Véges automaták és reguláris nyelvtanok ekvivalenciája

A nemdeterminisztikus véges automaták ugyanazt a nyelvosztályt ismerik fel, mint a determinisztikus véges automaták. A következő két tétel azt mutatja, hogy ez a nyelvosztály nem más, mint a reguláris nyelvek osztálya.

Tétel. **Ha L egy tetszőleges determinisztikus véges automata által felismert nyelv, akkor megadható egy olyan reguláris nyelvtan, amelyik az L nyelvet generálja.**

$\mathrm{A} = (Q, \Sigma, E, \{q_0\}, F)$ **az L nyelvet felismerő determinisztikus véges automata, azaz $L = L(\mathrm{A})$.**

Értelmezzük a $G = (Q, \Sigma, P, q_0)$ reguláris nyelvtant a következő szabályokkal:

- **Ha $(p, a, q) \in E$, $p, q \in Q$, $a \in \Sigma$ $\quad\Longrightarrow p \to aq \in P$**
- **Ha $(p, a, q) \in E$ és $q \in F$ $\quad\Longrightarrow p \to a \in P$ is**

Bebizonyítjuk, hogy $L(G) = L(\mathrm{A}) \setminus \{\varepsilon\}$.

Legyen $u = a_1 a_2 \ldots a_n \in L(\mathrm{A})$ és $u \neq \varepsilon$. Ekkor, mivel az A véges automata felismeri az u szót, létezik a

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} q_{n-1} \xrightarrow{a_n} q_n, \quad q_n \in F$$

séta. Ekkor P-ben léteznek a következő szabályok:

$$q_0 \to a_1 q_1, \quad q_1 \to a_2 q_2, \quad \ldots, \quad q_{n-2} \to a_{n-1} q_{n-1}, \quad q_{n-1} \to a_n$$

(utóbbi szabály jobb oldalán nem szerepel q_n, mivel $q_n \in F$), tehát létezik a

$$q_0 \Longrightarrow a_1 q_1 \Longrightarrow a_1 a_2 q_2 \Longrightarrow \ldots \Longrightarrow a_1 a_2 \ldots a_{n-1} q_{n-1} \Longrightarrow a_1 a_2 \ldots a_n$$

levezetés. Ezért $u \in L(G)$.

Fordítva, legyen $u = a_1a_2 \ldots a_n \in L(G)$, és $u \neq \varepsilon$. Ekkor létezik a

$$q_0 \Longrightarrow a_1q_1 \Longrightarrow a_1a_2q_2 \Longrightarrow \ldots \Longrightarrow a_1a_2 \ldots a_{n-1}q_{n-1} \Longrightarrow a_1a_2 \ldots a_n$$

levezetés, amelyben a

$$q_0 \to a_1q_1, \quad q_1 \to a_2q_2, \quad \ldots, \quad q_{n-2} \to a_{n-1}q_{n-1}, \quad q_{n-1} \to a_n$$

szabályokat használtuk, amelyek értelmezés szerint azt jelentik, hogy az A véges automatában létezik a következő séta:

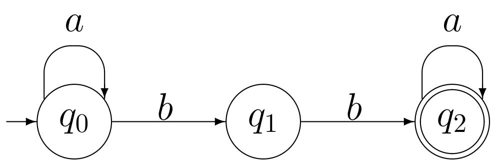
$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \ \cdots \ \xrightarrow{a_{n-1}} q_{n-1} \xrightarrow{a_n} q_n,$$

és mivel q_n végállapot, következik, hogy $u \in L(\mathrm{A}) \setminus \{\varepsilon\}$.

Ha a DVA ε-t is felismeri, akkor bevezetünk egy új q_0' kezdőszimbólumot q_0 helyett, bevesszük a szabályok közé a $q_0' \to \varepsilon$ szabályt, majd minden $q_0 \to \alpha$ szabály mellé bevesszük a $q_0' \to \alpha$ szabályt is.

Példa. $\mathrm{A} = (\{q_0, q_1, q_2\}, \{a, b\}, E, \{q_0\}, \{q_2\})$ determinisztikus véges automata, ahol $E = \big\{(q_0, a, q_0),\ (q_0, b, q_1), (q_1, b, q_2), (q_2, a, q_2)\big\}$.

δ	a	b
q_0	$\{q_0\}$	$\{q_1\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_2\}$	$\emptyset$



A tétel alapján: $G = (\{q_0, q_1, q_2\}, \{a, b\}, P, q_0)$

P szabályai a következők:
$q_0 \to aq_0 \mid bq_1$, $\quad q_1 \to bq_2 \mid b$, $\quad q_2 \to aq_2 \mid a$.

Vagy más jelöléssel:

$G = (\{S, A, B\}, \{a, b\}, P, S)$

P szabályai a következők:

$S \to aS \mid bA$
$A \to bB \mid b$
$B \to aB \mid a$.

Igazolható, hogy $L(\mathrm{A}) = \{a^m bba^n \mid m \geq 0, n \geq 0\}$.

A tétel bizonyításában megadott módszert könnyen átírhatjuk algoritmussá. Az $\mathrm{A} = (Q, \Sigma, E, \{q_0\}, F)$ determinisztikus automatából kapott $G = (Q, \Sigma, P, q_0)$ reguláris nyelvtan szabályait a következő algoritmussal határozzuk meg.

Automatából-reguláris-nyelvtan(A)

```
 1  P ← ∅
 2  for minden p ∈ Q
 3     do for minden a ∈ Σ
 4           do for minden q ∈ Q
 5                 do if (p, a, q) ∈ E
 6                       then P ← P ∪ {p → aq}
 7                            if q ∈ F
 8                              then P ← P ∪ {p → a}
 9  if q_0 ∈ F
10    then P ← P ∪ {q_0 → ε}
11  return G
```

Amennyiben az automata felismeri az üres szót is, a fenti algoritmus esetleg kiterjesztett reguláris nyelvet generál.

Könnyű belátni, hogy az algoritmus futási ideje $\Theta(n^2 m)$, ha az állapotok száma n és a betűk száma m. A 2–4. sorokban lévő három ciklus helyett lehet csupán egyet venni, ha az E elemeit vizsgáljuk, ekkor a futási idő legrosszabb esetben $\Theta(p)$, ahol p az átmenetek száma. Ez szintén $O(n^2m)$, mivel lehetséges, hogy minden átmenet jelen van. Ekkor az algoritmus a következőképpen írható le:

Automatából-reguláris-nyelvtan'(A)

$P \leftarrow \emptyset$
for minden $(p, a, q) \in E$
    do $P \leftarrow P \cup \{p \rightarrow aq\}$
        if $q \in F$
            then $P \leftarrow P \cup \{p \rightarrow a\}$
if $q_0 \in F$
    then $P \leftarrow P \cup \{q_0 \rightarrow \varepsilon\}$
return G

Tétel. Ha $L = L(G)$ reguláris nyelv, akkor megadható egy olyan nemdeterminisztikus véges automata, amely felismeri az L nyelvet.

Legyen $G = (N, T, P, S)$ az L nyelvet generáló reguláris nyelvtan. Definiáljuk az $\mathrm{A} = (Q, T, E, \{S\}, F)$ nemdeterminisztikus véges automatát a következőképpen.

- $Q = N \cup \{Z\}$, ahol $Z \notin N \cup T$ (vagyis egy új szimbólum),
- Minden $A \rightarrow aB$ szabályra bevesszük E-be az (A, a, B) átmenetet.
- Minden $A \rightarrow a$ szabályra bevesszük E-be az (A, a, Z) átmenetet.
- $F = \begin{cases} \{Z\} & \text{ha } G\text{-ben nem szerepel az } S \rightarrow \varepsilon \text{ szabály,} \\ \{Z, S\} & \text{ha } G\text{-ben szerepel az } S \rightarrow \varepsilon \text{ szabály.} \end{cases}$

Bebizonyítjuk, hogy $L(G) = L(A)$.

Legyen $u = a_1a_2 \dots a_n \in L(G)$, $u \neq \varepsilon$. Ekkor létezik u-nak egy G-beli levezetése: $S \Longrightarrow a_1A_1 \Longrightarrow a_1a_2A_2 \Longrightarrow \dots \Longrightarrow a_1a_2 \dots a_{n-1}A_{n-1} \Longrightarrow a_1a_2 \dots a_n$.

Ez a levezetés a következő szabályok alapján történt:

$$S \to a_1A_1,\ \ A_1 \to a_2A_2,\ \ \ldots,\ \ A_{n-2} \to a_{n-1}A_{n-1},\ \ A_{n-1} \to a_n.$$

Ekkor az A véges automata átmeneteinek értelmezése alapján létezik az

$$S \xrightarrow{a_1} A_1 \xrightarrow{a_2} A_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} A_{n-1} \xrightarrow{a_n} Z,\ \ Z \in F$$

séta. Ez azt jelenti, hogy $u \in L(\mathrm{A})$. Ha $\varepsilon \in L(G)$, akkor van $S \to \varepsilon$ szabály, de ekkor a kezdőállapot végállapot is, tehát $\varepsilon \in L(A)$. Ezért $L(G) \subseteq L(A)$.

Legyen most $u = a_1a_2\ldots a_n \in L(\mathrm{A})$. Ez azt jelenti, hogy létezik az

$$S \xrightarrow{a_1} A_1 \xrightarrow{a_2} A_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} A_{n-1} \xrightarrow{a_n} Z,\ \ Z \in F$$

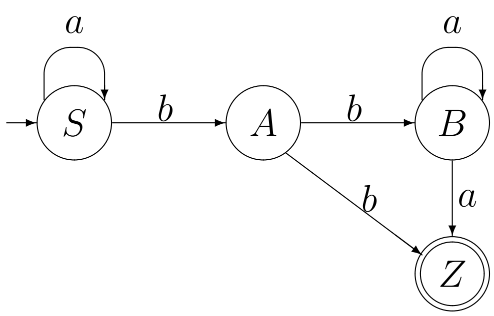
séta. Ha u az üres szó, akkor Z helyett S van, amely szintén végállapot. Más esetben csak Z szerepelhet utolsóként. Tehát G-ben szerepelnek a következő szabályok: $S \to a_1A_1,\ \ A_1 \to a_2A_2,\ \ \ldots,\ \ A_{n-2} \to a_{n-1}A_{n-1},\ \ A_{n-1} \to a_n$, és így létezik az

$$S \Longrightarrow a_1A_1 \Longrightarrow a_1a_2A_2 \Longrightarrow \ldots \Longrightarrow a_1a_2\ldots a_{n-1}A_{n-1} \Longrightarrow a_1a_2\ldots a_n$$

levezetés, tehát $u \in L(G)$, és ekkor $L(A) \subseteq L(G)$.

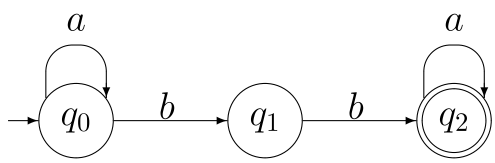
Példa. **Adott a $G = (\{S, A, B\}, \{a, b\},\ \{S \to aS, S \to bA, A \to bB, A \to b,\ B \to aB, B \to a\}, S)$ reguláris nyelvtan. A hozzá rendelt véges automata $A = (\{S, A, B, Z\}, \{a, b\}, E, \{S\}, \{Z\})$, ahol $E = \big\{(S, a, S), (S, b, A), (A, b, B), (A, b, Z), (B, a, B), (B, a, Z)\big\}$. Ennek átmenettáblázata a következő:**

δ	a	b
S	$\{S\}$	$\{A\}$
A	$\emptyset$	$\{B, Z\}$
B	$\{B, Z\}$	$\emptyset$
E	$\emptyset$	$\emptyset$



Ez a véges automata egyszerűsíthető. A B és Z állapotok összevonhatók egyetlen végállapottá.
Hozzárendelünk ehhez az automatához egy vele ekvivalens determinisztikus véges automatát.

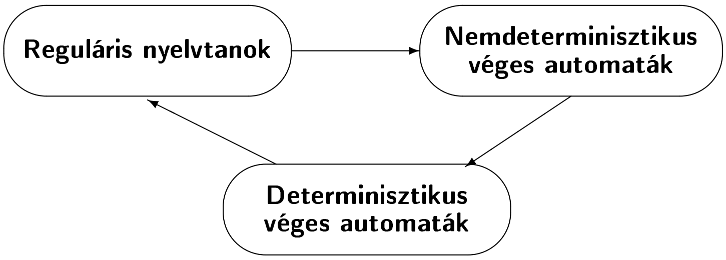
δ	a	b
$q_0 = \{S\}$	$\{q_0\}$	$\{q_1\}$
$q_1 = \{A\}$	$\emptyset$	$\{q_2\}$
$q_2 = \{B, Z\}$	$\{q_2\}$	$\emptyset$



Az előbbi tétel alapján írunk egy algoritmust, amely hozzárendeli a $G = (N, T, P, S)$ reguláris nyelvtanhoz az $\mathrm{A} = (Q, T, E, \{S\}, F)$ véges automatát.

Reguláris-nyelvtanból-automata(G)

$E \leftarrow \emptyset$
$Q \leftarrow N \cup \{Z\}$
for minden $A \in N$
  do for minden $a \in T$
    do if $(A \rightarrow a) \in P$
      then $E \leftarrow E \cup \{(A, a, Z)\}$
     for minden $B \in N$
       do if $(A \rightarrow aB) \in P$
         then $E \leftarrow E \cup \{(A, a, B)\}$
if $(S \rightarrow \varepsilon) \notin P$
  then $F \leftarrow \{Z\}$
  else $F \leftarrow \{Z, S\}$
return A



Akárcsak az AUTOMATÁBÓL-REGULÁRIS-NYELVTAN algoritmus esetében, a futási idő ebben az esetben is $\Theta(n^2m)$, ha a nemterminálisok száma n és a terminálisoké m. Lehetne a 3., 4. és 7. sorokban lévő ciklusokat helyettesíteni eggyel, amelyik a helyettesítési szabályokon megy végig. Ekkor az algoritmus lépésszáma $\Theta(p)$ lesz, ha p a szabályok száma. Az algoritmus a következő:

REGULÁRIS-NYELVTANBÓL-AUTOMATA'(G)

1 $E \leftarrow \emptyset$
2 $Q \leftarrow N \cup \{Z\}$
3 **for** minden $(A \rightarrow u) \in P$
4   **do if** $u = a$
5   **then** $E \leftarrow E \cup \{(A, a, Z)\}$
6   **if** $u = aB$
7   **then** $E \leftarrow E \cup \{(A, a, B)\}$
8 **if** $(S \rightarrow \varepsilon) \notin P$
9   **then** $F \leftarrow \{Z\}$
10   **else** $F \leftarrow \{Z, S\}$
11 **return** A

Bebizonyítottuk, hogy a reguláris nyelvek osztálya egybeesik mind a determinisztikus véges automaták, mind a nemdeterminisztikus véges automaták által felismert nyelvek osztályával.

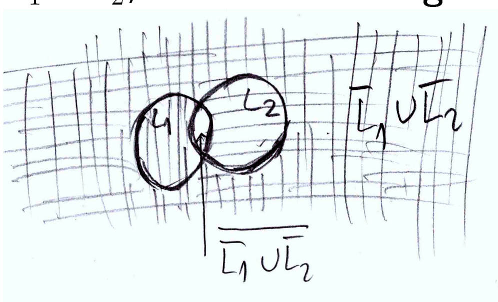
A következő három nyelvosztály megegyezik:
- a reguláris nyelvek osztálya,
- a determinisztikus véges automatákkal felismerhető nyelvek osztálya,
- a nemdeterminisztikus véges automatákkal felismerhető nyelvek osztálya.

Műveletek reguláris nyelvekkel

Ha L_1, L_2 reguláris, akkor regulárisak a következő nyelvek is: $L_1 \cup L_2$, L_1L_2, L_1^*. Ezenkívül a reguláris nyelvekre igazak a következő állítások is.

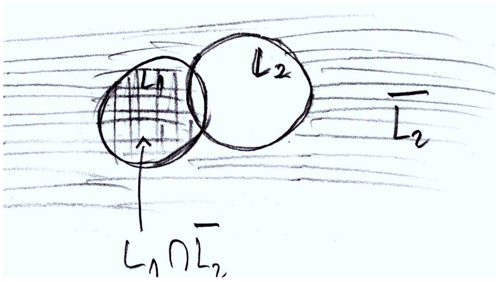
Egy reguláris nyelvnek a komplementuma is reguláris. Ez könnyen igazolható véges automaták segítségével. Legyen ugyanis L egy reguláris nyelv és $\mathrm{A} = (Q, \Sigma, E, \{q_0\}, F)$ egy, az L nyelvet felismerő teljes, determinisztikus véges automata. Könnyen belátható, hogy az $\overline{\mathrm{A}} = (Q, \Sigma, E, \{q_0\}, Q \setminus F)$ automata az $\overline{L}$ nyelvet ismeri fel. Így $\overline{L}$ is reguláris.

Két reguláris nyelvnek a metszete is reguláris. Mivel $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$, a metszet is reguláris.



$\overline{L_1}$ vízszintes vonalakkal satírozott rész
$\overline{L_2}$ függőleges vonalakkal satírozott rész

Két reguláris nyelvnek a különbsége is reguláris. **Mivel** $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$**, a különbség is reguláris.**



$\overline{L_2}$ vízszintes vonalakkal satírozott rész

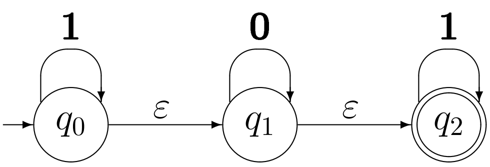
ε-lépéses véges automaták

Az ε**-lépéses véges automata annyiban különbözik a nemdeterminisztikus véges automatától, hogy megengedjük azt, hogy üres lépést is végezzen, azaz átmenjen egyik állapotból a másikba anélkül, hogy valamilyen bemeneti jelet olvasna. Az** ε**-lépéses** $\mathrm{A} = (Q, \Sigma, E, I, F)$ **véges automata átmeneteinek halmazára teljesül, hogy** $E \subseteq Q \times \big(\Sigma \cup \{\varepsilon\}\big) \times Q$.

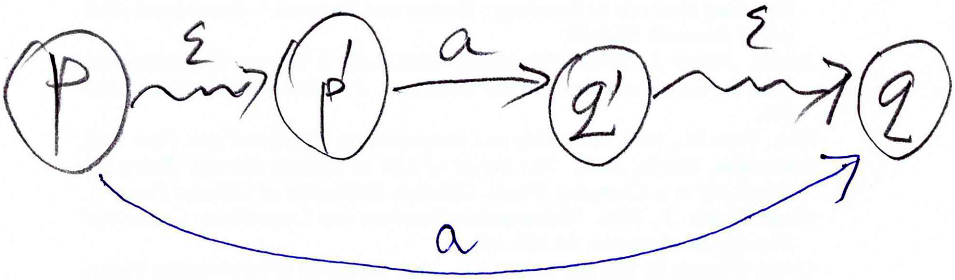
Az ε**-lépéses véges automata átmenetfüggvénye a következő:**

$$\delta : Q \times \big(\Sigma \cup \{\varepsilon\}\big) \to \mathcal{P}(Q), \quad \delta(p, a) = \{q \in Q \mid (p, a, q) \in E\}.$$

Az alábbi ábrán látható ε**-lépéses véges automata az** uvw **alakú szavakat ismeri fel, ahol** $u \in \{1\}^*$, $v \in \{0\}^*$ **és** $w \in \{1\}^*$.



Tétel. **Tetszőleges** ε**-lépéses véges automatához mindig megkonstruálható egy vele ekvivalens nemdeterminisztikus véges automata, amely** ε**-lépés nélküli.**



Az $\mathrm{A} = (Q, \Sigma, E, I, F)$ ε-lépéses véges automatával ekvivalens nemdeterminisztikus véges automata $\overline{\mathrm{A}} = (Q, \Sigma, \overline{E}, I, \overline{F})$ lesz. Algoritmusunk az $\overline{F}$ és az $\overline{E}$ halmazokat határozza meg.

Egy q állapotra jelöljük $\Lambda(q)$-val azon állapotok halmazát, amelyekbe el lehet jutni q-ból csupa ε-lépéssel (beleértve magát q-t is). Terjesszük ki ezt definíciót halmazokra is, azaz legyen

$$\Lambda(S) = \bigcup_{q \in S} \Lambda(q), \quad \forall S \subseteq Q.$$

Nyilvánvaló, hogy minden $q \in Q$-ra és $S \subseteq Q$-ra mind $\Lambda(q)$, mind $\Lambda(S)$ kiszámíthatók. A következőkben feltesszük, hogy ezek adottak.

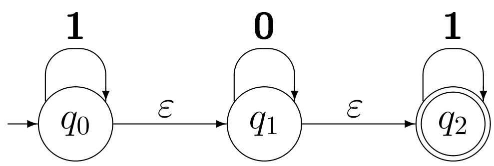
A következő algoritmus az átmenetek meghatározását a $\overline{\delta}$ átmenetfüggvény segítségével végzi, amelyet az algoritmus 5. sorában értelmezünk.

Ha $|Q| = n$ és $|\Sigma| = m$,, akkor a pszeudokód 2–6. soraiból látszik, hogy az algoritmus futási ideje legrosszabb esetben $O(n^2m)$.

EPSZILON-MENTESÍTÉS(A)

$\overline{F} \leftarrow F \cup \{q \in I \mid \Lambda(q) \cap F \neq \emptyset\}$
for **minden** $q \in Q$
  do for **minden** $a \in \Sigma$
    do $\Delta \leftarrow \bigcup_{p \in \Lambda(q)} \delta(p, a)$
    $\overline{\delta}(q, a) \leftarrow \Delta \cup \left(\bigcup_{p \in \Delta} \Lambda(p) \right)$
$\overline{E} \leftarrow \big\{(p, a, q), \mid p, q \in Q,\ a \in \Sigma, q \in \overline{\delta}(p, a)\big\}$
return $\overline{\mathrm{A}}$

Példa.



