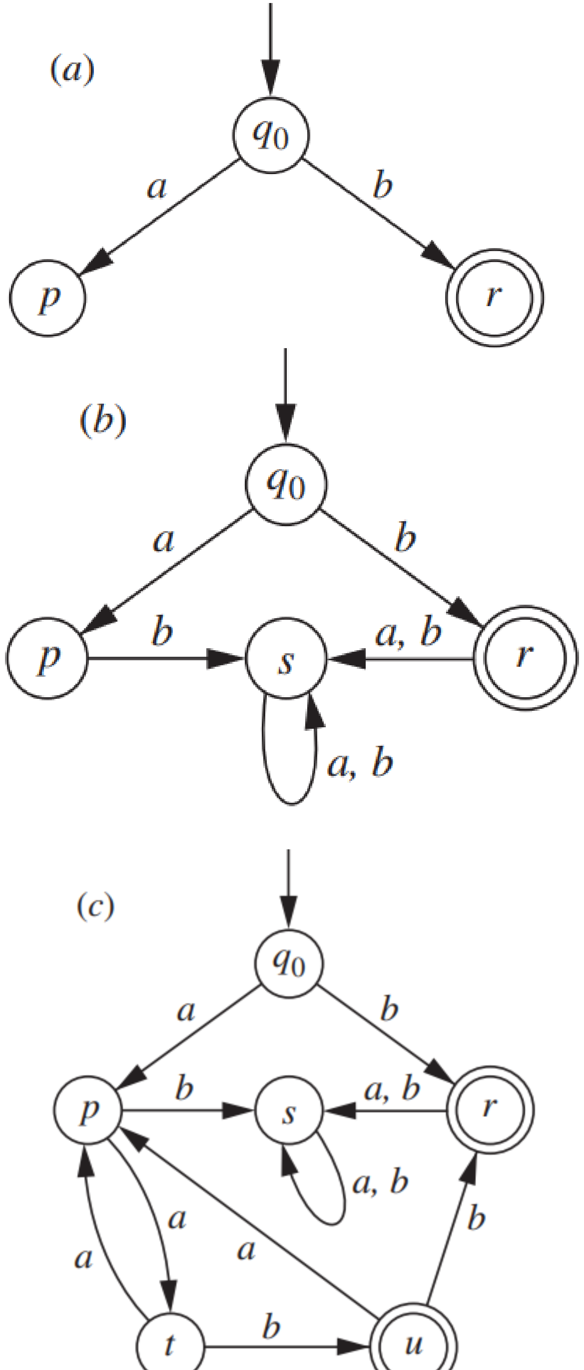
$\{aa, aab\}^*\{b\}$

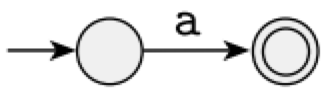
NFA



DFA

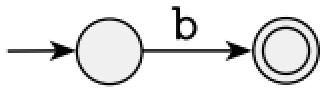


a

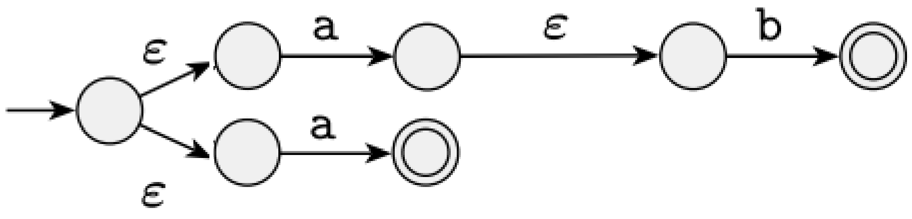


ab

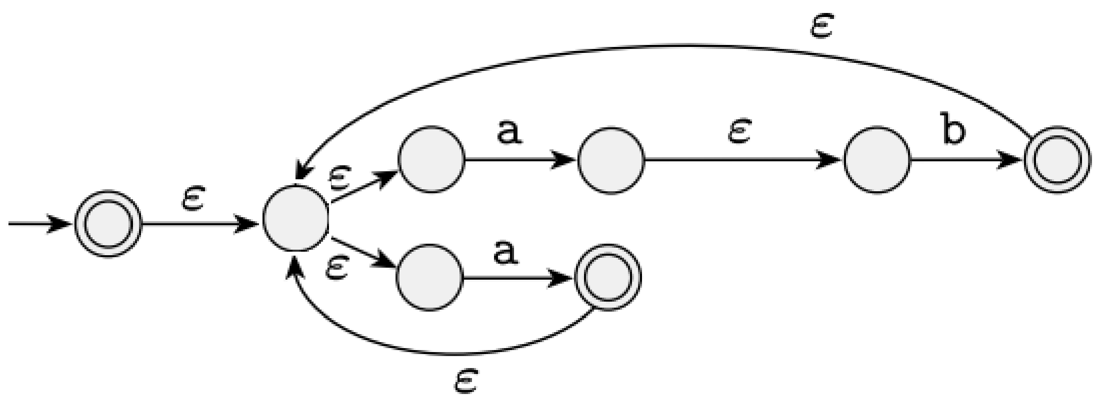
b



ab ∪ a

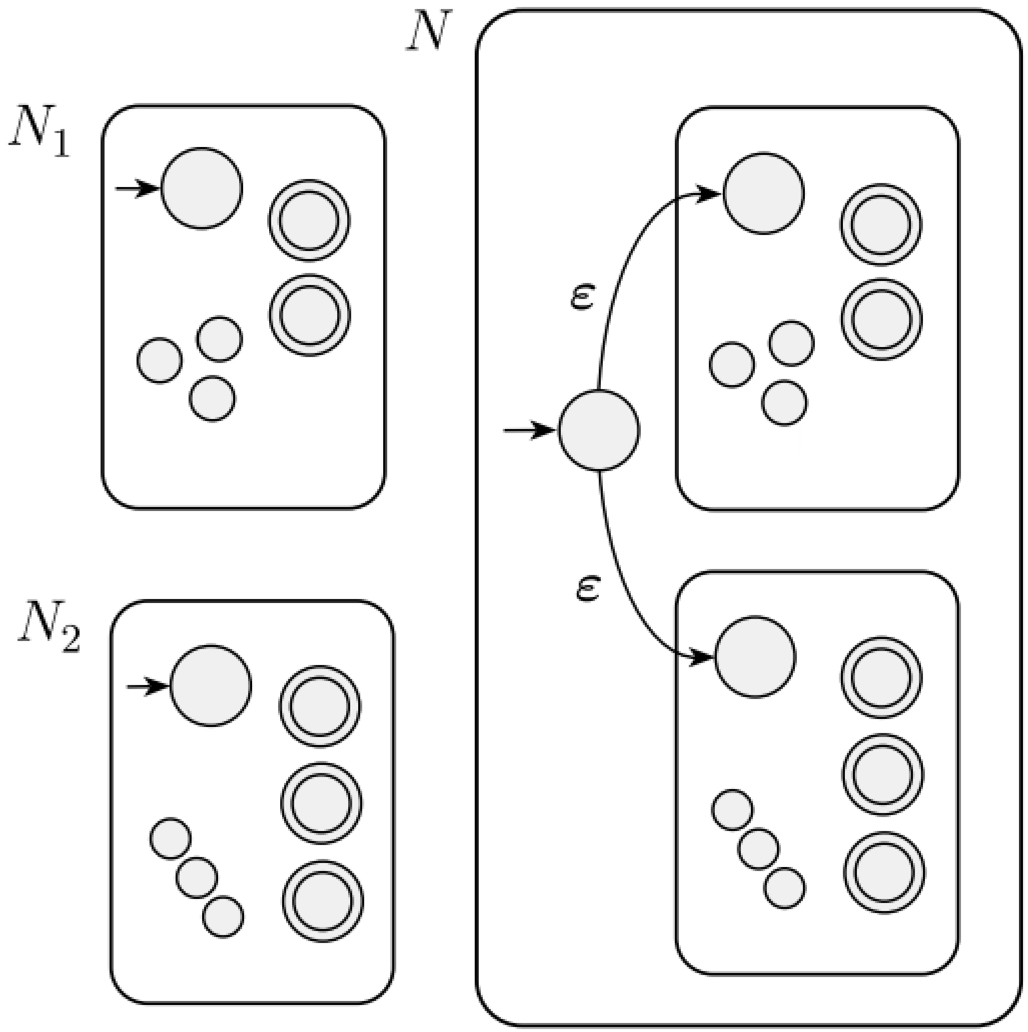


(ab ∪ a)*



$A1 \cup A2$'yi tanımak için bir NFA'nin oluşturulması

A1 ve A2 düzgün dilleri ise A1 ∪ A2 dili de düzenli dildir.



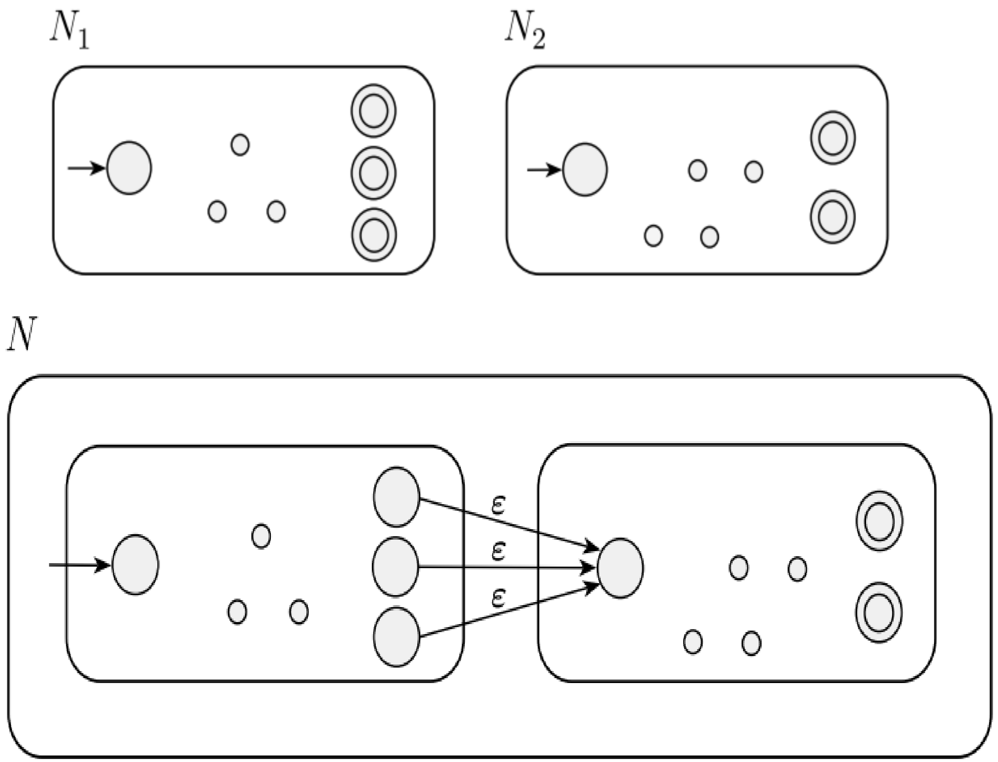
$N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1,
$N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognize A_2.

1. $Q = \{q_0\} \cup Q_1 \cup Q_2$.
2. The state q_0 is the start state of N.
3. $F = F_1 \cup F_2$.

$q \in Q$ and any $a \in \Sigma$

4. $$\delta(q,a) = \begin{cases} \delta_1(q,a) & q \in Q_1 \\ \delta_2(q,a) & q \in Q_2 \\ \{q_1, q_2\} & q = q_0 \text{ and } a = \varepsilon \\ \emptyset & q = q_0 \text{ and } a \neq \varepsilon. \end{cases}$$

A1 ve A2 düzgün dilleri ise A1 . A2 dili de düzenli dildir.



4. $q \in Q$ and any $a \in \Sigma_\varepsilon$,

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & q \in Q_1 \text{ and } q \notin F_1 \\ \delta_1(q,a) & q \in F_1 \text{ and } a \neq \varepsilon \\ \delta_1(q,a) \cup \{q_2\} & q \in F_1 \text{ and } a = \varepsilon \\ \delta_2(q,a) & q \in Q_2. \end{cases}$$

$N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1,
$N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognize A_2.

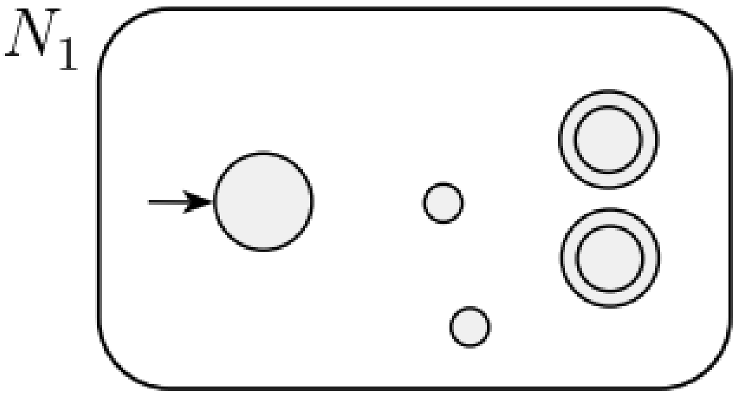
$N = (Q, \Sigma, \delta, q_1, F_2)$ to recognize $A_1 \circ A_2$.

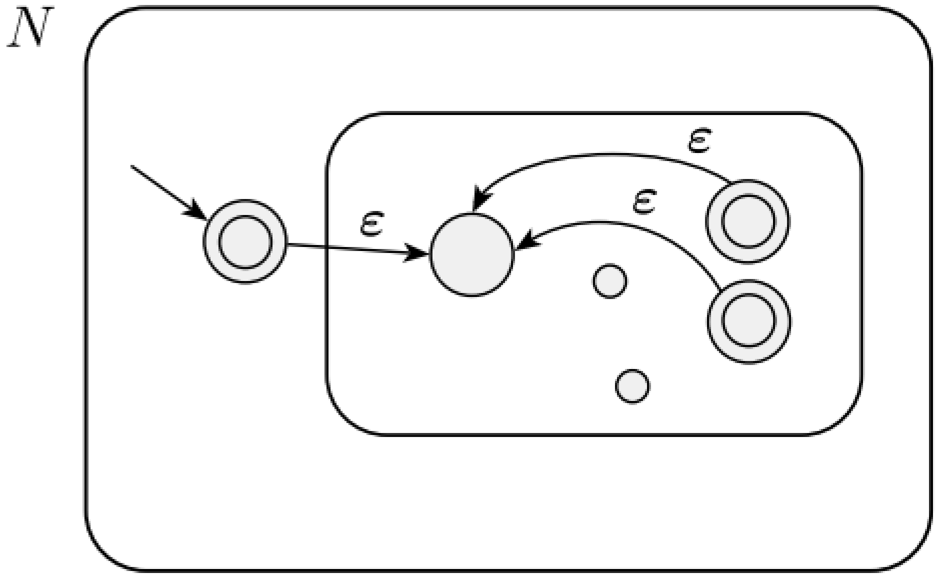
1. $Q = Q_1 \cup Q_2$.

2. The state q_1

3. The accept states F_2

A1 düzgün dil ise A1* dili de düzenli dildir.





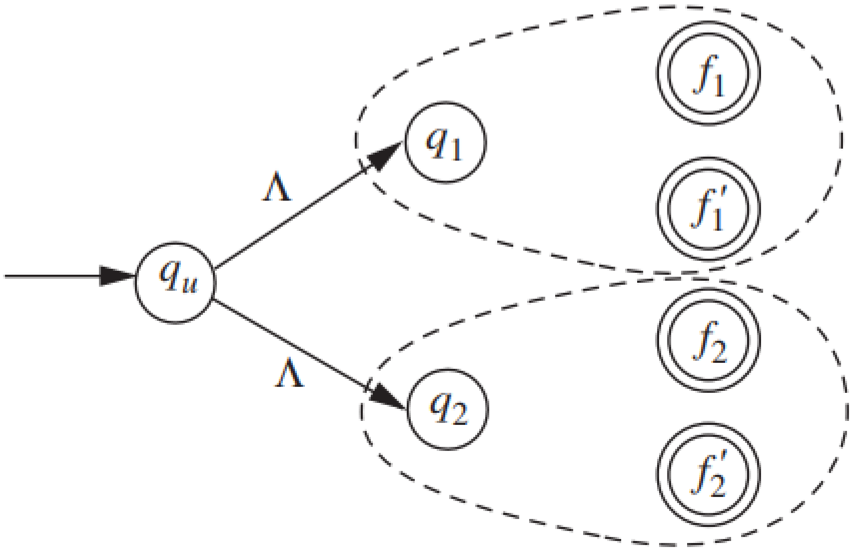
$N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1.

$N = (Q, \Sigma, \delta, q_0, F)$ to recognize A_1^*.

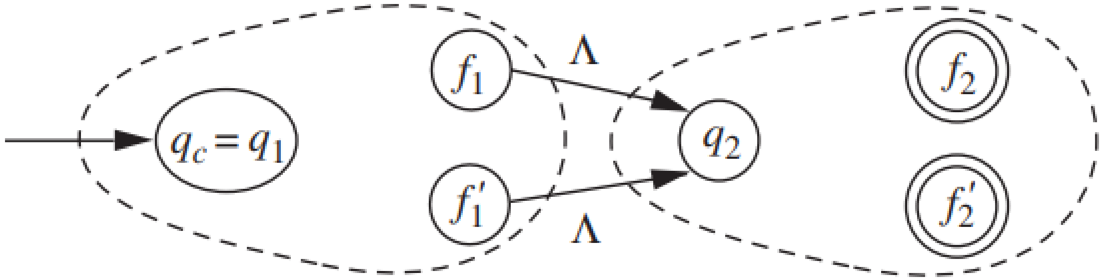
1. $Q = \{q_0\} \cup Q_1$.

2. The state q_0 is the new start state.

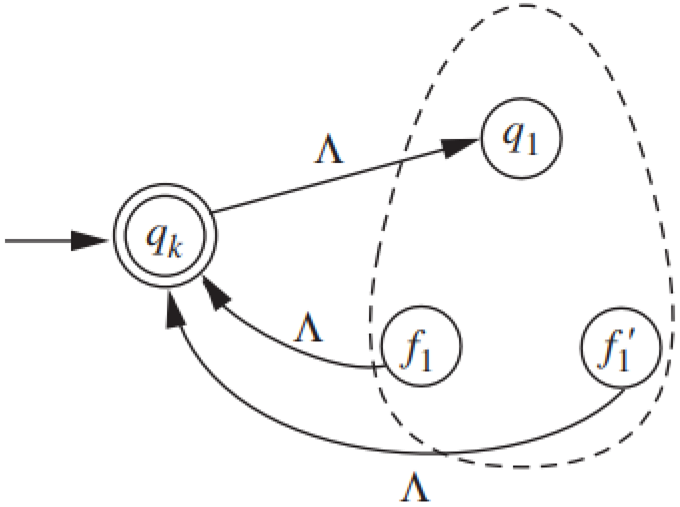
3. $F = \{q_0\} \cup F_1$.



VEYA



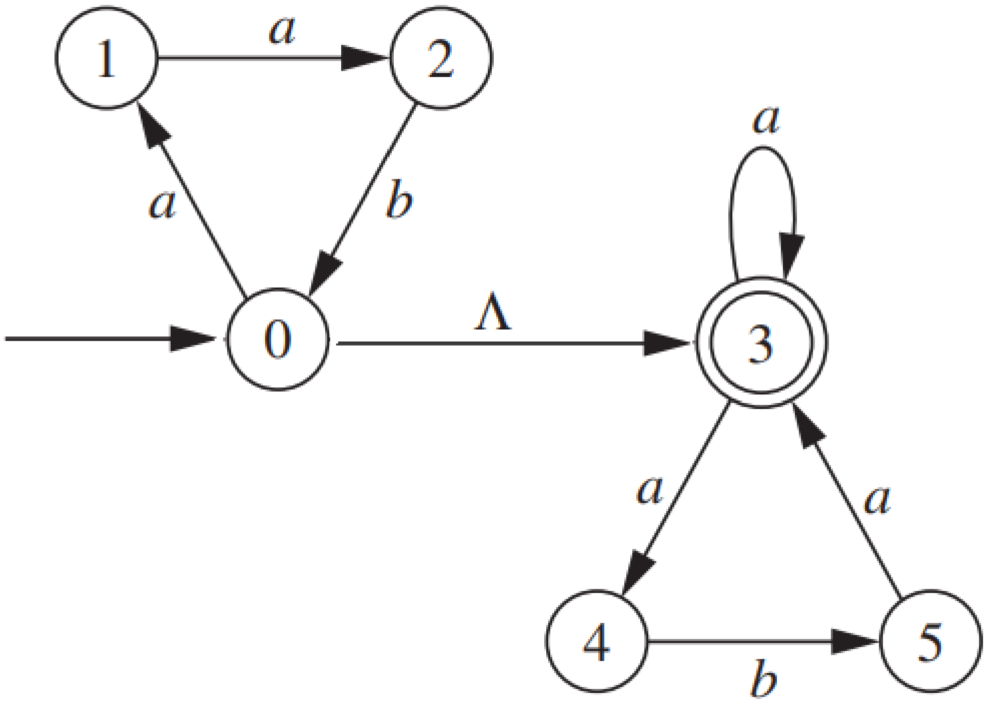
Bitiştirme

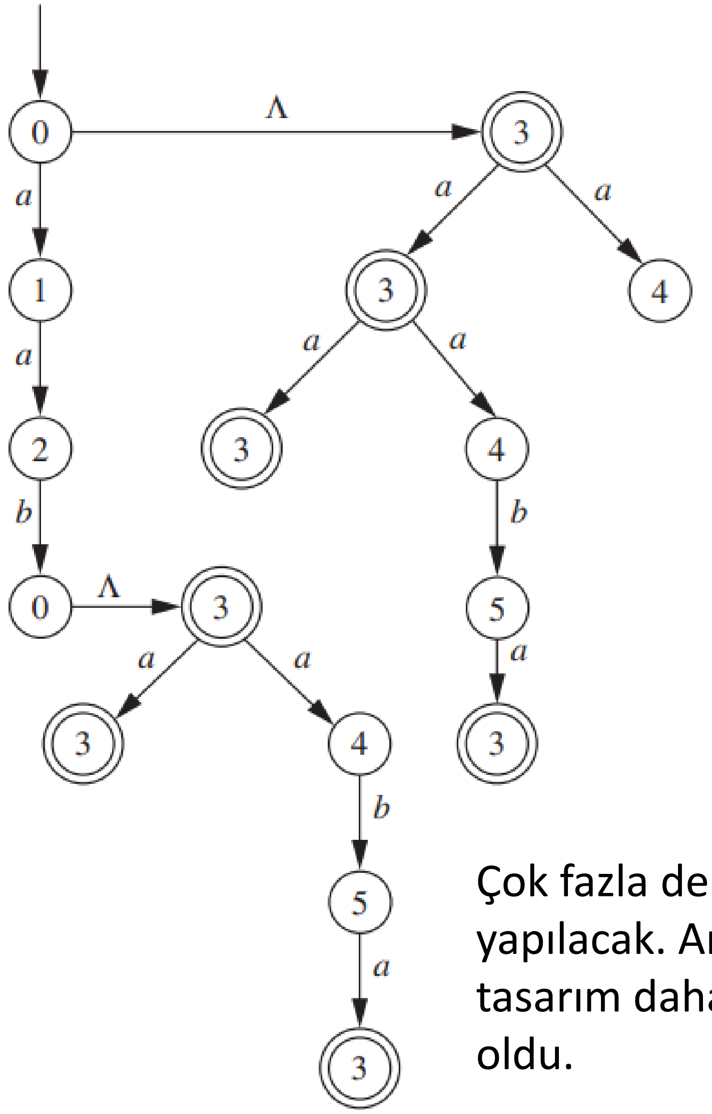


Yıldız kapanma

$\{aab\}^*\{a, aba\}^*$

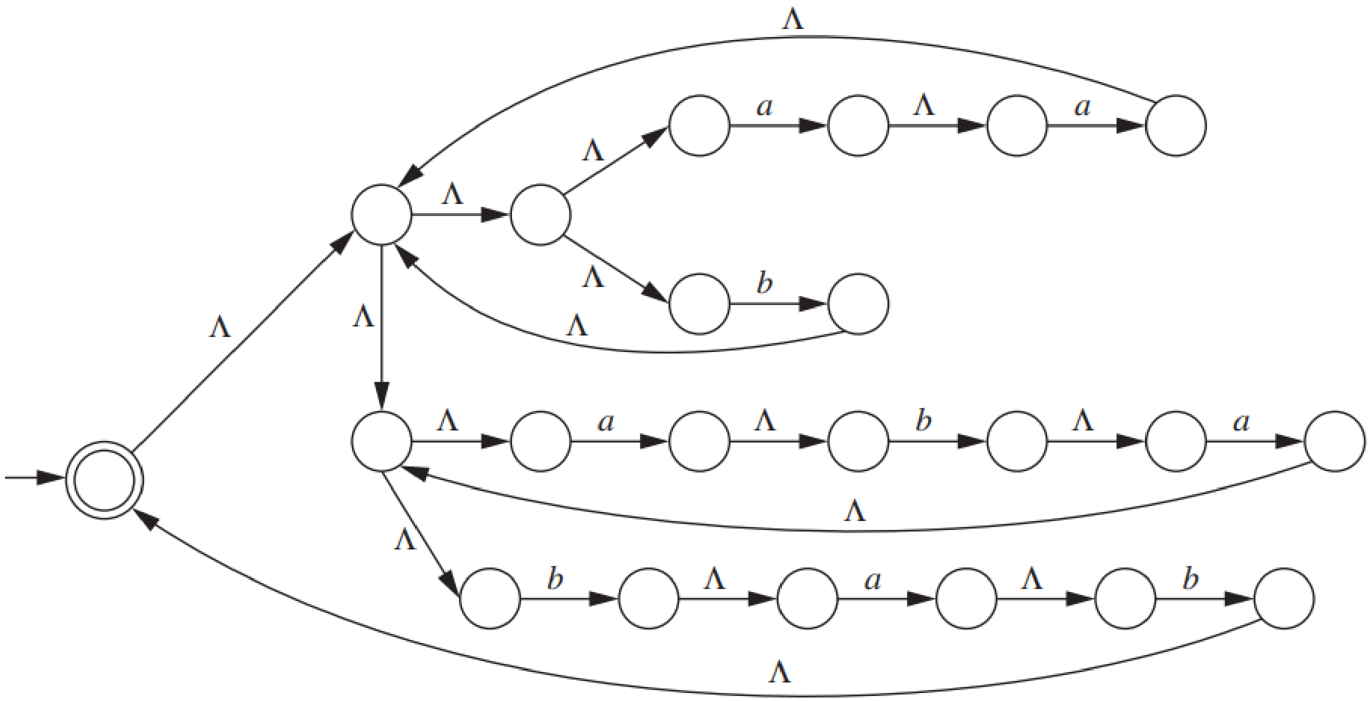
W=aababa

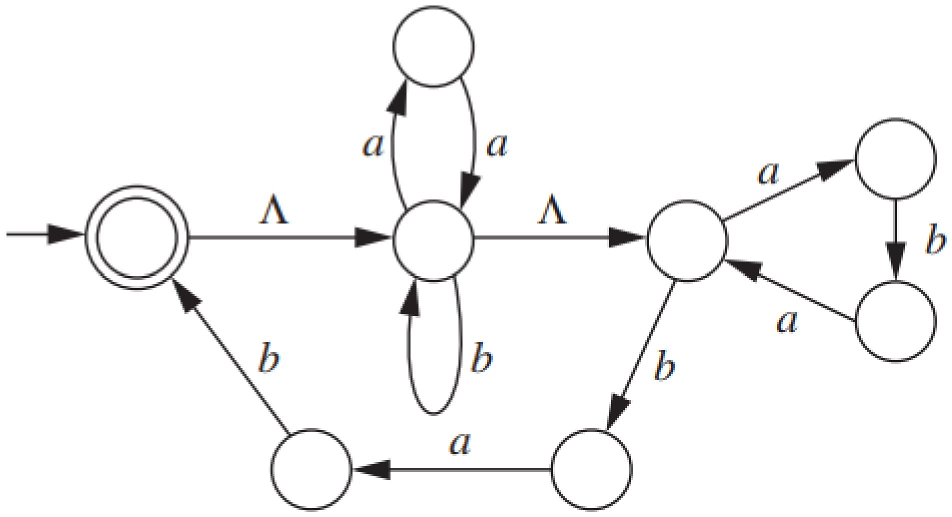




Çok fazla deneme yapılacak. Ancak tasarım daha kolay oldu.

$$((aa+b)^*(aba)^*bab)^*$$





A *nondeterministic finite automaton* (NFA) $(Q, \Sigma, q_0, A, \delta)$,

Q is a finite set of states;

Σ is a finite input alphabet;

$q_0 \in Q$ is the initial state;

$A \subseteq Q$ is the set of accepting states;

$\delta : Q \times (\Sigma \cup \{\Lambda\}) \rightarrow 2^Q$ is the transition function

The Λ-Closure of a Set of States

$M = (Q, \Sigma, q_0, A, \delta)$ is an NFA, and $S \subseteq Q$ is a set of states.
The Λ-closure of S is the set $\Lambda(S)$ that can be defined recursively as follows.

1. $S \subseteq \Lambda(S)$.
2. For every $q \in \Lambda(S)$, $\delta(q, \Lambda) \subseteq \Lambda(S)$.

Λ (S)’in hesaplanması

Bir durum, eğer Λ (S)'nin bir elemanı ise veya bir veya daha fazla geçiş kullanılarak S'nin bir elemanından ulaşılabiliyorsa, (S)'dedir.

T kümesi, S olacak şekilde başlatılr. Kümenin kendisi boşluk kapanma kümesine aittir. .
Her q ∈ T'yi dikkate alarak ve δ(q,Λ)’ ulaşılan durumlar, eğer o ana kadar T kümesinde yoksa T’ye eklenir.
T’ artık değişmeyinceye kadar bu işlemlere devam edilir.
T'nin son değeri Λ (S)'dir.

Let $M = (Q, \Sigma, q_0, A, \delta)$ be an NFA. We define the extended transition function

$$\delta^* : Q \times \Sigma^* \to 2^Q$$

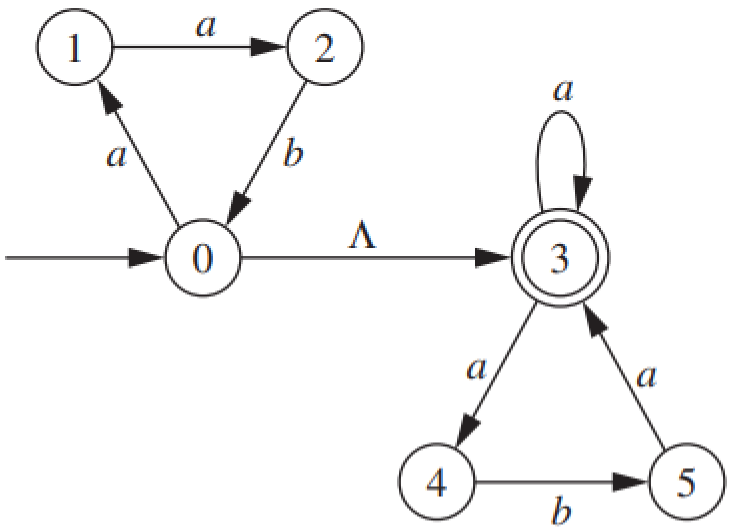
as follows:

1. For every $q \in Q$, $\delta^*(q, \Lambda) = \Lambda(\{q\})$.
2. For every $q \in Q$, every $y \in \Sigma^*$, and every $\sigma \in \Sigma$,

$$\delta^*(q, y\sigma) = \Lambda\left(\bigcup\{\delta(p, \sigma) \mid p \in \delta^*(q, y)\}\right)$$

A string $x \in \Sigma^*$ is accepted by M if $\delta^*(q_0, x) \cap A \neq \emptyset$. The language $L(M)$ accepted by M is the set of all strings accepted by M.

Aşağıdaki NFA-Λ makinesinde **‘aab’** katarını tarayınız.



Λ {0}={0,3}

Λ {0,a}=Λ(δ(0,a) U δ(3,a))=Λ ({1} U{3,4})={1,3,4}

Λ {0,aa}=Λ(δ(1,a)U δ(3,a) U δ(4,a))=Λ ({2} U {3,4} U Ø)={2,3,4}

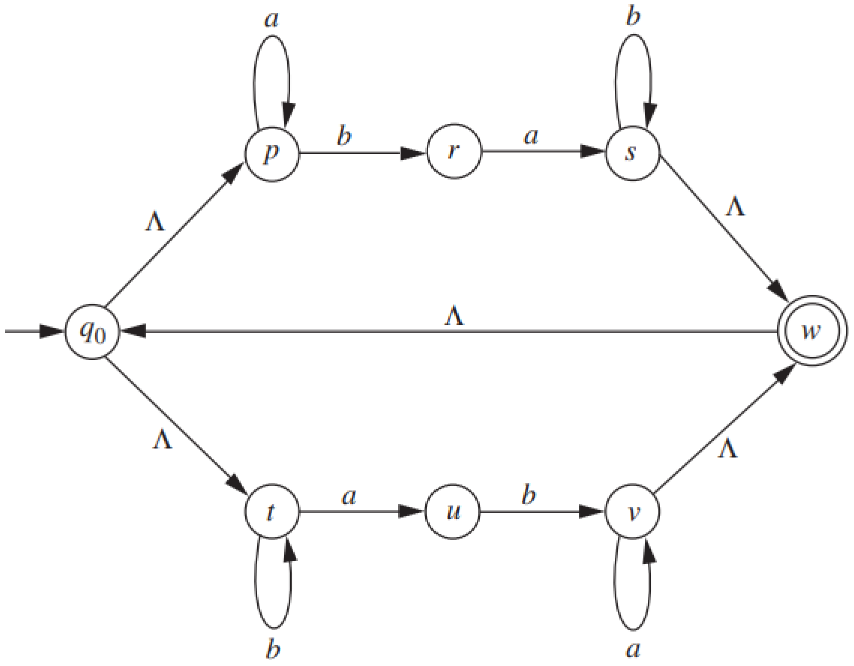
$$\delta^*(0, aa) = \{2, 3, 4\}.$$

$$\delta^*(0, aab)$$

$$\bigcup\{\delta(p, b) \mid p \in \{2, 3, 4\}\} = \delta(2, b) \cup \delta(3, b) \cup \delta(4, b) = \{0\} \cup \emptyset \cup \{5\}$$
$$= \{0, 5\}$$

Λ {0,5}={0,5, 3} Kabul durumu (3 nolu durum) içerdiğinden KABUL edilir.

ÖRNEK:Aşağıdaki NFA-Λ makinesinde **‘aba’** katarını tarayınız.



F={w}

$$\delta^*(q_0, \Lambda) = \Lambda(\{q_0\})$$
$$= \{q_0, p, t\}$$

$$\delta^*(q_0, a) = \Lambda\left(\bigcup \{\delta(k, a) \mid k \in \delta^*(q_0, \Lambda)\}\right)$$
$$= \Lambda\left(\delta(q_0, a) \cup \delta(p, a) \cup \delta(t, a)\right)$$
$$= \Lambda\left(\emptyset \cup \{p\} \cup \{u\}\right)$$
$$= \Lambda(\{p, u\})$$
$$= \{p, u\}$$

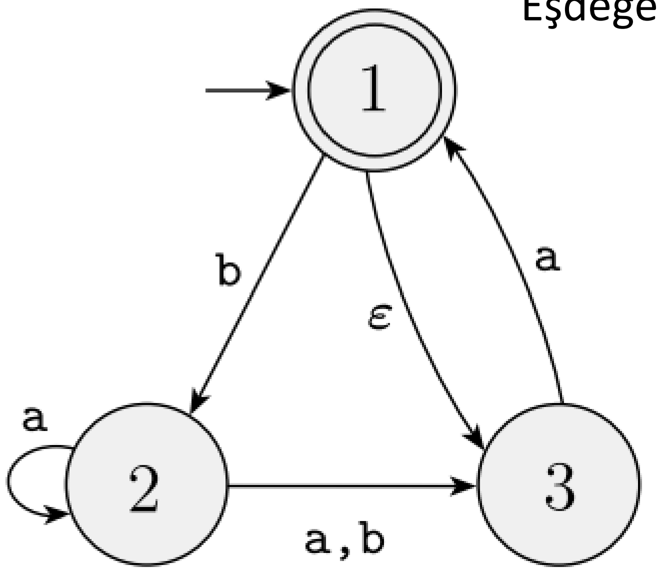
$$\delta^*(q_0, ab) = \Lambda\left(\bigcup \{\delta(k, b) \mid k \in \{p, u\}\}\right)$$
$$= \Lambda(\delta(p, b) \cup \delta(u, b))$$
$$= \Lambda(\{r, v\})$$
$$= \{r, v, w, q_0, p, t\}$$

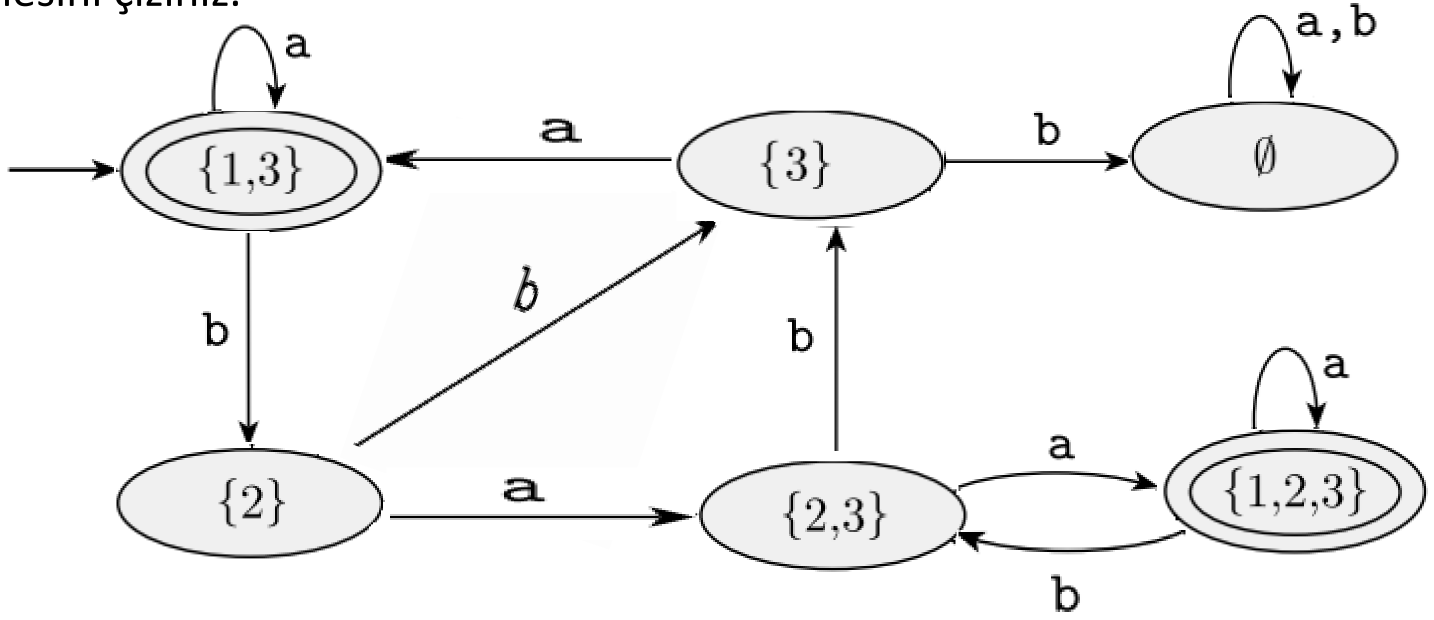
$$\delta^*(q_0, aba) = \Lambda\left(\bigcup \{\delta(k, a) \mid k \in \{r, v, w, q_0, p, t\}\}\right)$$
$$= \Lambda(\delta(r, a) \cup \delta(v, a) \cup \delta(w, a) \cup \delta(q_0, a) \cup \delta(p, a) \cup \delta(t, a))$$
$$= \Lambda(\{s\} \cup \{v\} \cup \emptyset \cup \emptyset \cup \{p\} \cup \{u\})$$
$$= \Lambda(\{s, v, p, u\})$$

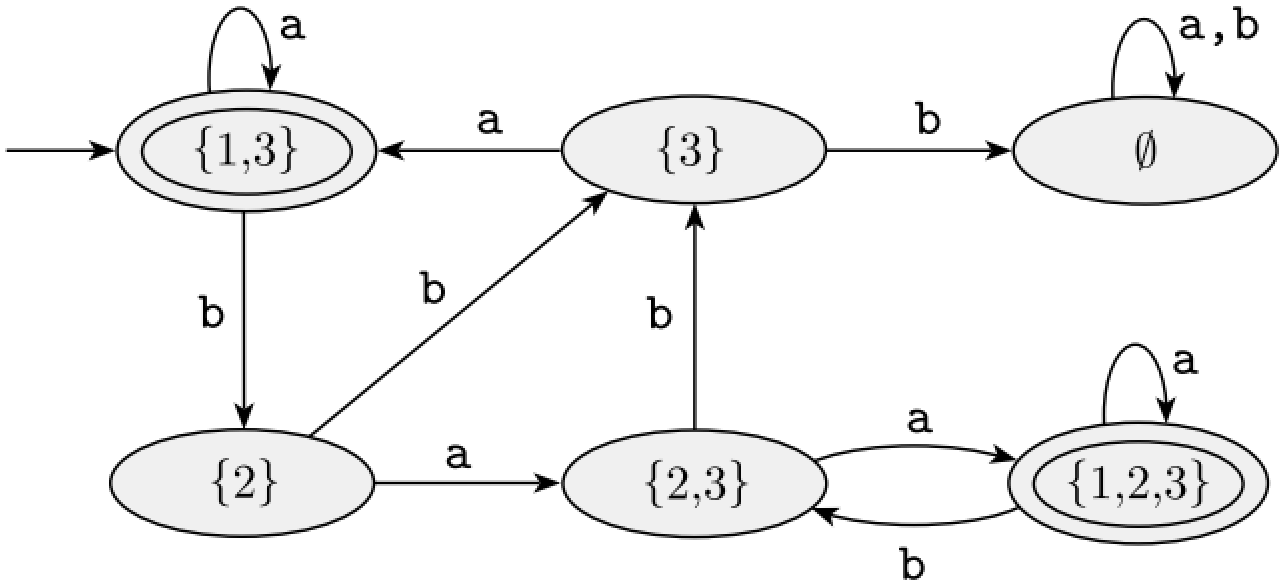
$$= \{s, v, p, u, w, q_0, t\}$$

Sonuç kümede «w» durumu olduğu için katar kabul edilir.

Eşdeğer NFA makinesini çiziniz.

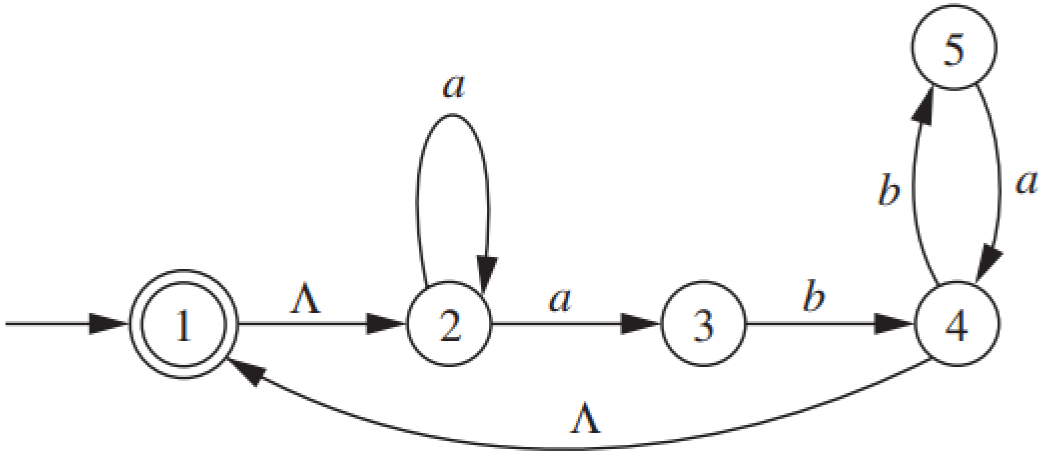


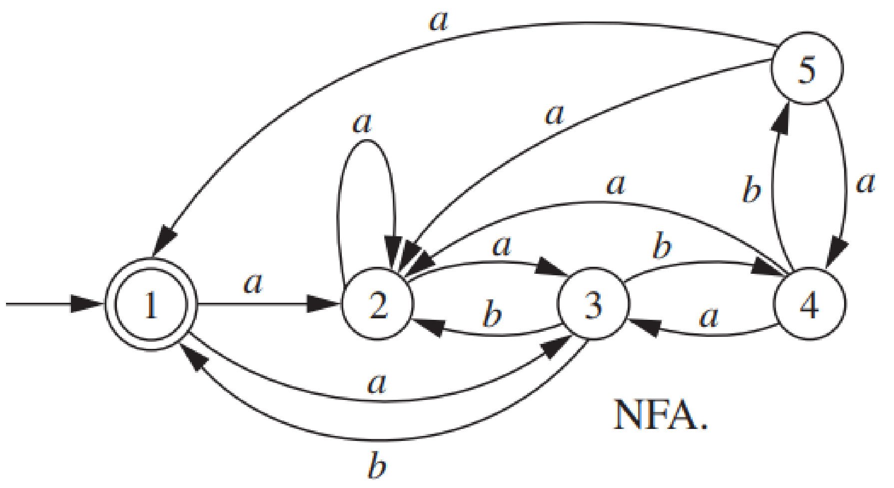




$(a^*ab\,(ba)^*)^*$

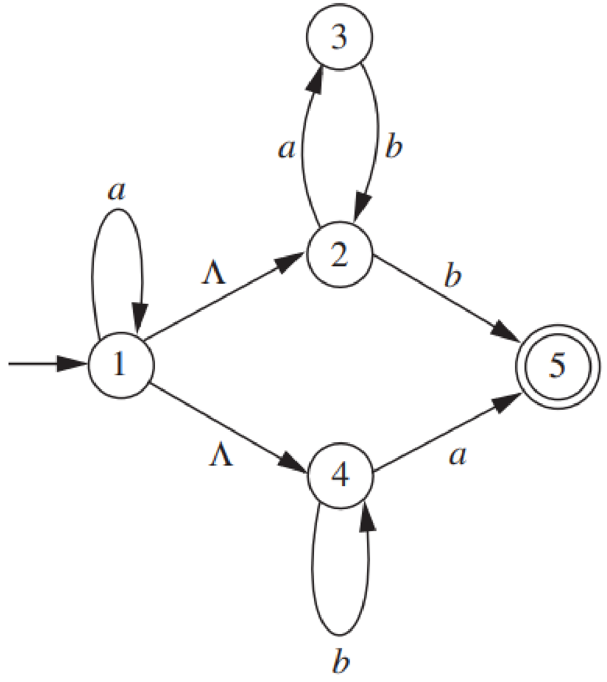
Regüler ifadesi için NFA-Λ makinesini çiziniz ve eşdeğer NFA makinesini çiziniz.

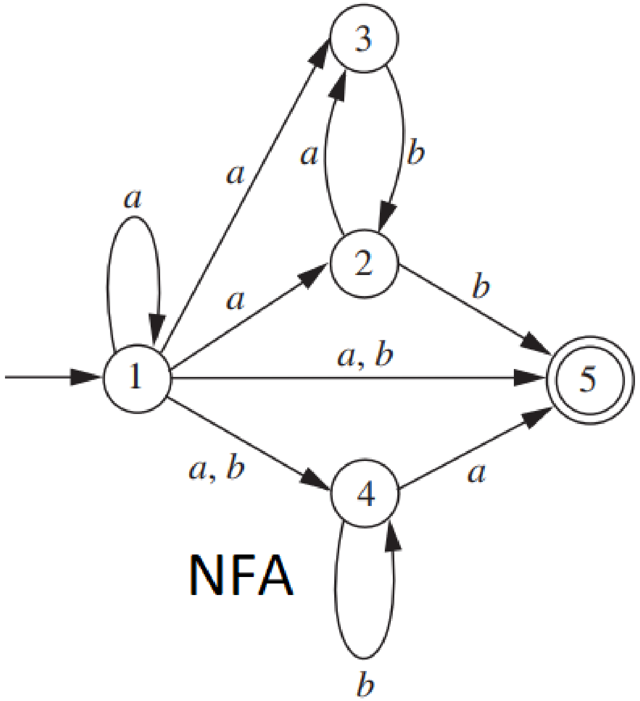




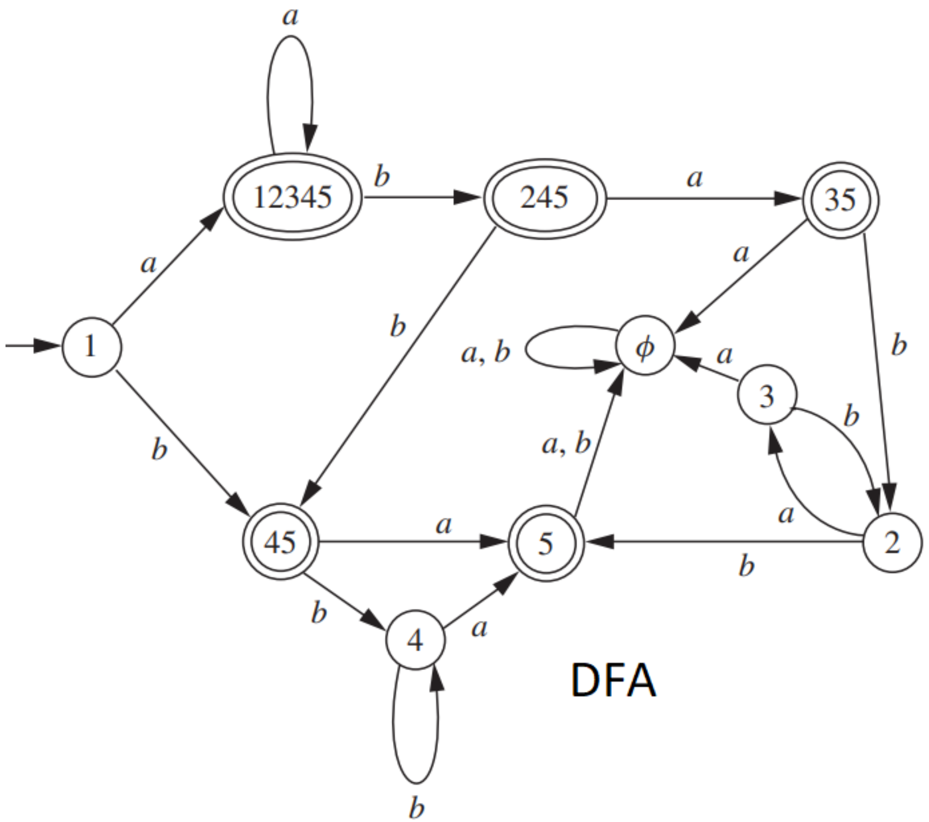
NFA.

q	$\delta(q, a)$	$\delta(q, b)$	$\delta(q, \Lambda)$	$\delta^*(q, a)$	$\delta^*(q, b)$
1	{1}	∅	{2, 4}	{2, 3}	∅
2	{3}	{5}	∅	{2, 3}	∅
3	∅	{2}	∅	∅	{1, 2, 4}
4	{5}	{4}	∅	{2, 3}	{5}
5	∅	∅	∅	{1, 2, 4}	∅





NFA



DFA

q	$\delta(q, a)$	$\delta(q, b)$	$\delta(q, \Lambda)$	$\delta^*(q, a)$	$\delta^*(q, b)$
1	$\{1\}$	$\emptyset$	$\{2, 4\}$	$\{1, 2, 3, 4, 5\}$	$\{4, 5\}$
2	$\{3\}$	$\{5\}$	$\emptyset$	$\{3\}$	$\{5\}$
3	$\emptyset$	$\{2\}$	$\emptyset$	$\emptyset$	$\{2\}$
4	$\{5\}$	$\{4\}$	$\emptyset$	$\{5\}$	$\{4\}$
5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

DURUM SAYISI İNDİRGEME