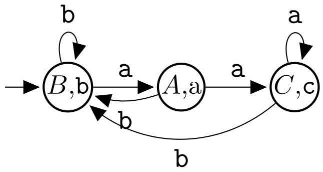
# 10. Fordítók

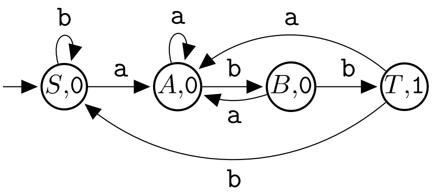
1. Adjon meg egy olyan Moore-automatát, aminek bemeneti ábécéje $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, kimeneti ábécéje $\Gamma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ és az automata minden szóban az egymás mellett levő `a` betűket a másodiktól kezdve lecseréli `c`-re. (Például `aabbaaa` $\mapsto$ `acbbacc`.)

   *Megoldás:* 3 állapot elég lesz (nem `a`-ra végződik, egy `a`-ra végződik, `aa`-ra végződik), mindegyikhez más-más kimeneti karakter tartozik. Ezek között az átmenetek világosak:



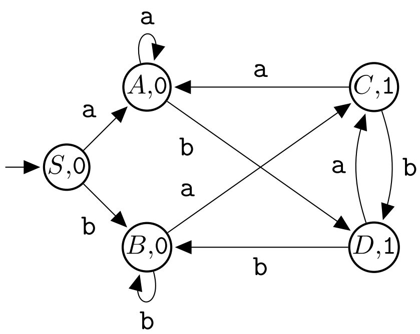

2. Adjon meg egy olyan Moore-automatát, aminek bemeneti ábécéje $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, kimeneti ábécéje $\Gamma = \{\mathtt{0}, \mathtt{1}\}$. Az automata általában `0`-t ír ki, de amikor az `abb` részszó megjelent a bemenetben, akkor `1`-t (azaz olyan `b`-nél, ami előtt két `a` is volt).

   *Megoldás:* 4 állapotot használunk, amelyek azt jelzik, hol tartunk az `abb` szóban.




3. Adjon meg egy Moore-automatát, ami minden $w \in \{\mathtt{a}, \mathtt{b}\}^*$ szóból egy olyan 0-1 sorozatot készít, amiben mindig ott van 1, ahol a karakter különbözik az előzőtől!

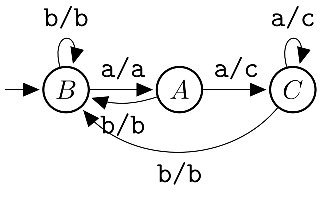
   *Megoldás:* A kezdőállapoton kívül 4 állapot van, melyek azt mutatják, hogy mi az utolsó karakter, és hogy ez ugyanaz-e, mint az előző ($A$ és $C$, illetve $B$ és $D$).



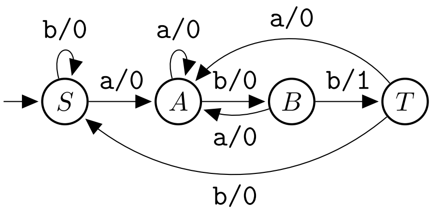

4. A tanult módon alakítsa át az előző Moore-automatákat Mealy-automatává!

   *Megoldás:* Az állapotok maradnak, kimenet az átmenetekre tolódik.

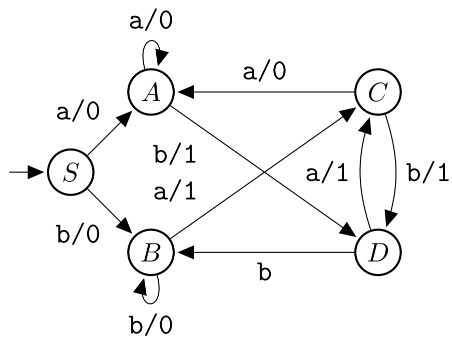
1. feladat




2. feladat

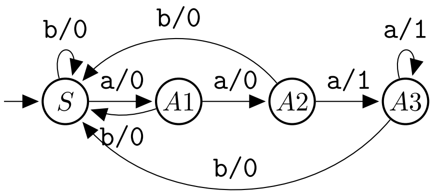



3. feladat



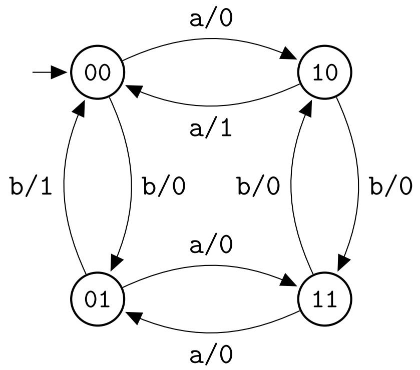
5. Adjon meg egy Mealy-automatát, ami minden $w \in \{\mathtt{a}, \mathtt{b}\}^*$ szóból egy olyan 0-1 sorozatot készít, amiben mindig ott van 1, ahol az eredeti szóban megjelent az `aaa` részszó, azaz egy `a` előtt van már két `a`.

   *Megoldás:* 4 állapotot használunk, amelyek azt jelzik, hol tartunk az `aaa` szóban. Minden `b` visszavisz a kezdőállapotba.



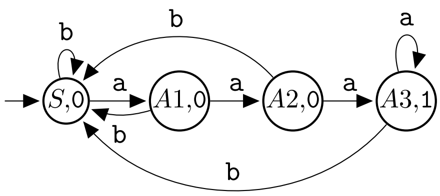

6. Adjon meg egy Mealy-automatát, ami minden nem üres $w \in \{\mathtt{a}, \mathtt{b}\}^*$ szóból egy olyan 0-1 sorozatot készít, amiben annyiszor fordul elő az `1`, ahány olyan kezdőszelete van $w$-nek, ahol az `a`-k és a `b`-k száma is páros.

   *Megoldás:* A már ismert, a paritásokat számon tartó 4 állapotú automatához adunk kimeneteket, ami akkor lesz 1, amikor a páros-párosnak megfelelő `00` állapotba lépünk.

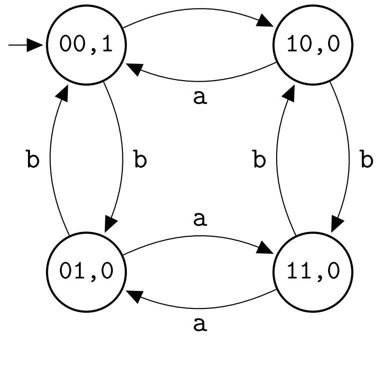



7. A tanult módon alakítsa át az előző Mealy-automatákat Moore-automatává!

   *Megoldás:* 5. feladat automatája: Minden eredeti állapotnak 2 változata lehet, a két kimeneti karakternek megfelelően, de csak azokat hozzuk létre, amelyek a kezdőállapotból elérhetők. Legyen a kezdőállapot az ($S$,`0`).




   6. feladat automatája: Minden eredeti állapotnak 2 változata lehet, a két kimeneti karakternek megfelelően. A `00`-ba lépve a kimenet `1` lesz, érdemes az ennek megfelelő `(00,1)` jelölésű állapotot választani kezdőállapotnak.



Látszik, hogy így minden eredeti állapotnak csak az egyik változata kell, ami világos, hiszen minden állapotba csak azonos kimenetű nyilak mutattak.

Ha a $(\texttt{00},\texttt{0})$ állapotot választjuk kezdőállapotnak, természetesen akkor ebbe nem fogunk visszalépni, az eredetileg a $\texttt{00}$-ba vivő átmenetek a $(\texttt{00},\texttt{1})$-be mennek.

8. Van-e olyan Moore-automata, amire teljesül, hogy minden $k \geq 1$ esetén $\mu(\texttt{a}^{2k}) = \texttt{0}^{2k}$ és $\mu(\texttt{a}^{2k+1}) = \texttt{01}^{2k}$ ?

*Megoldás:* Nincs, hiszen egy Moore-automatánál minden $w$ szóra teljesül, hogy $\mu(w\texttt{a}) = \mu(w)\mu(\texttt{a})$, ami a megadott $\mu$ esetén nem igaz.

9. Adjon meg olyan véges fordítót, ami az $\{\texttt{a},\texttt{b}\}^*$-on van értelmezve és minden $n > 0$ esetén az $\texttt{a}^n\texttt{b}^n$ szóból a $\texttt{c}^k$ szót állítja elő, ahol

   (a) $k = n$

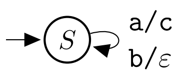
   (b) $k = 2n$

   (c) $k = \lfloor n/2 \rfloor$

   Az előző automaták mire fordítanak egy tetszőleges $w \in \{\texttt{a},\texttt{b}\}^*$ szót?

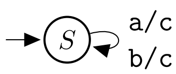
*Megoldás:* Mindegyik esetben sok megoldás van, arra kell vigyázni, hogy a megadott automaták nem akadhatnak el, hiszen a fordítás mindenhol értelmezett kell legyen. Egy-egy egyszerű megoldást adunk. A válasz arra, hogy mire fordítanak egy tetszőleges szót erősen függ a megadott automatától.

(a)



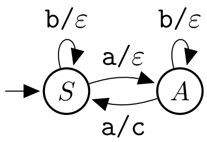

$\texttt{c}^k$, ahol $k$ a szóban előforduló $\texttt{a}$ betűk száma.

(b)




$\texttt{c}^k$, ahol $k$ a szó hossza.

(c)




$\texttt{c}^k$, ahol $k$ a szóban előforduló $\texttt{a}$ betűk számának a fele (lefelé kerekítve).

10. Van-e olyan véges fordító, aminek értékkészlete

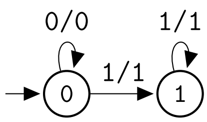
   (a) $\{\texttt{0}^n\texttt{1}^n : n \geq 0\}$ ?

   (b) $\{\texttt{0}^n\texttt{1}^k : n, k \geq 0\}$ ?

*Megoldás:*

(a) Nincs, mert véges fordító értékkészlete reguláris, a megadott nyelv pedig nem az.

(b) Van, pl. ha csak a $\texttt{0}^n\texttt{1}^k$ alakú szavakon van értelmezve, és ezeken a fordítás megegyezik a szóval. Ehhez egy fordítóautomata:

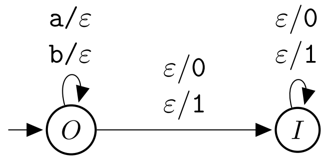



11. Adjon meg egy véges fordítót a következő relációhoz (bemenet–kimenet párokhoz): $\{\texttt{a},\texttt{b}\}^* \times \{\texttt{0},\texttt{1}\}^* - \{(\texttt{aa},\texttt{1})\}$, azaz az $(\texttt{aa},\texttt{1})$ kivételével minden pár benne van.

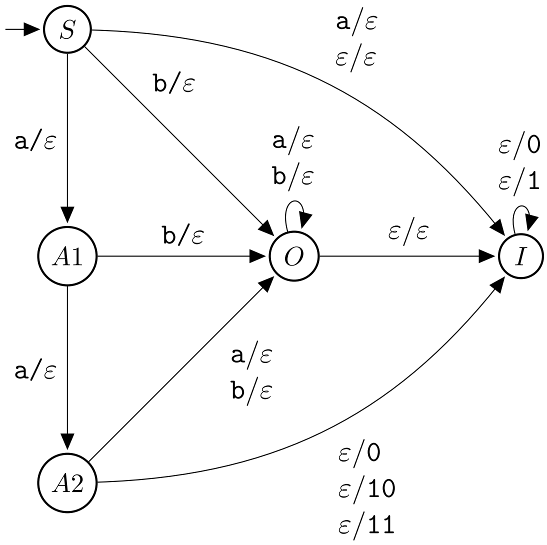
*Megoldás:* Természetesen nemdeterminisztikus automata kell, hiszen minden szónak végtelen sok fordítása lesz.

Itt most az egyszerűség kedvéért feltételezzük, hogy az automata, miután a bemenetet elolvasta, bármikor leállhat.

Előbb nézzünk egy véges fordítót az összes szópárhoz, azaz először legyen $F = \{\texttt{a},\texttt{b}\}^* \times \{\texttt{0},\texttt{1}\}^*$. Ehhez egy állapot is elég, de célszerűbb lesz szétválasztani a műveleteket: az $O$ állapotban tetszőleges karaktersorozatot beolvasunk, az $I$ állapotban az olvasástól függetlenül tetszőleges bitsorozatot kiírunk. Vegyük észre, hogy ha a bemenet vége előtt lép át az automata az $I$ állapotba, akkor elakad, és ezen a számítási úton nincs érvényes kimenet. Ha viszont a bemenetet végigolvasva kerül az $I$-be, akkor itt bármikor leállhat, lesz érvényes kimenet.




A kívánt fordítónál is elválaszthatjuk az írás és olvasás részt. Most is legyen egy mindent kiíró $I$ állapotunk. Ebbe egyből beléphetünk, ha a bemenet üres vagy egyetlen a betűből áll. Ha b-vel kezdődik, akkor az $O$ állapotban beolvasunk minden további karaktert. Lépjünk ugyanide, ha a bemenet ab-vel kezdődik ($A1$-ből), vagy ha aa után még folytatódik ($A2$-ből) Az $O$ után jöhet a mindent kiíró $I$ állapot. Az aa szó esetén a megengedett kimenetek: $\varepsilon$, $0(0+1)^*$, $(10+11)(0+1)^*$. Az első előáll, ha megállunk $A2$-ben, a többi lehetőség pedig a megfelelő bitek kiírása után az $I$-ben megvalósítható. Fontos, hogy itt a szükséges 10 és 11 kiírása egy lépésben történjen (az automata ne állhasson meg közben).




Vegyük észre, hogy ez az automata egyetlen bemeneten sem akad el. Az aa szó esetén az olvasás az $A2$ állapotban ér véget, ami után a kimenet bármi lehet csak 1 nem. Minden más bemenet olvasása vagy az $O$ állapotban ér véget, ahonnan írás (és olvasás) nélkül átléphetünk az $I$-be, vagy eleve az $I$ állapotban ér véget. Innen tetszőleges 0-1 kimeneti sorozathoz van megfelelő számítási út.

12. Legyen $M_1$ és $M_2$ két véges fordító, melyeknek bemeneti ábécéje $\Sigma$, kimeneti ábécéje pedig $\Gamma$. Igazolja, hogy ha $L_1, L_2 \subseteq \Gamma^*$ a két véges fordító értékkészlete, akkor $L_1 \cup L_2$ is előáll mint egy véges fordító értékkészlete.

    *Megoldás:* A két véges fordítót építsük össze egy új kezdőállapot felvételével, amiből két átmenet indul ki: mindkét eredeti kezdőállapotba $\varepsilon$ olvasással és $\varepsilon$ írással léphet át. Így egy (új) véges fordítót kapunk ($\varepsilon$-átmenetek megengedettek!). A kapott fordítás a két fordítás uniója lesz, és akkor ez áll az értékkészletekre is.

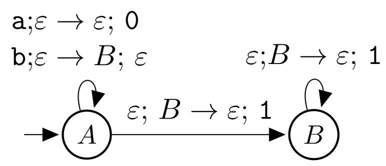
13. Tekintsük azt a fordítási feladatot, amikor minden $w \in \{a, b\}^*$ szóhoz egy olyan $0^k1^n$ szót rendelünk, ahol $k$ a $w$-ben szereplő a betűk, $n$ pedig a b betűk száma!

    (a) Adjon meg ehhez egy veremfordítót!

    (b) Adjon meg hozzá egy szintakszisvezérelt fordítási sémát!

    *Megoldás:*

    (a) A veremfordító működése: amikor a betűt lát, kiír egy 0-t, a b betűket meg a veremben gyűjti. Amikor a szó végére ért, akkor a veremből kiszedegeti a b-ket és mindegyiknél kiír egy 1-et.




(b) A fordítási sémában mindkét nyelvtan szabályos CF nyelvtan. Az 1. nyelvtanban balról jobbra generáljuk a karaktereket, míg a 2. nyelvtanban a `0` a változó elé, míg az `1` a változó mögé kerül és így a `0`-k elöl, az `1`-ek hátul gyűlnek:

$$S \to \varepsilon; \varepsilon \mid A; A$$
$$A \to \mathtt{a}A; \mathtt{0}A \mid \mathtt{b}A; A\mathtt{1} \mid \mathtt{a}; \mathtt{0} \mid \mathtt{b}; \mathtt{1}$$