4. CF nyelvtanok átalakítása

1. A tanult módszerrel szüntesse meg az egyszeres szabályokat a következő nyelvtanban!

$$S \to A \mid B \qquad A \to \mathtt{a}S\mathtt{b} \mid \mathtt{a} \qquad B \to S\mathtt{b} \mid C \qquad C \to S\mathtt{a} \mid S$$

Megoldás: Az egyszeres szabályok gráfja:

$$\begin{tikzcd} S \arrow[r] \arrow[d] & A \\ B \arrow[r] & C \arrow[ul] \end{tikzcd}$$

Látszik, hogy az S, B, C változókból mind a 4 csúcs elérhető, A-ból egyetlen más csúcs sem. Az új nyelvtan:

$$S \to \mathtt{a}S\mathtt{b} \mid \mathtt{a} \mid S\mathtt{b} \mid S\mathtt{a} \qquad A \to \mathtt{a}S\mathtt{b} \mid \mathtt{a} \qquad B \to \mathtt{a}S\mathtt{b} \mid \mathtt{a} \mid S\mathtt{b} \mid S\mathtt{a} \qquad C \to \mathtt{a}S\mathtt{b} \mid \mathtt{a} \mid S\mathtt{b} \mid S\mathtt{a}$$

(A kapott nyelvtanban vannak felesleges szimbólumok, ezek kiküszöbölése további feladat lehet.)

2. Adjon meg egy CF nyelvtant, amely az $\{\mathtt{a}^k\mathtt{b}^n\mathtt{c}^m : k,n,m \geq 1,\ k=n \text{ vagy } k=m\}$ nyelvet generálja!

Megoldás: Kihasználjuk, hogy $L = L_{ab} \cup L_{ac}$, ahol $L_{ab} = \{\mathtt{a}^k\mathtt{b}^n\mathtt{c}^m : k,n,m \geq 1,\ k=n\}$ és $L_{ac} = \{\mathtt{a}^k\mathtt{b}^n\mathtt{c}^m : k,n,m \geq 1,\ k=m\}$. Ezeket az S_{ab}, illetve az S_{ac} változók generálják, a teljes nyelvtan:

$$\begin{aligned} &S \to S_{ab} \mid S_{ac} \\ &S_{ab} \to XC \qquad X \to \mathtt{a}X\mathtt{b} \mid \mathtt{ab} \qquad C \to \mathtt{c}C \mid \mathtt{c} \\ &S_{ac} \to \mathtt{a}S_{ac}\mathtt{c} \mid \mathtt{a}B\mathtt{c} \qquad B \to \mathtt{b}B \mid \mathtt{b} \end{aligned}$$

Lényegében az L_{ab} nyelv is két részből van generálva, az X változóból kapjuk az $\{\mathtt{a}^k\mathtt{b}^n : k,n \geq 1,\ k=n\}$ nyelvet, ehhez van hozzáfűzve a C-ből megkapható $\{\mathtt{c}^m : m \geq 1\}$. Hasonló történik az L_{ac} nyelv esetében is.

3. Adjon minél magasabb osztályú nyelvtant, amely a szabályos zárójelsorozatokat generálja! Az abc két eleme `(` és `)`.

Megoldás: 2. osztályú (CF) nyelvtanra példák (az egyszerűség kedvéért az üres sorozatot kihagyom a nyelvből, hogy az ε-mentesítés ne takarja el a lényeget):
1. változat Azt használja ki, hogy egy jó sorozat felbontható néhány egymás utáni külső zárójelpárra, és mindegyikben egy-egy jó sorozat lehet:

$$Z \to ZZ \mid (Z) \mid ()$$

Az első szabállyal generálható annyi Z, ahány külső zárójelpár kell. A 2-3. szabályokkal oldhatjuk meg ezek belsejét.
2. változat Az első külső zárójellel kezdi a generálást, ebben és utána jó sorozatnak kell állnia:

$$Z \to (Z)Z \mid ()Z \mid ()$$

3. változat Egy tetszőleges külső zárójelpárral kezd:

$$Z \to Z(Z)Z \mid (Z)Z \mid (Z) \mid ()$$

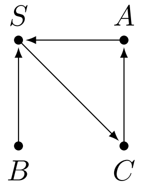
(Igazából ez az üres szót is generáló $X \to X(X)X \mid \varepsilon$ nyelvtannak az üres szót nem generáló CF változata.)

Még meg kell mutatni, hogy a nyelv nem reguláris. Tegyük fel, hogy az, és $p > 0$ a pumpálási hossza. Vegyük azt a z sorozatot, ami p darab nyitó zárójellel kezdődik és utána p darab csukó zárójel van. Ekkor $z \in L$ és $|z| = 2p \geq p$ teljesül. A z minden, a (reguláris) pumpálási lemmának megfelelő $z = uvw$ felosztásában v valahány, de nem nulla nyitó zárójelből áll. Ezért pl. k=2 esetén a pumpált $uv^2w \notin L$, ami ellentmond a pumpálási lemmának, tehát a nyelv nem reguláris.

4. A tanult módszerrel alakítsa át a következő nyelvtant olyanra, amelyben már nincsenek egyszeres szabályok és felesleges szimbólumok!

$$S \to \mathtt{a}A \mid B\mathtt{b} \mid C \qquad A \to A\mathtt{b} \mid S \qquad B \to \mathtt{c} \mid S \qquad C \to A \mid \mathtt{cc}$$

Megoldás: Előbb az egyszeres szabályoktól szabadulunk meg. Ezek gráfja:



Látszik, hogy ebben B-ből minden, az S, A, C változókból B kivételével minden elérhető.

Ennek megfelelően az új nyelvtan:

$$\begin{aligned} S &\to \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \\ A &\to \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \\ C &\to \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \\ B &\to \mathtt{c} \mid \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \end{aligned}$$

1. típusú felesleges változók megtalálásához: $T_0 = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ $T_1 = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, S, A, B, C\} = T_2$. Mivel T_2 minden szimbólumot tartalmaz, ezért ilyen típusú felesleges nincs.

2. típus: $S_0 = \{S\}$, $S_1 = \{S, \mathtt{a}, A, B, \mathtt{b}, \mathtt{c}\} = S_2$. Tehát C felesleges, a nélküle kapott nyelvtan:

$$\begin{aligned} S &\to \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \\ A &\to \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \\ B &\to \mathtt{c} \mid \mathtt{a}A \mid B\mathtt{b} \mid A\mathtt{b} \mid \mathtt{cc} \end{aligned}$$

(Ebben már nincs a definíció értelmében felesleges változó, de kisebb nyelvtan ettől még lehetséges!)

5. Adjon meg olyan „majdnem CF" nyelvtant, amiben nincs egyszeres szabály, de az ε-szabályok kiküszöbölésekor keletkezik egyszeres szabály!

Megoldás: Például $S \to AB \qquad A \to \mathtt{a}A \mid \varepsilon \qquad B \to \mathtt{b}$ jó, mert $A \to \varepsilon$ miatt az átalakított nyelvtanban lesz $S \to B$ szabály is.

6. A tanult módon szüntesse meg a felesleges szimbólumokat az alábbi környezetfüggetlen nyelvtanban!

$$\begin{aligned} S &\to AB \mid AC \\ A &\to \mathtt{a}A\mathtt{b} \mid \mathtt{b}A\mathtt{a} \mid \mathtt{a} \\ B &\to \mathtt{b}AB \mid \mathtt{aa}B \mid B\mathtt{a}D \\ C &\to \mathtt{ab}C\mathtt{a} \mid \mathtt{a}BC \mid \mathtt{ba} \\ D &\to \mathtt{b}D \mid \mathtt{a}C\mathtt{a} \end{aligned}$$

Megoldás: Az 1. típusú felesleges szimbólumok (nem tűnnek el) kiszűrése:

$$T_0 = \{\mathtt{a}, \mathtt{b}\} \qquad T_1 = \{\mathtt{a}, \mathtt{b}, A, C\} \qquad T_2 = \{\mathtt{a}, \mathtt{b}, A, C, S, D\} = T_3,$$

azaz B felesleges, tehát minden B-t tartalmazó szabályt el kell hagyni. Ennek eredménye:

$$\begin{aligned} S &\to AC \\ A &\to \mathtt{a}A\mathtt{b} \mid \mathtt{b}A\mathtt{a} \mid \mathtt{a} \\ C &\to \mathtt{ab}C\mathtt{a} \mid \mathtt{ba} \\ D &\to \mathtt{b}D \mid \mathtt{a}C\mathtt{a} \end{aligned}$$

A 2. típusú felesleges szimbólumok (nem elérhetők) kiszűrése az előbb kapott nyelvtanból:

$$S_0 = \{S\} \qquad S_1 = \{S, A, C\} \qquad S_2 = \{S, A, C, \mathtt{a}, \mathtt{b}\} = S_3,$$

azaz D felesleges, tehát minden D-t tartalmazó szabályt el kell hagyni. A felesleges szimbólumok nélküli nyelvtan:

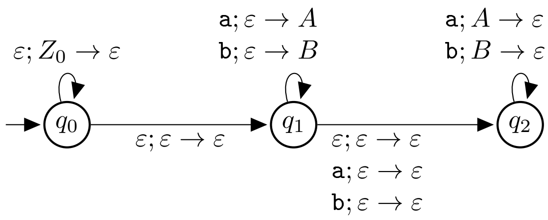
$$\begin{aligned} S &\to AC \\ A &\to \mathtt{a}A\mathtt{b} \mid \mathtt{b}A\mathtt{a} \mid \mathtt{a} \\ C &\to \mathtt{ab}C\mathtt{a} \mid \mathtt{ba} \end{aligned}$$

(Ebből egyszerűbb meghatározni a generált nyelvet is, ugye?)

7. Legyen $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. Adjon üres veremmel elfogadó veremautomatát a Σ feletti palindromok nyelvéhez! (Palindrom az, ami visszafelé olvasva is ugyanúgy néz ki, pl. ε, $\mathtt{aba}$, $\mathtt{aa}$.)

Megoldás: Az ötlet, hogy eleinte az olvasott karaktereket berakjuk a verembe (q_1 állapot), majd egy pont után, ha azt olvassuk, amit a verem tetejéről leszedtünk, akkor tovább lépünk, különben a számítás elakad (q_2 állapot). Azt, hogy mikor kell váltani nem kell tudnunk – erre jó a nemdeterminisztikus automata. Ahhoz, hogy a palindormoknál a végén kiürüljön a verem, érdemes még az elején a verem alját jelző szimbólumot kidobni (q_0 állapot). Ez azért is jó ötlet, mert az üres szó is palindrom.

Például három állapottal egy lehetséges megvalósítás, amiben azért, hogy a struktúra még világosabb legyen (és kevesebbet kelljen írni), a $q_0 \to q_1$ átmenetben semmi nem történik a bemeneten és a veremben. A $q_1 \to q_2$ átmenet viszont kezeli a páratlan hosszú palindromok esetét – mindegy, hogy a középső karakter micsoda.



Az üres szó esetén a verem kiürül q_0-ban, és bár tovább tud lépni q_1-be, majd q_2-be, a verem végig üres marad, tehát a számítás elfogadó. Nem üres szónál hiába ürül ki a verem q_0-ban, tovább kell lépnie, hiszen még nem olvasta el a szót. