# A számításelmélet alapjai I.

gyakorlati jegyzet

Boda Bálint

2023. tavaszi félév

# Tartalomjegyzék

# 1. fejezet

# Alapfogalmak

**Definíció** (ábécé)**.** Szimbólumok egy véges nemüres halmaza. Például. $V = \{a, b\}$.

**Definíció** (szimbólum)**.** Egy tetszőleges $V$ ábécé elemeit szimbólumoknak vagy betűknek nevezzük.

**Definíció** (szó)**.** Egy $u \in V^*$ (V ábécé elemiből álló véges) sorozatot $V$ feletti szónak (vagy sztringnek) nevezünk.

## 1.1. Szavak

**Definíció.** Legyen $u \in V^*$ egy szó, ekkor a benne lévő betűk számát $u$ hosszának nevezzük és $l(u)$-val vagy $|u|$-el jelöljük.

**Jelölés.** Egy $\delta \in V$ betű az $u \in V^*$ szóban lévő előfordulásinak számát $l(u)_\delta$-val vagy $|u|_\delta$-val jelöljük.

**Definíció** (üres szó)**.** Legyen $V$ egy ábécé, ekkor üres szónak nevezzük azt az $\varepsilon$ szót melyre $|\varepsilon| = 0$.

**Megjegyzés.** Világos, hogy $\varepsilon \in V^*$ bármely $V$ abécé esetén.

**Definíció** ($V^+$)**.** Tetszőleges $V$ ábécé esetén $V^+$ jelöli az $V$ feletti nemüres szavak halmazát, azaz a $V^+ = V^* \setminus \{\varepsilon\}$ halmazt.

### 1.1.1. Műveletek

#### Konkatenáció

**Definíció.** Legyen $V$ egy ábécé és legyenek $u = s_1 \dots s_n$ és $v = t_1 \dots t_k$ $V$ feletti szavak. Ekkor az $uv := s_1 \dots s_n t_1 \dots t_k$ szót $u$ és $v$ konkatenáltjának nevezzük.

#### Tulajdonságok

1. $|uv| = |u| + |v|$
2. általában nem kommutatív (kivétel egyetlen betűből álló ábécék)
3. asszociatív: $u, v, w \in V^* \implies u(vw) = (uv)w$
4. $\forall u, v \in V^* : uv \in V^*$ ($V^*$ a konkatenáció műveletére zárt)
5. létezik egységelem: $\forall u \in V^* : u\varepsilon = u$

Így $V^*$ félcsoport.

**Hatványozás**

**Definíció.** Legyen $i \in \mathbb{N}^+$ és $u \in V^*$. Ekkor $u$ $i$-edik hatványának nevezzük az $u$ szó $i$ darab példányának konkatenáltját.

$$u^0 = \varepsilon,\ u^i = uu^{i-1}\,(i \in \mathbb{N}^+)$$

**Megjegyzés.** Nyilván $\varepsilon^0 = \varepsilon$.

**Tulajdonságok**

1. $u^{n+k} = u^n u^k \ \ (k, n \in \mathbb{N})$
2. $(ab)^n \neq a^n b^n$

**Tükörkép**

**Definíció.** Legyen $u = a_1 \ldots a_n$, ekkor a szó tükörképének (megfordítottjának) nevezzük a

$$u^R = a_n \ldots a_1 \ (1 \leq i \leq n : u_i = u^R_{n+1-i})$$

szót. Alternatív jelölés: $u^{-1}$, $\mathrm{rev}(u)$.

**Megjegyzés.** Ha $u = u^R$ a szót palindrómának (vagy palindrom tulajdonságúnak) nevezzük.

**Tulajdonságok**

1. $\varepsilon^R = \varepsilon$
2. $\left(u^R\right)^R = u$
3. $(uv)^R = v^R u^R$
4. $(u^i)^R = \left(u^R\right)^i \ \ (i \in \mathbb{N})$

### 1.1.2. Részszavak

### 1.1.3. Részszó

Legyen $V$ egy ábécé és legyenek $u$ és $v$ szavak $V$ felett.

**Definíció** (részszó)**.** Az $u$ szó részszava a $v$ szónak, ha $\exists x, y \in V^* : v = xuy$.

**Definíció** (valódi részszó)**.** Az $u$ szó valód részszava a $v$ szónak, ha részszó és $u \neq v$ és $u \neq \varepsilon$.

**Definíció** (prefixum)**.** Ha, $v = xuy$, úgy hogy $x = \varepsilon$, akkor $u$-t $v$ prefixumának nevezzük.

**Definíció** (szuffixum)**.** Ha, $v = xuy$, úgy hogy $y = \varepsilon$, akkor $u$-t $v$ szuffixumának nevezzük.

Az $u$ szót $v$ valódi prefixumainak/szuffixumainak nevezzük, ha $u \neq \varepsilon \wedge u \neq v$.

## 1.2. Nyelv

**Definíció** (nyelv)**.** Legyen $V$ egy ábécé, ekkor nyelvnek nevezzük az $L \subseteq V^*$ halmazt. Ekkor $L$-t $V$

**Jelölés.** $\emptyset$-el jelöljük az üres nyelvet. $\emptyset \neq \{\varepsilon\}$

### 1.2.1. Műveletek

Mivel a nyelvek halmazok értelmezzük az unió, metszet, különbség és komplementer műveleteket.

**Konkatenáció**

**Definíció.** Legyen $V^*$ egy ábécé és $L_1, L_2 \subseteq V^*$, ekkor $L_1$ és $L_2$ konkatenációjának nevezzük az

$$L_1 L_2 = \{u_1, u_2 | u_1 \in L_1, u_2 \in L_2\}$$

a nyelvet.

**Példa.**

$$\{a, b\} \{ab, b\} = \{aab, ab, bab, bb\}$$

**Tulajdonságok**

1. Minden $L$ nyelv esetén: $\{\varepsilon\} L = L \{\varepsilon\}$
2. Asszociatív
3. Egységelem: $\{\varepsilon\}$.

**Hatványozás**

**Definíció.** Legyen $V^*$ egy ábécé és $L \subseteq V^*$, ekkor $L$ $i$-edik hatványának nevezzük a

$$L^0 = \{\varepsilon\}, \qquad L^i = LL^{i-1} \quad (i \geq 1)$$

a nyelvet.

**Megjegyzés.** $\emptyset^0 = \{\varepsilon\}$.

**Iteratív lezárt**

**Definíció.** Egy $L$ nyelv iteratív lezártjának nevezzük az

$$L^* = \bigcup_{i \geq 0} L^i = L^0 \cup L^1 \cup \dots$$

nyelvet.

**Pozitív lezárt**

**Definíció.** Egy $L$ nyelv pozitív lezártjának nevezzük az

$$L^+ = \bigcup_{i \geq 1} L^i = L^0 \cup L^1 \cup \dots = L^* \setminus \{\varepsilon\}$$

nyelvet.

### 1.2.2. Feladatok

Legyenek

$$
\begin{aligned}
L_1 &= \{a, bb, aba\} \\
L_2 &= \{ab^n \mid n \geq 0\} = \{a, ab, abb, \dots\} \\
L_3 &= \{u \in \{a, b\}^* \mid l_a(u) = l_b(u)\} = \{\varepsilon, ab, ba, \dots\} \\
L_4 &= \{u \in \{a, b\}^* \mid l_b(u) \bmod 2 = 0\} = \{\varepsilon, a, bb, abb, aabb, \dots\} \\
L_5 &= \{\varepsilon, ba\}
\end{aligned}
$$

nyelvek. Határozzuk meg:

$$
\begin{aligned}
L_1 \cap L_2 &= \{a\} \\
L_1 \cap L_3 &= \emptyset \\
L_1 \cap L_4 &= \{a, bb\} \\
L_2 \setminus L_1 &= \{ab^n \mid n \geq 1\} \\
L_1 L_5 &= \{a, aba, bb, bbba, ababa\} \\
L_1^2 &= \{aa, abb, aaba, bba, bbbb, bbaba, abaa, ababb, abaaba\}
\end{aligned}
$$

Legyenek

$$
\begin{aligned}
L_1 &= \{a^n b^m \mid m \geq n \land n \geq 0\} = \{\varepsilon, b, ab, abb, \dots\} \\
L_2 &= \{ab\}^* = \{\varepsilon, ab, abab, \dots\}
\end{aligned}
$$

nyelvek. Határozzuk meg:

$$
\begin{aligned}
L_1 \cap L_2 &= \{\varepsilon, ab\} \\
L_1 \setminus L_2^* &= \{a^n b^m \mid m \geq n \geq 2\} \cup \{b\}^+ \cup \{ab^n \mid n \geq 1\} \\
L_1^* &= \{\varepsilon, b, ab, abb, bb, bab, abab, \dots\} \\
L_2 \; L_1^* &= \emptyset \qquad (ab \in L_1^* \text{ miatt}) \\
L_2^* &= L_2
\end{aligned}
$$

# 2. fejezet

# Grammatikák

## 2.1. Generatív grammatika

**Definíció** (grammatika)**.** Egy $G = (N, T, P, S)$ rendezett négyest (generatív) grammatikának vagy nyelvtannak nevezünk, ha $N$ és $T$ diszjunkt (azaz $N \cap T = \emptyset$) véges ábécék. Ekkor

- $N$ a nem terminális szimbólumok halmaza,
- $T$ (vagy $\Sigma$) a terminális szimbólumok halmaza,
- $S \in N$ a grammatika kezdőszimbóluma,
- $P = \{(x, y) \mid x, y \in (N \cup T)^*$ szavak úgy, hogy $x$ legalább egy nem terminális betűt tartalmaz$\}$, az ún. (átírási) szabályok (vagy produkciók) halmaza.

**Jelölés.** Gyakran $(x, y)$ helyett az $x \to y$ jelölést használjuk, egy szabály leírására. Természetesen ez csak akkor lehetséges ha az adott ábécének nem eleme $\to$.

**Definíció** (egylépéses levezetés)**.** Legyen $G = (N, T, P, S)$ egy grammatika és legyen $u, v \in (N \cup T)^*$. Azt mondjuk $v$ közvetlen levezethető az $u$ szóból $G$-ben (jelekkel: $u \Rightarrow_G v$), ha

$$\exists (x, y) \in P: \; u = u_1 x u_2 \text{ és } v = u_1 y u_2, \quad (u_1, u_2 \in (N \cup T)^*)$$

**Definíció** (többlépéses levezetés)**.** Legyen $G = (N, T, P, S)$ egy grammatika és legyen $u, v \in (N \cup T)^*$. Azt mondjuk $v$ több lépésben levezethető az $u$ szóból $G$-ben (jelekkel: $u \Rightarrow_G^* v$), ha

$$u = v \vee \exists \left(n \geq 1 \wedge w_0, \ldots, w_n \in (N \cup T)^*\right), \text{ hogy } w_{i-1} \Rightarrow_G w_i \, (1 \leq i \leq n), \; w_0 = u \text{ és } w_n = v$$

**Definíció** (generált nyelv)**.** Legyen $G = (N, T, P, S)$ egy grammatika, ekkor a $G$ által generált nyelvnek nevezzük az $S$ kezdőszimbólumból több lépésben levezethető terminális szavak halmazát, azaz a

$$L(G) = \{u \in T^* \mid S \Rightarrow_G^* u\}$$

nyelvet.

**Példa.** Legyen

$$\begin{aligned} &G = (\{S, A, B\}, \{a, b\} \, P, S) \\ &P = \{S \to B|bb, \; B \to aaA, \; A \to a|\varepsilon \; \} \end{aligned}$$

Adjuk meg $L(G)$-t!

**Megjegyzés.** Egy $S \to B|bb$ az $S \to B$ és $S \to bb$ szabályokat jelöli.

$$\begin{aligned} &S \to bb \\ &S \to B \to aaA \to a \\ &S \to B \to aaA \to \varepsilon \end{aligned}$$

Így: $L(G) = \{bb, aa, a\}$.

### 2.1.1. Feladatok

**1. Feladat** Legyen $G_i = (\{S, A, B\}, \{a, b\}, P_i, S)$. Határozzuk meg az $L(G_i)$ nyelvet, ha

- $P_1 = \{S \to aaS|a\}$
- $P_2 = \{S \to aSb|\varepsilon\}$
- $P_3 = \{S \to ASB|\varepsilon,\ AB \to BA,\ BA \to AB,\ A \to a,\ B \to b\}$

**Megoldás**

$$L(G_1) = \{a, aaa, aaaaa \dots\} = \left\{a^{(2n+1)} \mid n \geq 0\right\}$$
$$L(G_2) = \{\varepsilon, ab, aabb \dots\} = \{a^n b^n \mid n \geq 0\}$$

A harmadik nyelv meghatározása már nehezebb feladat. Tekintsünk pár példa levezetést:

$$\begin{aligned}
&S \to A\underline{S}B \to \underline{A}B \to a\underline{B} \to ab \\
&S \to ASB \to AB \to BA \to bA \to ba \\
&S \to A\underline{S}B \to AA\underline{S}BB \to A\underline{AB}B \to \underline{AB}AB \to BA\underline{AB} \to \underline{B}ABA \to \dots \to baba \\
&S \to ASB \to AASBB \to AABB \to ABAB \to BAAB \to baab
\end{aligned}$$

Ezek alapján $L(G_3) = \{u \in \{a, b\}^* \mid l_a(u) = l_b(u)\}$.

## 2.2. A grammatikák Chomsky féle osztályzása

Legyen $G = (N, T, P, S)$ egy grammatika. A $G$ grammatika $i$-típusú ($i = 0, 1, 2, 3$), ha a $P$ szabályhalmazra teljesülnek a következők:

- $i = 0$ (mondatszerkezetű grammatika): nincs korlátozás
- $i = 1$ (környezetfüggő grammatika):
  - $P$ minden szabálya $u_1 A u_2 \to u_1 v u_2$ alakú, ahol $u_1, u_2, v \in (N \cup T)^*, A \in N$ és $v \neq \varepsilon$
  - Kivétel: $P$ tartalmazhatja az $S \to \varepsilon$ szabályt, de csak akkor, ha $S$ nem fordul elő egyetlen szabály jobb oldalán sem.
- $i = 2$ (környezetfüggetlen): $P$ minden szabálya $A \to v$ alakú ($A \in N,\ v \in (N \cup T)^*$)
- $i = 3$ (reguláris): $P$ minden szabálya $A \to uB$ vagy $A \to u$ alakú ($A, B \in N,\ u \in T^*$)

Az adott osztályokat $\mathcal{G}_i$-vel jelöljük.

**Definíció** (nyelvosztály)**.** Az $i$ típusú nyelvek osztályának nevezzük a

$$\mathcal{L}_i = \{L \mid \exists G \in \mathcal{G}_i, \text{ hogy } L = L(G)\}$$

**1. Tétel** (Chomsky nyelvhierarchia tétel)**.**

$$\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0$$

### 2.2.1. Feladatok

**1. Feladat** Írjuk fel azt a grammatikát, mely a 4-el osztható bináris számok nyelvét generálja! Milyen osztályba sorolható a generált nyelv?

**Megoldás** Egy kettes számrendszerbeli szám akkor osztható néggyel, ha utolsó két számjegye 0. Gondoskodnunk kell továbbá arról, hogy ne legyenek felesleges nullák az elején. Így

$$G = \left(\{S, B\}, \{0, 1, \varepsilon\}, \left\{S \to \underbrace{0}_{3.} \mid \underbrace{1B00}_{2.}, B \to \underbrace{\varepsilon}_{3} .\mid \underbrace{0B}_{3.} \mid \underbrace{1B}_{3.}\right\}, S\right)$$

(Az adott szabály jobb oldala alatt tüntettem fel annak szintjét.) Mivel kettes a legkisebb szint ezért a generált nyelv is 2-es szintű.

A feladat megoldható más módon is:

$$G = (\{S, A\}, \{0, 1, \varepsilon\}, \{S \to 0|1A, A \to \varepsilon|0A|1A\}, S)$$

amiből már reguláris nyelv adódik.

**2. Feladat** Írjuk fel azt a grammatikát, ami az $L(G) = \{a^n b^m c^n \mid n \geq 0, m \geq 3\}$, nyelvet generálja!

**Megoldás** Írjuk fel $L(G)$ néhány elemét: $\{bbb, abbbc, aabbbcc \dots\}$. Így $G = (\{S, A, B, C\}, \{a, b, c\}, P, S)$, ahol

$$P = \left\{\begin{array}{l} S \to ASC, \\ S \to BBB, \\ A \to a, \\ B \to b, \\ C \to c \end{array}\right\}$$

## 2.3. Reguláris kifejezések

### 2.3.1. A 3-as normálforma

**Definíció.** Egy $G = (N, \Sigma, P, S)$ reguláris grammatikát hármas normálformájúnak (3NF) nevezünk, ha

$\forall p \in P$ produkciós szabályra igaz, hogy $A \to aB$ vagy $A \to \varepsilon$ alakú $(A, B \in N, a \in \Sigma)$

**3NF-re hozás**

1. Ellenőrzés, hogy a nyelv valóban reguláris-e
2. Hossz redukció ($A \to a_1, \dots, a_n B$ alakú szabályok felbontása)
3. Láncmentesítés
   (a) Láncszabályok ($A \to B$ alakúak) felírása
   (b) $U$ halmazok meghatározása
   (c) szabályhalmaz átalakítása

**Példa.** Tekintsük a következő szabálykészletet:

$$S \to abS \mid B, \quad B \to bB \mid V, \quad V \to aa \mid b$$

1. Hossz redukció:

$$\begin{aligned} S \to abS &\Rightarrow S \to aZ_1,\ Z_1 \to bS \\ V \to aa &\Rightarrow V \to aZ_2,\ Z_2 \to aZ_3,\ Z_3 \to \varepsilon \end{aligned}$$

   Így az új szabálykészlet:

$$S \to aZ_1 \mid B,\ Z_1 \to bS, \quad B \to bB \mid V, \quad V \to aZ_2,\ Z_2 \to aZ_3,\ Z_3 \to \varepsilon,\ V \to bZ_3$$

2. Láncmentesítés:
   Láncszabályok: $S \to B,\ B \to V$
   $U$ halmazok:

   (a) $B$:

$$\begin{aligned} U_1(B) &= \{B\} \\ U_2(B) &= U_1(B) \cup \{\text{nemterminálisok melyekből } U_1(B) \text{ egyik eleme elérhető}\} \\ &= U_1(B) \cup \{S\} = \{B, S\} \\ U_3(B) &= U_2(B) \cup \varnothing = \underline{\{B, S\} = U(B)} \end{aligned}$$

   (b) $V$:

$$\begin{aligned} U_1(V) &= \{V\} \\ U_2(V) &= U_1(V) \cup \{B\} \\ U_3(V) &= U_2(V) \cup \{S\} \\ U_4(B) &= U_3(B) \cup \varnothing = \underline{\{B, S, V\} = U(V)} \end{aligned}$$

   Szabályhalmaz átírása:

$$S \to aZ_1 \mid bB \mid aZ_2 \mid bZ_3,\ Z_1 \to bS, \quad B \to bB \mid aZ_2 \mid bZ_3 \quad Z_2 \to aZ_3,\ Z_3 \to \varepsilon$$

# 3. fejezet

# Véges automaták

Véges automatának nevezünk egy olyan analitikus eszközt, mely képes eldönteni egy adott szóról, hogy az általa ismert reguláris nyelv eleme-e.

## 3.1. Alapfogalmak

**Definíció. Determinisztikus véges automatának** (DVA) nevezünk egy $A = (Q, \Sigma, \delta, Q_0, F)$ rendezett ötöst, ahol

- $Q$ a lehetséges állapotok véges nemüres halmaza
- $\Sigma$ az inputszimbólumok ábécéje
- $\delta : Q \times \Sigma \to Q$ az ún. állapotátmeneti függvény
- $q_0 \in Q$ a kezdőállapot
- $F \subseteq Q$ az elfogadó állapotok halmaza.

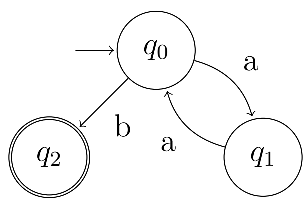
Legyen $A = (\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{q_2\})$ $\delta$ több módon is megadható:

- Függvény: $\delta(q_0, a) = q_1,\ \delta(q_0, b) = q_2,\ \delta(q_1, a) = q_0$
- Grammatikaszerűen: $M_\delta = \{q_0a \to q_1,\ q_0b \to q_2,\ q_1a \to q_0\}$
- Táblázatosan:

| | $a$ | $b$ |
|---|---|---|
| $\to q_0$ | $q_1$ | $q_2$ |
| $q_1$ | $q_0$ | $-$ |
| $\leftarrow q_2$ | $-$ | $-$ |

  A kezdőállapotot $\to$-val, az elfogadó állapotokat $\leftarrow$-val jelöljük.

- Állapotátmeneti gráffal:




  ahol, $\to$ jelöli a kezdőállapotot és dupla kör az elfogadó állapotokat.

**Definíció. Nemdeterminisztikus véges automatát** hasonlóan definiáljuk, annyi különbséggel, hogy a $\delta$ függvény $Q$ helyett, $Q$ hatványhalmazába képez, azaz $\delta : Q \times \Sigma \to \mathcal{P}(Q)$.

$\delta$ hasonlóan adható meg mint DVA esetén, és a nemdeterminisztikusság minden esetben könnyen leolvasható:

- függvény értéke halmaz lesz
- grammatika szerű jelölésben vagy ($|$) jelenik meg valamelyik kifejezés jobb oldalán
- táblázatban halmazok vannak
- egy csúcsból több azonos címkéjű él vezet ki

**Definíció** (konfiguráció)**.** Legyen $A = (Q, \Sigma, \delta, Q_0, F)$ egy véges automata, ekkor **konfigurációnak** nevezünk egy $u \in Q\Sigma^*$ szót.

**Definíció** (egylépéses redukció)**.** Legyen $A = (Q, \Sigma, \delta, Q_0, F)$ egy véges automata és legyenek $u, v$ konfigurációk. Azt mondjuk $u$ egy lépésben redukálható $v$-re, ha létezik $p \in \delta(q, a)$ szabály és $w \in \Sigma^*$ szó, hogy $u = qaw$ és $v = pw$.

**Definíció** (redukció)**.** Az $A = (Q, \Sigma, \delta, Q_0, F)$ véges automata az $u \in Q\Sigma^*$ szót a $v \in Q\Sigma^*$ szóra redukálja (jelölés: $u \Rightarrow_A^* v$), ha vagy $u = v$ vagy valamely $k \geq 1$-re léteznek $w_0, \ldots, w_k$ konfigurációk melyekre $w_{i-1} \Rightarrow_A w_i (1 \geq i \geq k)$, $w_0 = u$ és $w_k = v$.

**Definíció** (felismert nyelv)**.** Legyen $A = (Q, \Sigma, \delta, Q_0, F)$ egy véges automata. Az $A$ automata által elfogadott (vagy felismert) nyelvnek nevezzük az

$$L(A) = \{u \in \Sigma^* | q_0 u \Rightarrow_A^* p \wedge q_0 \in Q \wedge p \in F\}$$

nyelvet.

### 3.1.1. Feladatok

**1. Feladat** Adjunk meg automatát és generatív grammatikát a következő reguláris kifejezésekhez:

a) $(a+b)^* a$

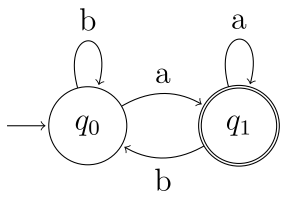
b) $1\,(1+0)^*\,0+0$

c) $a\,(a+b+c\,(a+b))^*$

**Megoldás**

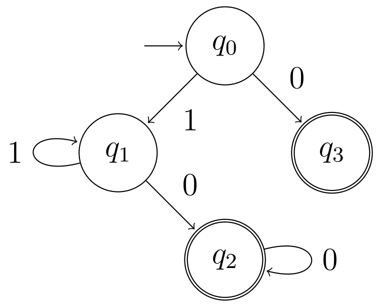
a) Grammatika: $(\{S\}, \{a,b\}, \{S \to aS \mid bS \mid a\})$

Automata:




b) Grammatika: $(\{S,C\}, \{0,1\}, P, S)$, ahol
$P = \{S \to 0 \mid 1C,\ C \to 0C \mid 1C \mid 0\}$

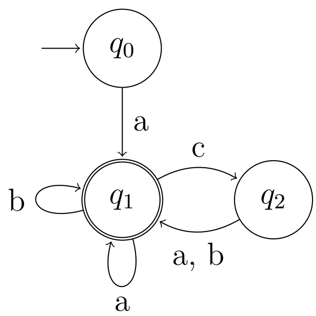
Automata:




c) Grammatika: $(\{S,C,E\}, \{a,b,c\}, P, S)$, ahol

$$P = \left\{ \begin{array}{c} S \to aE, \\ E \to \varepsilon \mid aE \mid bE \mid cC, \\ C \to aE \mid bE \end{array} \right\}$$

Automata:



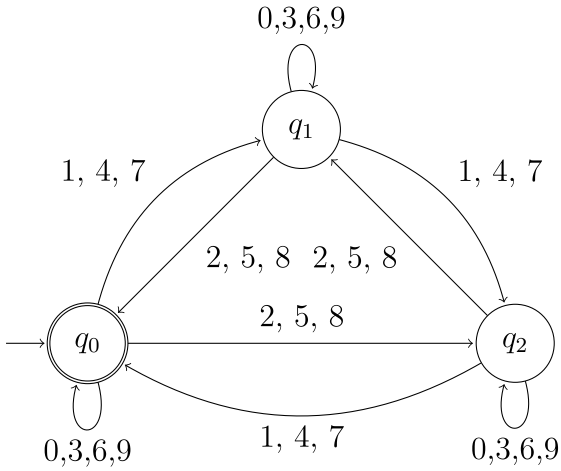
**2. Feladat** Adjunk meg automatát, mely felismeri a következő nyelveket:

a) 3-al osztható természetes számok (vezető nullákat nem kezelve)

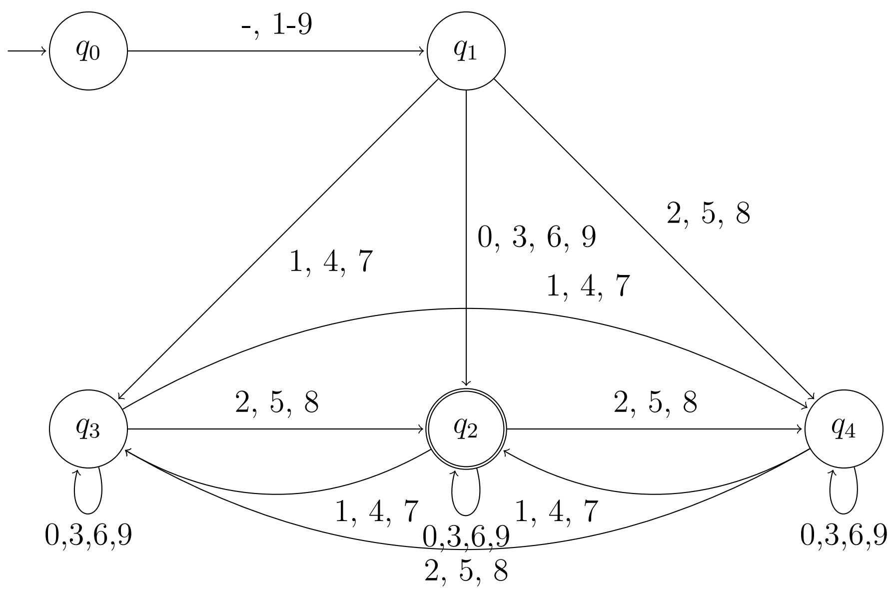
b) 3-al osztható egész számok (vezető nullákat kezelve)

## Megoldás

a) Egy szám akkor osztható hárommal, ha számjegyeinek összege osztható hárommal. Az automata azonban nem képes sem aritmetikai műveletekre, sem ezek eredményének eltárolására. Megfeleltethetjük, azonban a különböző állapotokat a hárommal való osztás maradékosztályainak.
Például: Legyen a vizsgálandó szó 156. Az automata először feldolgozza az 1-es szimbólumot, melynek hatására a $q_1$ állapotba kerül, mivel $1 \mod 3 = 1$. Ezt követően, mivel jelenleg a $q_1$ állapotban vagyunk és $5 \mod 3 = 2$, ezért a 0-ás maradékosztályba azaz $q_0$-ba kerülünk.Ezután mivel $6 \mod 3 = 0$, ezért a $q_0$ állapotban maradunk.



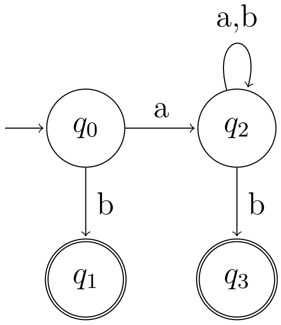

b) A megoldás nagyon hasonló annyi különbséggel, hogy fel kell vennünk két extra állapotot. Az egyik állapot a mínusz jelet a másik pedig a vezető nullákat fogja kezelni.



# 3.2. Automaták determinizálása

Alapötlet: hozzunk létre összevont állapotokat pl. az $\{q_1, q_2, q_3\}$ állapotot vonjuk össze a $q_{123}$ állapottá.

**Példa.** Tekintsük a következő automatát:




| | $a$ | $b$ |
|---|---|---|
| $\rightarrow q_0$ | $\{q_2\}$ | $\{q_1\}$ |
| $q_1$ | $\varnothing$ | $\varnothing$ |
| $\leftarrow q_2$ | $\{q_2\}$ | $\{q_2, q_3\}$ |
| $\leftarrow q_3$ | $\varnothing$ | $\varnothing$ |

**Megjegyzés.** A nemdeterminisztikus automaták esetében a táblázatban gyakran egyelemű állapothalmazok helyett csak magát az állapotot írjuk le.

Világos hogy az automata nem determinisztikus, ezért determinizáljuk: Az $q_0$ és $q_1$ sorokat változatlanul leírjuk a $q_2$ sorban azonban el kell végeznünk a $q_2$ és a $q_3$ összevonását.

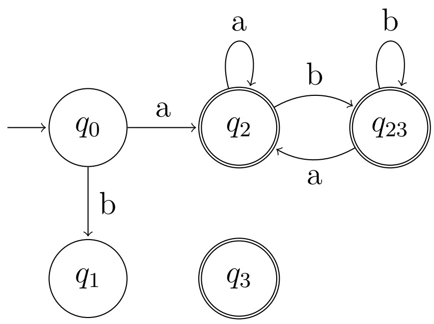
| | $a$ | $b$ |
|---|---|---|
| $\rightarrow q_0$ | $q_2$ | $q_1$ |
| $q_1$ | $\varnothing$ | $\varnothing$ |
| $\leftarrow q_2$ | $q_2$ | $q_{23}$ |
| $\leftarrow q_3$ | $\varnothing$ | $\varnothing$ |
| $?q_{23}$ | ? | ? |

Mivel $q_2$ és $q_3$ közül legalább az egyik elfogadó állapot volt így az összevont állapot is elfogadó lesz. Ezt követően már csak az összevonásokat kell elvégeznünk:

1. $\delta(q_{23}, a) = \delta(q_2, a) \cup \delta(q_3, a) = \{q_2\} \cup \varnothing = \{q_2\}$ Egyelemű, nem kell további összevonás
2. $\delta(q_{23}, b) = \delta(q_2, b) \cup \delta(q_3, b) = \{q_{23}\} \cup \varnothing = \{q_{23}\}$ Egyelemű, nem kell további összevonás

Így az automata:

| | $a$ | $b$ |
|---|---|---|
| $\rightarrow q_0$ | $q_2$ | $q_1$ |
| $q_1$ | $\varnothing$ | $\varnothing$ |
| $\leftarrow q_2$ | $q_2$ | $q_{23}$ |
| $\leftarrow q_3$ | $\varnothing$ | $\varnothing$ |
| $\leftarrow q_{23}$ | $q_2$ | $q_{23}$ |




Észrevehetjük, hogy a $q_3$ állapot elérhetetlenné vált, azaz az automata elvesztette összefüggőségét. Az összefüggőség megőrzéséhez bevezethetünk egy egyszerű változtatást az algoritmusba, hogy egy sort állapotot csak akkor veszünk az új automatához, ha arra egy korábbi sor már hivatkozik.

Tekintsünk egy bonyolultabb példát:

**Példa.** Determinizáljuk a következő automatát az optimalizált algoritmus alapján!

| | $a$ | $b$ |
|---|---|---|
| $\to q_0$ | $\{q_1, q_2\}$ | $\emptyset$ |
| $\leftarrow q_1$ | $q_1$ | $q_2$ |
| $\leftarrow q_2$ | $q_0$ | $q_1$ |
| $q_3$ | $\{q_1, q_3\}$ | $q_0$ |

1. $\delta(q_0, a)$ állapotokat össze kell vonni ($q_{12}$)
2. Meg kell határozni a $q_{12}$ sort:
   (a) $\delta(q_{12}, a) = \delta(q_1, a) \cup \delta(q_2, a) = q_{01}$
   (b) $\delta(q_{12}, b) = \delta(q_1, b) \cup \delta(q_2, b) = q_{12}$
3. Meg kell határozni a $q_{01}$ sort:
   (a) $\delta(q_{01}, a) = \delta(q_0, a) \cup \delta(q_1, a) = q_{12} \cup q_1 = q_{12}$
   (b) $\delta(q_{01}, b) = \delta(q_0, b) \cup \delta(q_1, b) = q_2$
4. Meg kell határozni a $q_2$ sort.
5. Meg kell határozni a $q_1$ sort.

| | $a$ | $b$ |
|---|---|---|
| $\to q_0$ | $q_{12}$ | $\emptyset$ |
| $\leftarrow q_{12}$ | $q_{01}$ | $q_{12}$ |
| $\leftarrow q_{01}$ | $q_{12}$ | $q_2$ |
| $\leftarrow q_2$ | $q_0$ | $q_1$ |
| $\leftarrow q_1$ | $q_1$ | $q_2$ |

Mivel minden állapot ismert az algoritmus optimalizált verziója véget ér.

**Példa.** Determinizáljuk a következő automatát az optimalizált algoritmus alapján!

| | $a$ | $b$ |
|---|---|---|
| $\to q_0$ | $\{q_0, q_1\}$ | $\{q_0, q_2\}$ |
| $q_1$ | $q_3$ | $\emptyset$ |
| $q_2$ | $\emptyset$ | $q_4$ |
| $\leftarrow q_3$ | $q_3$ | $q_3$ |
| $\leftarrow q_4$ | $q_4$ | $q_4$ |

| | $a$ | $b$ |
|---|---|---|
| $\to q_0$ | $q_{01}$ | $q_{02}$ |
| $q_{01}$ | $q_{013}$ | $q_{02}$ |
| $q_{02}$ | $q_{01}$ | $q_{024}$ |
| $\leftarrow q_{013}$ | $q_{013}$ | $q_{023}$ |
| $\leftarrow q_{024}$ | $q_{014}$ | $q_{024}$ |
| $\leftarrow q_{023}$ | $q_{013}$ | $q_{0234}$ |
| $\leftarrow q_{014}$ | $q_{0134}$ | $q_{024}$ |
| $\leftarrow q_{0234}$ | $q_{0134}$ | $q_{0234}$ |
| $\leftarrow q_{0134}$ | $q_{0134}$ | $q_{0234}$ |

## 3.3. Véges automaták összefüggővé alakítása

Egy automata összefüggő, ha minden állapot elérhető. Az összefüggővé alakítás algoritmusa a következő:

- $H_0 = \{q_0\}$
- $H_{i+1} = H_i \cup \{H_i$ valamely eleméből elérhető állapotok$\}$

Az algoritmus akkor áll meg, ha $H_{i+1} = H_i$. Az automata összefüggő, ha $Q = H$. Az automata összefüggővé alakítható ha lecseréljük $Q$-t $Q \cap H$-ra.