δ	**0**	**1**	ε
q_0	$\emptyset$	$\{q_0\}$	$\{q_1\}$
q_1	$\{q_1\}$	$\emptyset$	$\{q_2\}$
q_2	$\emptyset$	$\{q_2\}$	$\emptyset$

Alkalmazzuk az EPSZILON-MENTESÍTÉS **algoritmust.**

δ	0	1	ε	Λ
q_0	$\emptyset$	$\{q_0\}$	$\{q_1\}$	$\{q_0, q_1, q_2\}$
q_1	$\{q_1\}$	$\emptyset$	$\{q_2\}$	$\{q_1, q_2\}$
q_2	$\emptyset$	$\{q_2\}$	$\emptyset$	$\{q_2\}$

$\overline{\delta}$	0	1
q_0	$\{q_1, q_2\}$	$\{q_0, q_1, q_2\}$
q_1	$\{q_1, q_2\}$	$\{q_2\}$
q_2	$\emptyset$	$\{q_2\}$

Bővebben:

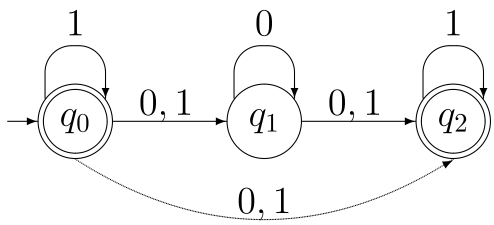
$\Lambda(q_0) = \{q_0, q_1, q_2\}$, $\Lambda(q_1) = \{q_1, q_2\}$, $\Lambda(q_2) = \{q_2\}$
$\Lambda(I) = \Lambda(q_0)$, **és ennek metszete az** F**-fel nem üres, ezért** $\overline{F} = F \cup \{q_0\} = \{q_0, q_2\}$.

$(q_0, 0):$
$\Delta = \delta(q_0, 0) \cup \delta(q_1, 0) \cup \delta(q_2, 0) = \{q_1\}, \quad \{q_1\} \cup \Lambda(q_1) = \{q_1, q_2\}$
$\overline{\delta}(q_0, 0) = \{q_1, q_2\}.$
$(q_0, 1):$
$\Delta = \delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_2, 1) = \{q_0, q_2\}, \quad \{q_0, q_2\} \cup (\Lambda(q_0) \cup \Lambda(q_2)) = \{q_0, q_1, q_2\}$
$\overline{\delta}(q_0, 1) = \{q_0, q_1, q_2\}$
$(q_1, 0):$
$\Delta = \delta(q_1, 0) \cup \delta(q_2, 0) = \{q_1\}, \quad \{q_1\} \cup \Lambda(q_1) = \{q_1, q_2\}$
$\overline{\delta}(q_1, 0) = \{q_1, q_2\}$

$(q_1, 1):$
$\Delta = \delta(q_1, 1) \cup \delta(q_2, 1) = \{q_2\}, \quad \{q_2\} \cup \Lambda(q_2) = \{q_2\}$
$\overline{\delta}(q_1, 1) = \{q_2\}$
$(q_2, 0): \Delta = \delta(q_2, 0) = \emptyset$
$\overline{\delta}(q_2, 0) = \emptyset$
$(q_2, 1):$
$\Delta = \delta(q_2, 1) = \{q_2\}, \quad \{q_2\} \cup \Lambda(q_2) = \{q_2\}$
$\overline{\delta}(q_2, 1) = \{q_2\}.$

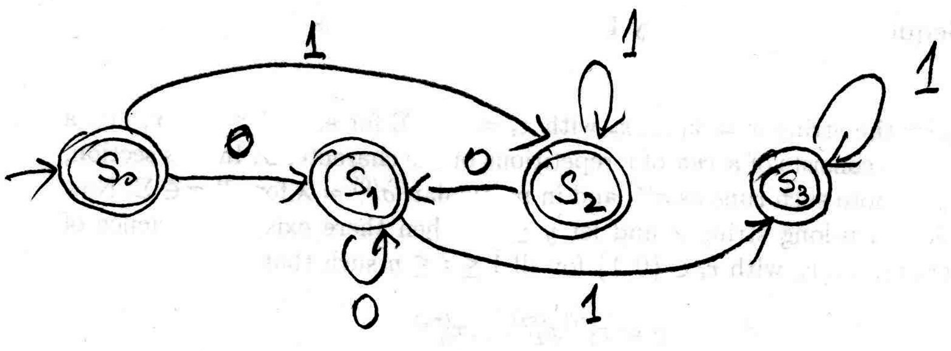
Tehát a $\overline{\mathrm{A}}$ véges automata átmenettáblázata és átmenetdiagramja:

$\overline{\delta}$	0	1
q_0	$\{q_1, q_2\}$	$\{q_0, q_1, q_2\}$
q_1	$\{q_1, q_2\}$	$\{q_2\}$
q_2	$\emptyset$	$\{q_2\}$



Ekvivalens determinisztikus véges automata hozzárendelése:

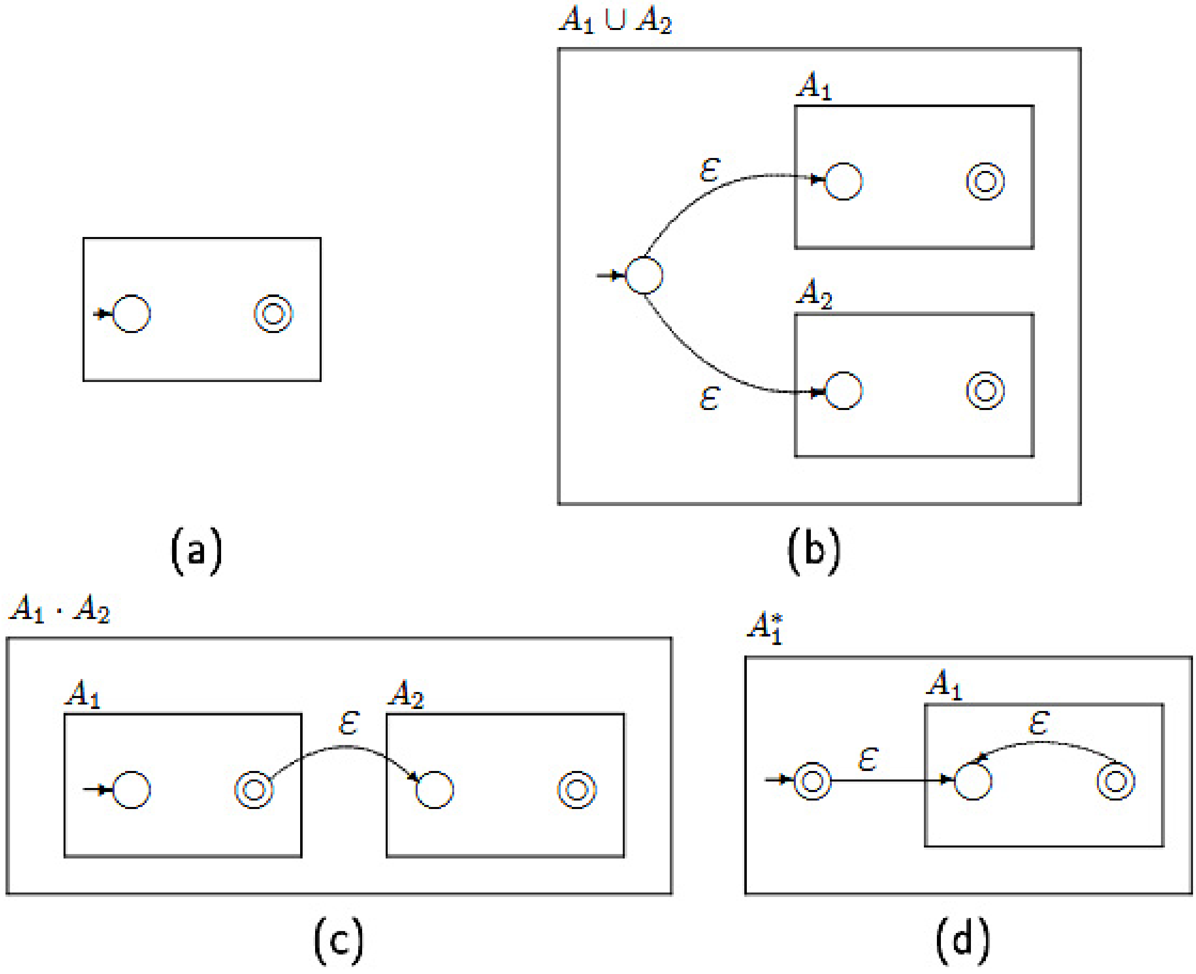
	0	1
$S_0 = \{q_0\}$	$\{S_1\}$	$\{S_2\}$
$S_1 = \{q_1, q_2\}$	$\{S_1\}$	$\{S_3\}$
$S_2 = \{q_0, q_1, q_2\}$	$\{S_1\}$	$\{S_2\}$
$S_3 = \{q_2\}$	$\emptyset$	$\{S_3\}$



Műveletek véges automatákkal

A következőkben értelmezzük a véges automatákon a reguláris műveleteket: egyesítés, szorzat, iteráció. Eredményül ε-lépéses automatát kapunk.

A műveleteket szemléletesen ábrákkal is megadjuk. Egy nyíllal ellátott körrel jelöljük a kezdőállapotokat, és két koncentrikus körből álló jellel a végállapotokat.



Legyenek $\mathrm{A}_1 = (Q_1, \Sigma_1, E_1, I_1, F_1)$ **és** $\mathrm{A}_2 = (Q_2, \Sigma_2, E_2, I_2, F_2)$ **véges automaták.**

A művelet eredménye az A**-val jelölt** $\mathrm{A} = (Q, \Sigma, E, I, F)$ ε**-lépéses automata. Feltételezzük, hogy minden esetben** $Q_1 \cap Q_2 = \emptyset$**. Ha ez nem teljesül, akkor valamelyik állapothalmaz elemeit átnevezzük.**

Egyesítés. (b) ábra

$\mathrm{A} = \mathrm{A}_1 \cup \mathrm{A}_2$, **ahol**
$Q = Q_1 \cup Q_2 \cup \{q_0\}$,
$\Sigma = \Sigma_1 \cup \Sigma_2$,
$I = \{q_0\}$,
$F = F_1 \cup F_2$,
$E = E_1 \cup E_2 \cup \bigcup_{q \in I_1 \cup I_2} \big\{(q_0, \varepsilon, q)\big\}$. $\qquad L(A_1 \cup A_2) = L(A_1) \cup L(A_2)$.

Ugyanazt az eredményt kapjuk, ha kezdőállapot-halmaznak vesszük a $I_1 \cup I_2$ halmazt egy újabb kezdőállapot helyett. Ekkor egyáltalán nem lesznek ε-lépések.

Szorzat. (c) ábra

$\mathrm{A} = \mathrm{A}_1 \cdot \mathrm{A}_2$, **ahol**
$Q = Q_1 \cup Q_2$,
$\Sigma = \Sigma_1 \cup \Sigma_2$,
$F = F_2$,
$I = I_1$,
$E = E_1 \cup E_2 \cup \bigcup_{\substack{p \in F_1 \\ q \in I_2}} \big\{(p, \varepsilon, q)\big\}$ $\qquad L(A_1 \cdot A_2) = L(A_1)L(A_2)$

Iteráció. (d) ábra

$\mathrm{A} = \mathrm{A}_1{}^*$, **ahol**
$Q = Q_1 \cup \{q_0\}$,
$\Sigma = \Sigma_1$,
$F = F_1 \cup \{q_0\}$,
$I = \{q_0\}$
$E = E_1 \cup \bigcup_{p \in I_1} \big\{(q_0, \varepsilon, p)\big\} \cup \bigcup_{\substack{q \in F_1 \\ p \in I_1}} \big\{(q, \varepsilon, p)\big\}$. $\qquad L(A_1^*) = \big(L(A_1)\big)^*$

Az előbbiekben definiált három művelet segítségével újabb bizonyítását adtuk annak, hogy a reguláris nyelvek zártak a reguláris műveletekre nézve, ugyanis az eredményül kapott ε-lépéses automata átalakítható nemdeterminisztikus véges automatává.

Determinisztikus véges automaták minimalizálása

$\mathrm{A} = (Q, \Sigma, E, \{q_0\}, F)$ teljes, DVA **minimális**, ha bármely vele ekvivalens $\mathrm{A}' = (Q', \Sigma, E', \{q_0'\}, F')$ teljes, DVA esetében teljesül, hogy $|Q| \leq |Q'|$.

Az $\mathrm{A} = (Q, \Sigma, E, \{q_0\}, F)$ DVA p és q állapota **ekvivalens**, ha tetszőleges u szóra, mindkettőből végállapotba jutunk vagy egyikből sem jutunk végállapotba, azaz

$$p \equiv q \text{ ha minden } u \in \Sigma^* \text{ szóra } \begin{cases} p \xrightarrow{u} r,\ r \in F \text{ és } q \xrightarrow{u} s,\ s \in F & \text{vagy} \\ p \xrightarrow{u} r,\ r \notin F \text{ és } q \xrightarrow{u} s,\ s \notin F. \end{cases}$$

Ha két állapot nem ekvivalens, akkor megkülönböztethetők.

Az alábbi algoritmusban csillaggal jelöljük a megkülönböztethető állapotpárokat, az egymással ekvivalenseket pedig összevonjuk.

Nem rendezett állapotpárokhoz állapotpárokból álló listát rendelünk a későbbi megcsillagozás reményében, azaz ha az algoritmus során egy állapotpárt megcsillagoztunk, akkor megcsillagozzuk a hozzárendelt lista összes elemét is.

Az alábbi algoritmust olyan DVA-ra alkalmazzuk, amelyből már kizártuk az elérhetetlen állapotokat. Mivel a véges automata determinisztikus és teljes, a $\delta(p, a)$ pontosan egy elemet tartalmaz, itt is alkalmazzuk $elem$ függvényt, amely az egyelemű halmaz egyetlen elemét adja vissza.

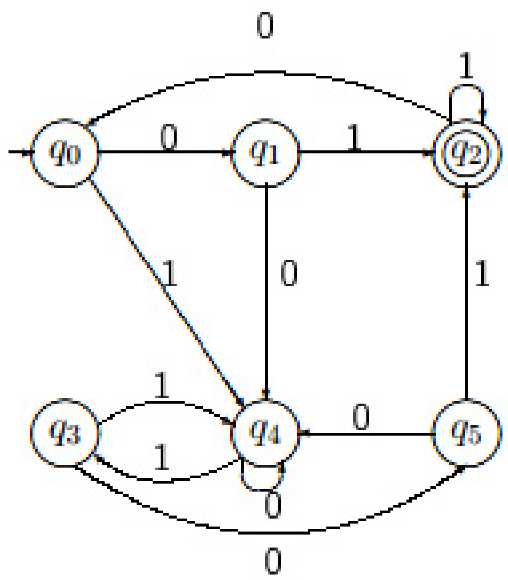
AUTOMATA-MINIMALIZÁLÁSA(A)

jelöljük meg egy-egy csillaggal az összes olyan $\{p, q\}$ állapotpárt, amelyre $p \in F$ és $q \notin F$ vagy fordítva.
minden jelöletlen $\{p, q\}$ állapotpárhoz rendeljünk egy üres listát,
minden jelöletlen $\{p, q\}$ állapotpárra és minden $a \in \Sigma$ betűre vizsgáljuk meg az $\{elem(\delta(p, a)), elem(\delta(q, a))\}$ állapotpárokat,
    ha az így kapott állapotpárok közül valamelyik meg van csillagozva, akkor csillagozzuk meg a $\{p, q\}$ párt is, egyetemben a már előzőleg a $\{p, q\}$ párhoz rendelt lista elemeivel,
    különben, ha a fenti állapotpárok közül egy sincs megcsillagozva, akkor írjuk be a $\{p, q\}$ párt a $\{elem(\delta(p, a)), elem(\delta(q, a))\}$ párokhoz rendelt lista mindegyikébe, feltéve, hogy $\delta(p, a) \neq \delta(q, a)$,
vonjuk össze a megjelöletlen (ekvivalens) állapotpárokat
return A

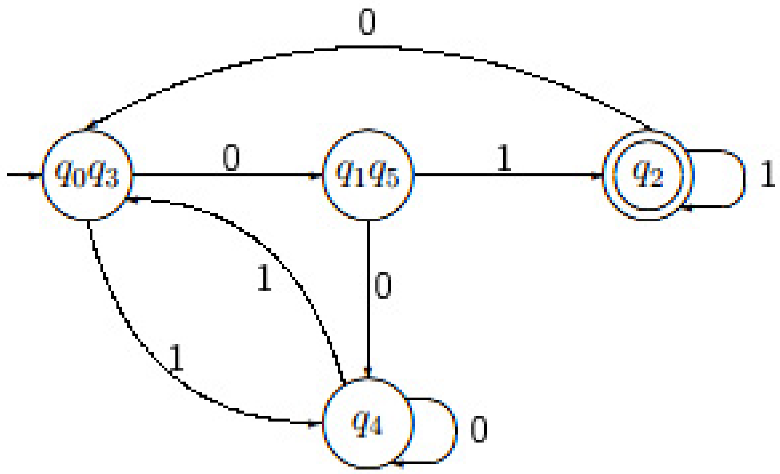
Az algoritmus befejeztével, ha a táblázatban egy cella nem tartalmaz csillagot, akkor a neki megfelelő sor és oszlop indexe két ekvivalens állapot, tehát összevonható. Az összevonást mindaddig folytatjuk, ameddig csak lehetséges. Általánosan, az ekvivalenciareláció az állapotok halmazát ekvivalenciaosztályokra bontja. Minden ilyen osztály állapotai egyetlen állapottá vonhatók össze.

Megjegyzés. Algoritmusunk abban az esetben is alkalmazható, ha a determinisztikus véges automata nem teljes, azaz vannak olyan állapotok, amelyekből adott bemeneti jelre nincs átmenet. Ekkor $\{\emptyset, \{q\}\}$ pár is előfordulhat, és ha q végállapot, úgy tekintjük, mintha ez a pár meg lenne csillagozva.

1. példa.

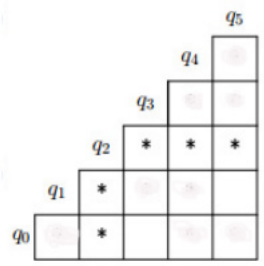


	q_1	q_2	q_3	q_4	q_5
q_4					*
q_3				*	*
q_2			*	*	*
q_1		*	*	*	
q_0	*	*		*	*



Az algoritmus alkalmazásához egy táblázatot használunk, amelyben a csillagozást végezzük. A $\{p, q\}$ állapotpár megjelölését (megcsillagozását) a p sor és q oszlop (vagy q sor és p oszlop) találkozásánál elhelyezett csillag jelzi.

Először megcsillagozzuk a $\{q_2, q_0\}$, $\{q_2, q_1\}$, $\{q_2, q_3\}$, $\{q_2, q_4\}$ és $\{q_2, q_5\}$ párokat (mivel q_2 az egyetlen végállapot).

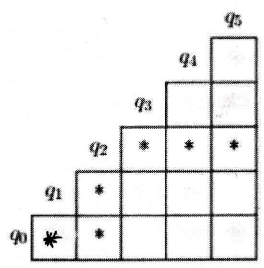


Ezután sorra vesszük a csillaggal meg nem jelölt állapotpárokat, és az algoritmus szerint megvizsgáljuk őket. Kezdjük a $\{q_0, q_1\}$ párral. Hozzárendeljük a következő állapotpárokat:

$\{elem\big(\delta(q_0, 0)\big), elem\big(\delta(q_1, 0)\big)\}, \{elem\big(\delta(q_0, 1)\big), elem\big(\delta(q_1, 1)\big)\}$, azaz $\{q_1, q_4\}$, $\{q_4, q_2\}$.

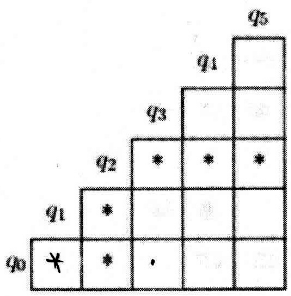
Tehát
$\{q_0, q_1\}$ 0 : $\{q_1, q_4\}$
$\{q_0, q_1\}$ 1 : $\{q_4, q_2\}$

Mivel $\{q_4, q_2\}$ már meg van jelölve, megjelöljük $\{q_0, q_1\}$-t is.

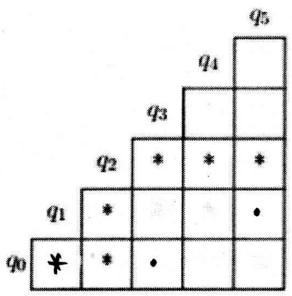


A $\{q_0, q_3\}$ pár esetében a két új pár $\{q_1, q_5\}$ és $\{q_4, q_4\}$. A $\{q_1, q_5\}$ párhoz (mivel nincs megcsillagozva) hozzárendeljük egy listában a $\{q_0, q_3\}$-t, azaz $\{q_1, q_5\} \longrightarrow \{q_0, q_3\}$. Ha később valamikor megcsillagozzuk a $\{q_1, q_5\}$ állapotpárt, akkor meg fogjuk, csillagozni a $\{q_0, q_3\}$-at is.
Egy ponttal jelöljük a táblázatban, hogy egy párt megvizsgáltunk, de nem tudtuk megcsillagozni.
Tehát

$\{q_0, q_3\}$ $0:$ $\{q_1, q_5\}$
$\{q_0, q_3\}$ $1:$ $\{q_4, q_4\}$ $\{q_1, q_5\} \longrightarrow \{q_0, q_3\}$

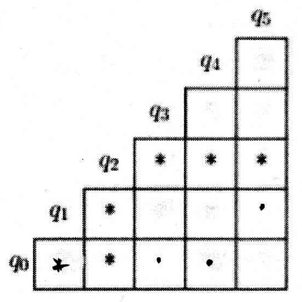


Most $\{q_1, q_5\}$-tel folytatva, a $\{q_4, q_4\}$ és $\{q_2, q_2\}$ párokat kapjuk, amelyekhez az algoritmus szerint semmit sem rendelünk.

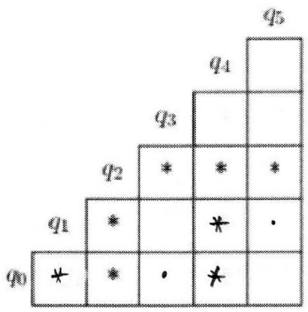


Folytatjuk a $\{q_0, q_4\}$ párral. A hozzárendelt párok $\{q_1, q_4\}$ és $\{q_4, q_3\}$. Egyik sincs megcsillagozva, ezért hozzájuk rendeljük egy-egy listá-

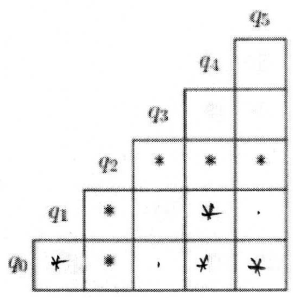
ban a $\{q_0, q_4\}$ **párt, azaz** $\{q_1, q_4\} \longrightarrow \{q_0, q_4\}$, $\{q_4, q_3\} \longrightarrow \{q_0, q_4\}$.



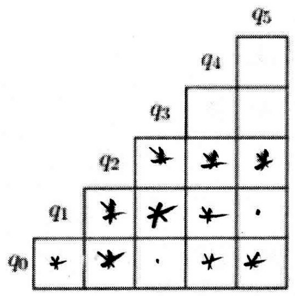
Most a $\{q_1, q_4\}$ **párral folytatva a** $\{q_4, q_4\}$, $\{q_2, q_3\}$ **párokat kapjuk, és mivel ez utóbbi meg van jelölve csillaggal, megjelöljük a** $\{q_1, q_4\}$ **párt és a listában hozzárendelt** $\{q_0, q_4\}$ **párt is.**



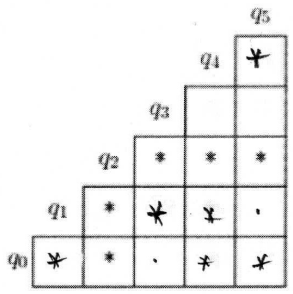
$\{q_0, q_5\}$ 0 : $\{q_1, q_4\}$**, ez meg van csillagozva, megcsillagozzuk** $\{q_0, q_5\}$**-öt is.**



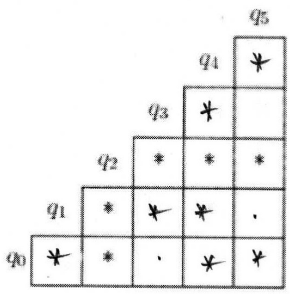
$\{q_1, q_3\}$ 0 **:** $\{q_4, q_5\}$
$\{q_1, q_3\}$ 1 **:** $\{q_2, q_4\}$**, ez meg van csillagozva, megcsillagozzuk** $\{q_1, q_3\}$**-at is.**



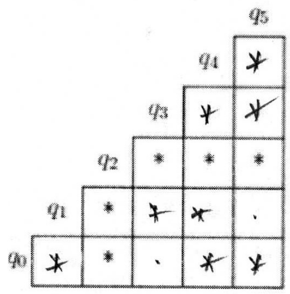
$\{q_4, q_5\}$ 0 **:** $\{q_4, q_4\}$
$\{q_4, q_5\}$ 1 **:** $\{q_2, q_4\}$**, ez meg van csillagozva, megcsillagozzuk** $\{q_4, q_5\}$**-öt is.**



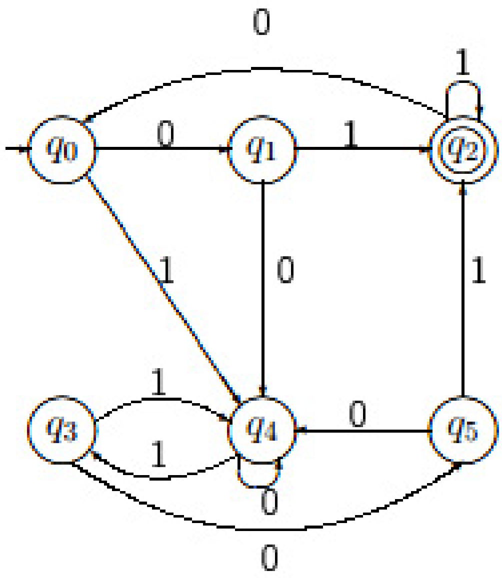
$\{q_3, q_4\}$ 0 : $\{q_5, q_4\}$, ez meg van csillagozva, megcsillagozzuk $\{q_3, q_4\}$-et is.



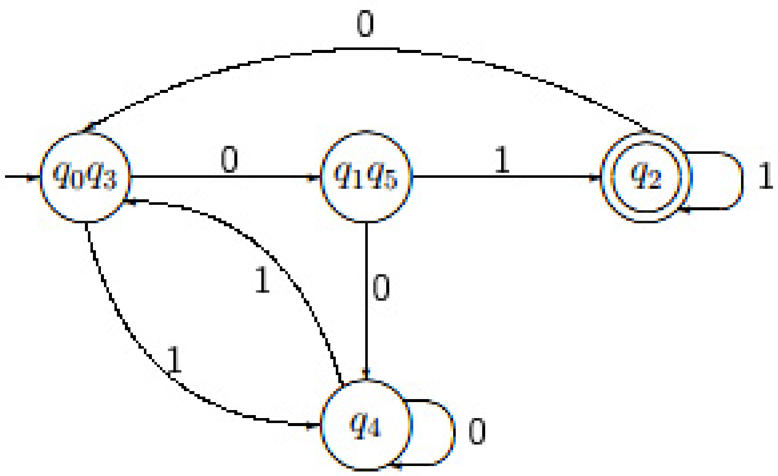
$\{q_3, q_5\}$ 0 : $\{q_5, q_4\}$, ez meg van csillagozva, megcsillagozzuk $\{q_3, q_5\}$-öt is.



Végül azt kapjuk, hogy $q_0 \equiv q_3$ és $q_1 \equiv q_5$. Ezeket összevonva, az ábrán látható determinisztikus véges automatát kapjuk, amelyik ekvivalens az eredetivel.

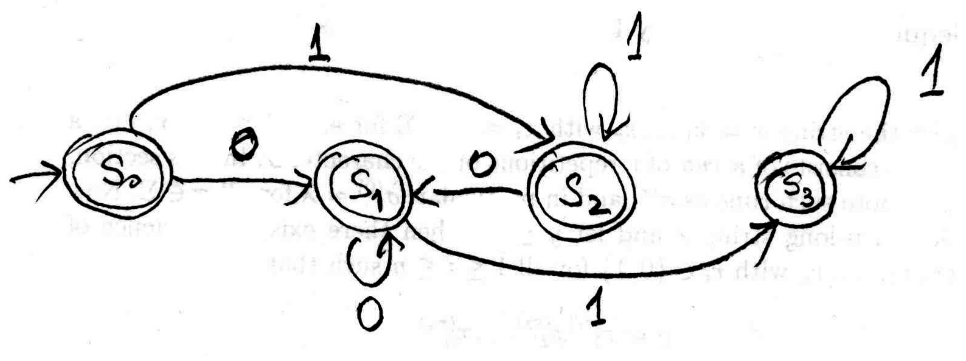


					q_5
				q_4	*
			q_3	*	*
		q_2	*	*	*
	q_1	*	*	*	
q_0	*	*		*	*

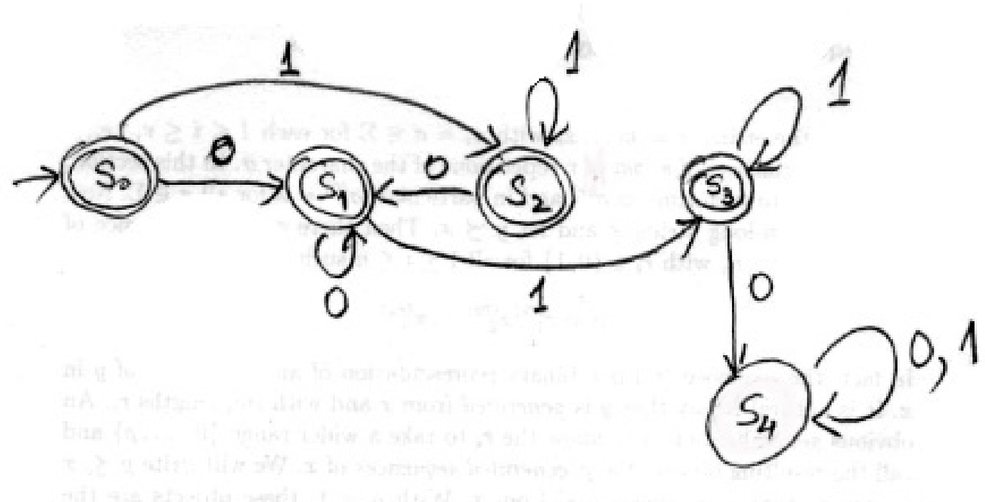


2. példa

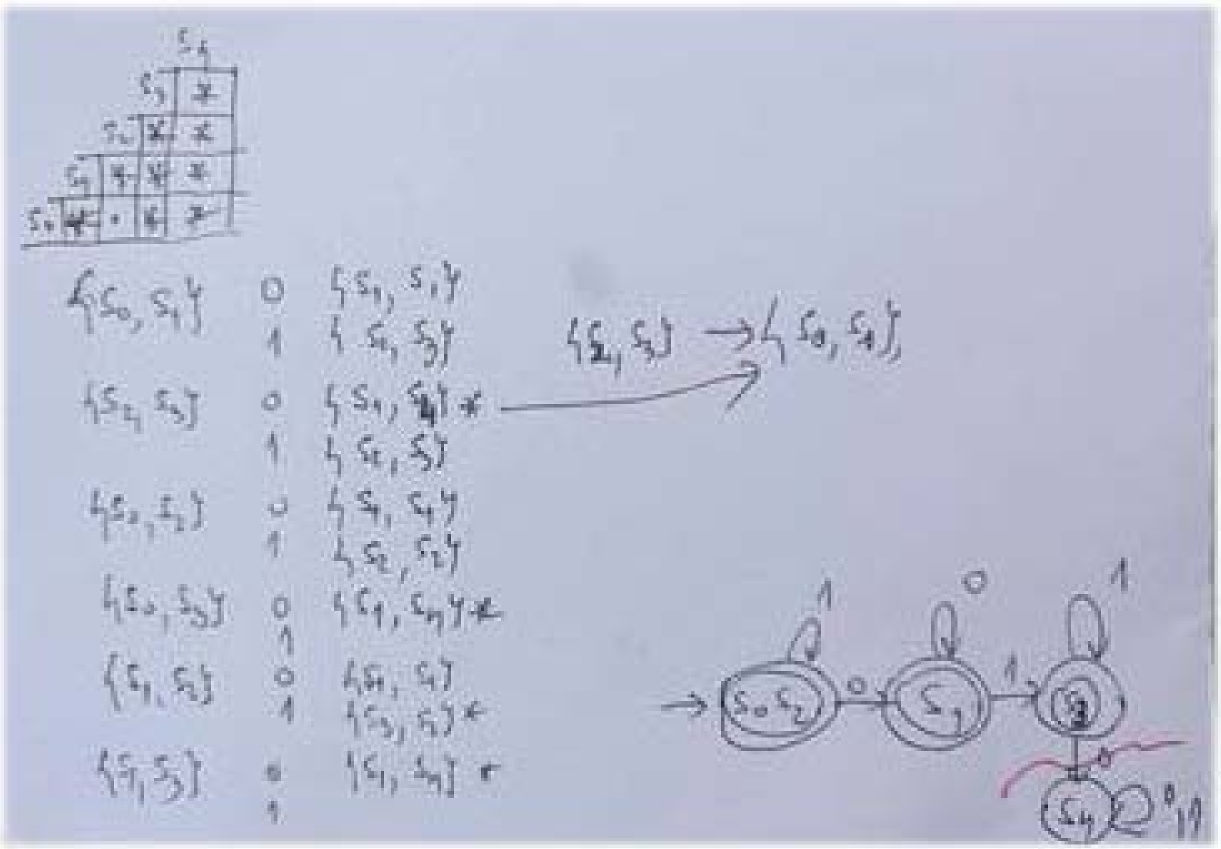
Minimalizáljuk az ε-lépéses automatáknál kapott következő automatát:



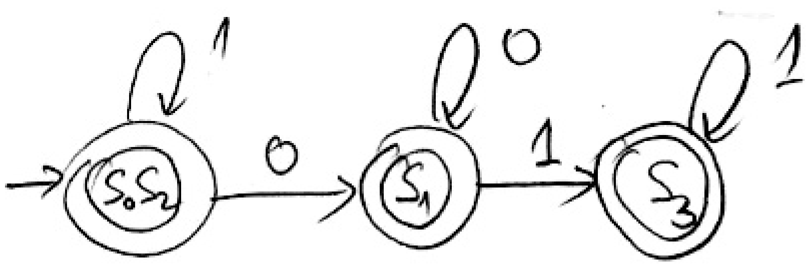
Először teljessé tesszük egy S_4 csapdaállapottal:



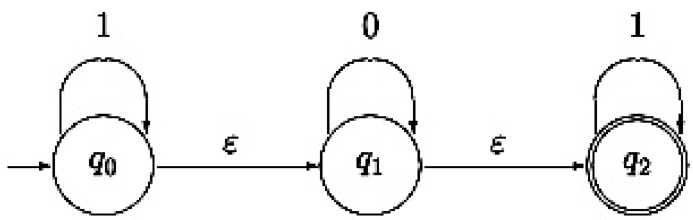
Alkalmazzuk az algoritmust, majd töröljük a csapdaállapotot.



Az eredmény:



Tehát az eredeti



automatától eljutunk a vele ekvivalens következő automatához: