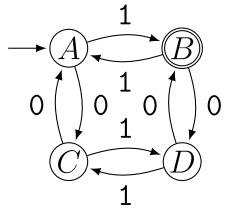
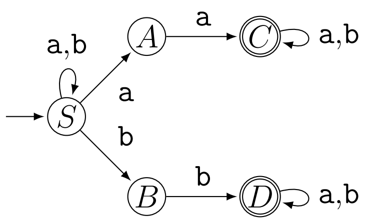
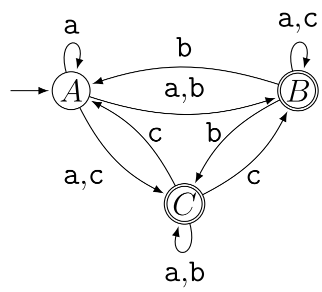
1. Legyen $\Sigma = \{0, 1\}$. Adjon meg egy determinisztikus véges automatát, amely azokat a szavakat fogadja el, amelyekben páros sok nulla és páratlan sok egyes van!

   Megoldás:
   Az ötlet az, hogy számoljuk a nullák és egyesek paritását. Mivel mindegyikre kettő, azaz összesen négy lehetőség van, 4 állapota lesz a DVA-nak. A kezdőállapot (A), amikor mindkettőből páros van (a 0 páros szám!). Egy `0` hatására a nullák paritása, egy `1` hatására az egyesek paritása változik, ennek felelnek meg az átmenetek. Az elfogadó állapot (B), ahova páros sok nullával és páratlan sok egyessel jutunk.

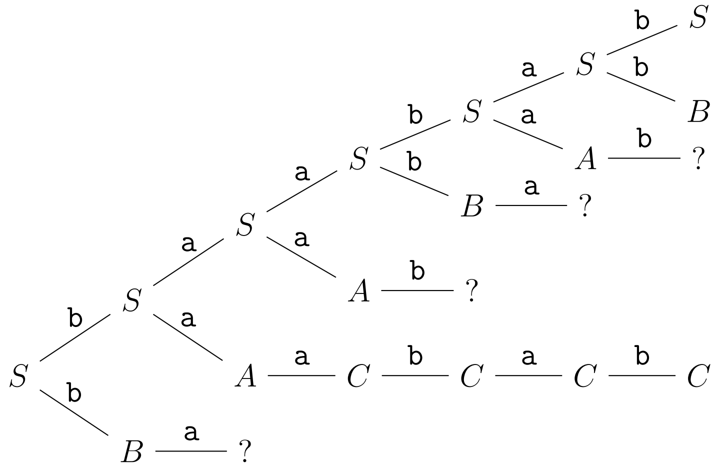


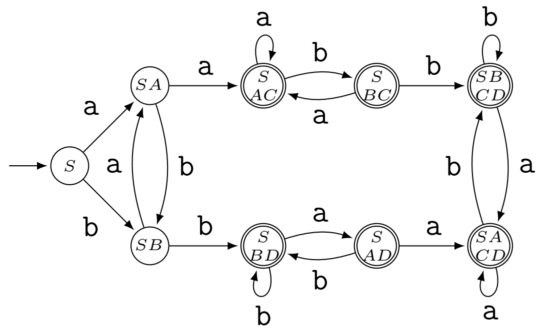
2. Mindkét nemdeterminisztikus véges automatára
   (a) adja meg a `baabab` szóhoz tartozó számítási fát!
   (b) A tanult eljárással készítsen belőlük determinisztikus véges automatát!
   (c) Milyen nyelvet fogadnak el ezek a véges automaták?





Megoldás: Nézzük az elsőt! Erre a számítási fa és a determinizált automata:

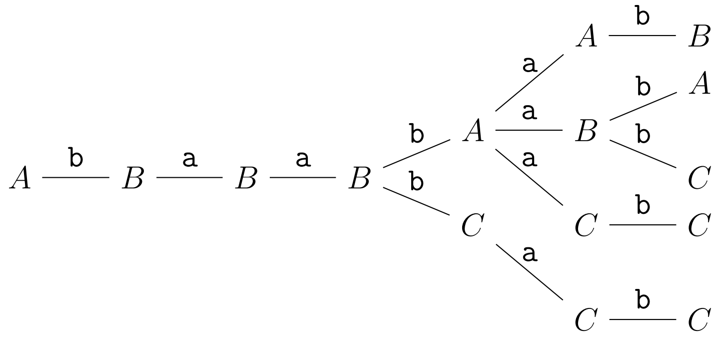


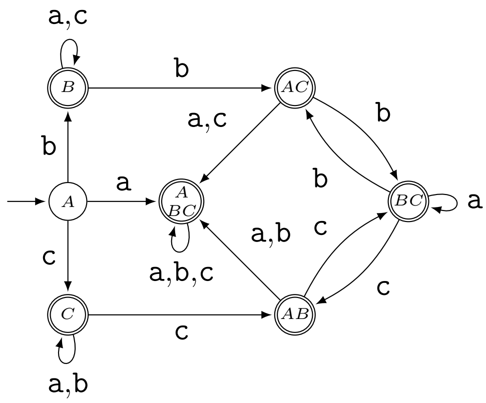


A számítási fáról leolvasható, hogy elfogadja a szót, hiszen van olyan számítási út, ami végig ér és az elfogadó C állapotban végződik.

A nemdeterminisztikus változatból jól látszik, hogy azokat a szavakat fogadja el, amelyekben az `aa` és a `bb` részszavak közül legalább az egyik előfordul. Ha egy ilyen előfordulás első karakterénél hagyjuk el az S állapotot, akkor elfogadó állapotban végződik a számítás, különben meg nem, mert nem tud eljutni a C vagy D állapotig. A determinisztikus változaton azt lehet megfigyelni, hogy az SAC és SBD állapotokba akkor érünk, amikor először jelenik meg két egyforma karakter egymás után. Innen már csak elfogadó állapotok között mozgunk (amiket, épp ezért, ha akarjuk, össze is vonhatnánk egyetlen elfogadó állapotba).

A második automatánál a számítási fa és a determinizálással kapott automata:





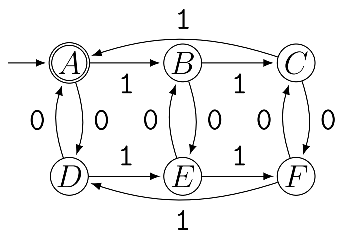
A számítási fában egyik úton sem akadunk el, és mivel van olyan ág, ami nem A-val végződik, a szót az automata elfogadja.

Itt a DVA-n látszik, hogy a szó első karakterével elfogadó állapotba jutunk, és a további karakterek hatására is mindig elfogadó állapot következik. Tehát a nyelv az összes nem üres szóból áll.

Ez a nemdeterminisztikus változatból is leolvasható. Az automata az üres szót nem fogadja el, de az első karakterrel át tudunk menni az elfogadó B és C állapotokból legalább az egyikbe, a továbbiakban mindig van olyan lehetőség, ami nem az A állapotba (az egyetlen nem elfogadó állapotba) vezet.

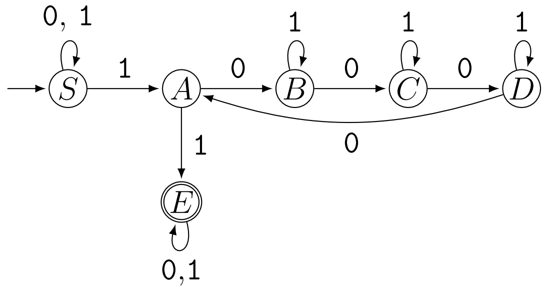
3. Legyen $\Sigma = \{0, 1\}$. Adjon meg egy determinisztikus véges automatát, amely azokat a szavakat fogadja el, amelyekben a nullák száma páros, az egyesek száma osztható 3-mal!

Megoldás:
Az 1. feladathoz hasonló, de itt az egyeseket modulo 3 kell számolni. Az elfogadó állapot a kezdőállapot lesz.



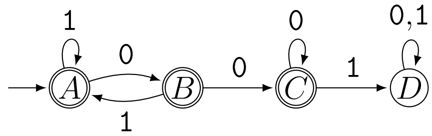
4. Igazolja, hogy reguláris az a nyelv, amelyik az összes olyan 0/1 sorozatot tartalmazza, amelyben van két olyan `1`, hogy a közöttük álló `0`-k száma osztható 4-gyel. (A két választott `1` között további `1`-ek is előfordulhatnak.)

Megoldás:
Ahhoz, hogy igazoljuk a regularitást elegendő egy NVA-t mutatni. Az ötlet az, hogy „nemdeterminisztikusan kivárjuk” amíg a megfelelő `1` nem jön, innen a `0`-kat számoljuk modulo 4 (az esetleg közben jövő `1`-ek nem változtatnak az aktuális állapoton). Amikor a 4-gyel osztható darab `0` után egy `1` jön, akkor elfogadó állapotba megyünk, és ott is maradunk, hiszen a szó már biztos jó. Egy lehetséges megoldás:



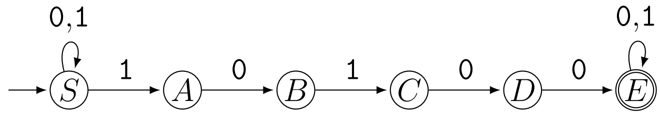
5. Legyen $\Sigma = \{0, 1\}$. Adjon meg egy determinisztikus véges automatát, amely azokat a szavakat fogadja el, amelyekben nem szerepel a `001` részszó.

Megoldás:
Előbb felrajzolunk egy DVA-t, ami a komplementerét fogadja el, majd legyen $F' = Q - F$.

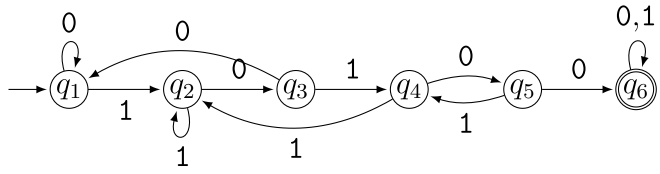


6. Adjon nemdeterminisztikus véges automatát, amely azokat a szavakat fogadja el, amiben szerepel az `10100` részszó!

Megoldás:
Az ötlet az, hogy a kezdőállapotban várunk amíg a megfelelő részhez érünk (tehát a kilépés nemdeterminisztikus lesz). Utána egy állapotláncon végigmenve ellenőrizzük a minta meglétét (ha közben „hibás” karakter jön, akkor a számítás elakad). A megtalált minta után bármi jön, el kell fogadni a szót, azaz ebben az elfogadó állapotban maradunk, bármi is érkezik. Egy lehetséges megoldás:

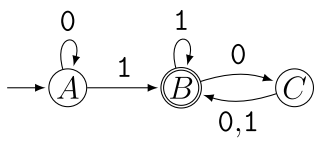


Determinisztikus automatát sem nehéz megadni, pl. így:



Ez nem pontosan ugyanaz, mint amit az előző NVA determinizálásával kapunk, bár annak egyszerűsítésével is megkapható.

7. Mely szavakat fogadja el ez az automata? ($\Sigma = \{0, 1\}$)



Megoldás: Jellemezzük az egyes állapotokat a megfelelő nyelvekkel! Kicsit nézegetve az ábrát a következőre juthatunk
L_A : az `1`-t nem tartalmazó szavak.
L_B : van benne `1`, és az utolsó `1` után páros sok `0` van,
L_C : van benne `1`, és az utolsó `1` után páratlan sok `0` van.
Bizonyítás: $\varepsilon \in L_A$, és valóban kezdetben az A állapotban van. Az 1 hosszú szó vagy az A állapotba vagy a B állapotba vezet, a jellemzésnek megfelelően.

A továbbiakban azt mutatjuk meg, hogy egy szóval akkor és csak akkor jutunk az X állapotba, ha a szó az L_X nyelvbe tartozik. A szó hossza szerinti teljes indukcióval járunk el. A legfeljebb 1 hosszú szavakra már tudjuk, hogy ez igaz. Tegyük fel most, hogy a legfeljebb k hosszú szavakra a jellemzés helyes és tekintsünk egy $w = a_1 a_2 \cdots a_k a_{k+1}$ szót. Három eset van, attól függően, hogy a k-adik karakter után melyik állapotban vagyunk.

az A állapotban: Ekkor eddig csupa `0` jött. Ha $a_{k+1} = 0$, akkor megmarad ez a tulajdonság és így $w \in L_A$, különben pedig $w \in L_B$. Vegyük észre, hogy az automata mozgása ennek megfelelő, az első esetben marad az A állapotban, a másodikban átlép a B-be.

a B állapotban: Az indukciós feltevés szerint mivel a B állapotban vagyunk a k-adik karakter után, ezért $a_1 \cdots a_k \in L_B$, ami azt jelenti, hogy páros sok `0` van az utolsó `1` után. Ha a következő karakter `1`, akkor ez a tulajdonság marad (nulla darab `0` lesz az utolsó `1` után), különben meg a végső `0`-k paritása változik, $w \in L_C$. Megint látjuk, hogy az automata mozgása ennek megfelelő.

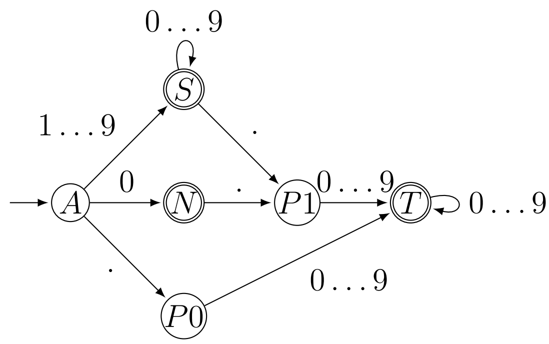
a C állapotban: hasonló az előzőhöz: az indukció szerint $a_1 \cdots a_k \in L_C$, tehát volt már `1`, és az utolsó `1` után páratlan sok `0` volt. Most ha egy `0` jön, akkor a végén levő `0`-k paritása változik. Ha pedig `1` a következő karakter, akkor is átkerülünk L_B-be, hiszen a végén levő `0`-k száma nulla lesz.

Azt láttuk, hogy minden esetben az automata mozgásának megfelelően kerülünk át egyik nyelvből a másikba. Mivel a három nyelv az összes szó egy partícióját alkotja, ezzel beláttuk, hogy ez a jellemzés helyes.

Az automata által elfogadott nyelv az elfogadó állapotnak megfelelő L_B nyelv, azaz az olyan szavak alkotják, amelyekben van `1`, és az utolsó `1` után páros sok (akár 0 darab) `0` következik.

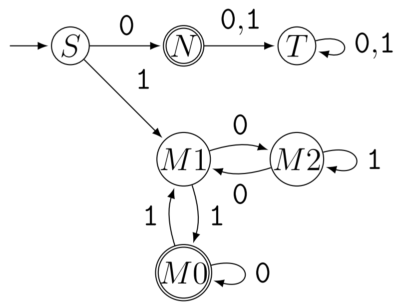
8. Készítsen olyan véges automatát, amely a tizedes tört alakban felírt racionális számokat fogadja el. (Σ a tizedespontból és a `0, 1, 2, 3, 4, 5, 6, 7, 8, 9` számjegyekből áll.) Az elfogadandó szám vagy tizedespont nélküli egész szám (pl. `123`), vagy tartalmaz tizedespontot. Az utóbbi esetben azt is el kell fogadni, ha az egészrész hiányzik. (pl. helyes az `123.456` vagy a `.456` is, de nem fogadható el `123.` és ha a bemenet csak egyetlen pontból áll). Megköveteljük továbbá azt is, hogy az egészrész ne kezdődjön felesleges 0-kal (de pl. a `0.456` helyes).

   Megoldás:
   Itt az áttekinthetőség kedvéért egy lehetséges hiányos automatát adunk meg, ami szükség esetén a szokott módon teljessé tehető. Gondoljuk át, milyen állapotokra van szükség: a kezdőállapotból (A) átléphetünk egy a 0-t elfogadóba (N). Ha itt egy számjegy jön, akkor a bemenet helytelen, ezeket az átmeneteket elhagyjuk, és így a számítás elakad. A nullától különböző egész számokhoz egy másik állapot (S) tartozik. Itt maradunk bármilyen további számjegy esetén, és ez is elfogadó állapot kell legyen. Ezzel az egész számokat kezeltük. Tizedespont jöhet N vagy S után, és a megfelelő állapot ($P1$) elutasító kell legyen. Vagy a tizedespont lehet a legelső karakter, ekkor viszont egy újabb állapot kell, mert ez nem elfogadó ($P0$). Bármelyik változatban a tizedespont után jöhetnek még számjegyek, és ezekkel ismét elfogadó állapotba kerülünk (T). Ha egy újabb pont következik, akkor a bemenet hibás, ezt itt ismét elakadással jelezzük. Tehát a hiányos VA:



9. Legyen $\Sigma = \{0,1\}$. A jelsorozatokat tekintsük mint bináris számokat. Adjon véges automatát, amely pont a hárommal osztható számokat fogadja el! Vegye figyelembe, hogy szám 0-val nem kezdődik, kivéve maga a 0 és hogy a számokat a legmagasabb helyiértékű számjegytől kezdjük olvasni!

   Megoldás:
   A lehetséges 3 maradéknak egy-egy állapot feleljen meg. Az ezek közötti átmenetekhez vegyük észre, hogy ha a bináris alakban felírt x szám után egy 0-t írunk, akkor a szám értéke kétszerese lesz, ha pedig 1-et, akkor kétszerese+1. Ebből könnyen kiszámolható, melyik maradékosztályból melyikbe jutunk egy 0 vagy egy 1 hatására. Még arra kell figyelni, hogy ha az első karakter `0`, akkor ezt elfogadjuk, de ha jön még utána valami, az már nem jó.



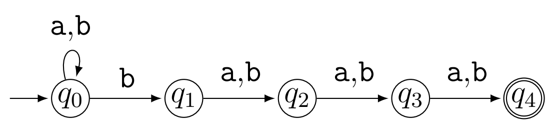
10. Az L_k nyelv álljon az olyan $\Sigma = \{a,b\}$ szavakból, amelyekben hátulról számítva a k-adik karakter `b`.

    (a) Mutassa meg, hogy minden $k \geq 1$ esetén van az L_k nyelvet elfogadó, $k+1$ állapotú nemdeterminisztikus véges automata!

    (b) Mutassa meg, hogy minden, az L_k nyelvet elfogadó determinisztikus véges automatának legalább 2^k állapota van!

    Megoldás:

    Az (a) részhez a „mintaillesztő" NVA mintájára könnyű automatát adni, pl. L_4-hez ez jó lesz:



A (b)-hez mutatunk 2^k darab szót, amelyek közül semelyik kettő nem végződhet ugyanabban az állapotban. Ez garantálja, hogy kell legalább 2^k állapot. Vegyük az összes k hosszú szót. Ezekből pont 2^k darab van. Közülük tetszőleges w_1 és w_2 szóra teljesül, hogy van olyan $1 \leq i \leq k$, hogy hátulról az i-edik betűjük eltér, legyen mondjuk ez w_1-ben a és w_2-ben b. Egészítsük ki mindkét szót $k - i$ darab a betűvel. Az így kapott $s_1 = w_1\mathtt{a}^{k-i} \notin L_k$, hiszen hátulról a k-adik betűje a, míg $s_2 = w_2\mathtt{a}^{k-i} \in L_k$, mert itt meg ez a betű b.

Ráadás: Ezzel van egy alsó becslésünk a DVA méretére. Mekkorát sikerül konstruálni?

11. Bizonyítsa be, hogy minden olyan NVA, ami nem fogadja el az ε-t, átalakítható úgy, hogy ugyanazt a nyelvet ismerje fel, de pontosan egy elfogadó állapota legyen.

    Megoldás: Ha nem fogadja el az ϵ-t, akkor a kezdőállapot nem elfogadó. Vegyünk fel egy új f állapotot. Ha valamilyen nyíl F-beli állapotba mutat, akkor vegyünk fel e mellé egy olyat is, ami ugyanonnan indul, de f-be mutat. Azaz, ha $\delta(r,a) \cap F \neq \emptyset$ valamilyen $r \in Q$ és $a \in \Sigma$ esetén, akkor legyen $\delta'(r,a) = \delta(r,a) \cup f$, különben $\delta'(r,a) = \delta(r,a)$. Valamint legyen $\delta'(f,a) = \emptyset$ minden $a \in \Sigma$ esetén és $F' = \{f\}$.

    Ha az eredeti automatában van egy szóra elfogadó számítás, akkor az utolsó lépést (és van utolsó lépés, mert az üres szót nem fogadjuk el) az újra cserélve egy elfogadó számítást kapunk M'-ben. Ez fordítva is igaz. (Ha $F = \emptyset$, akkor f nem lesz elérhető.)

12. Egy L nyelvből az L^R nyelvet úgy kapjuk, hogy minden L-beli szót megfordítunk, azaz fordított sorrendben írjuk le a karaktereket. Bizonyítsa be, hogy ha L reguláris, akkor L^R is az.

    Megoldás: Ha L reguláris, akkor van hozzá NVA (ami lehet persze DVA is egyben). Ebből az előző feladat megoldása szerint készítünk egy olyan NVA-t amiben 1 elfogadó állapot van. Ha a nyelv tartalmazza az ε-t, akkor a fenti módszerrel olyan NVA készíthető, amiben két elfogadó állapot van, a kezdő q_0 és ez az új f állapot, amibe csak befele menek nyilak. Végül ebben az új NVA-ban megfordítunk minden nyilat, legyen $q_0' = f$ és ha $\varepsilon \notin L$, akkor $F' = \{q_0\}$, ha pedig $\varepsilon \in L$, akkor $F' = \{q_0, f\}$.

    Világos, hogy egy nemüres szó elfogadó számítási útvonalán visszafelé haladva egy olyan számítási utat kapunk, ami a fordított szót fogadja el és az ε-t is jól kezeljük.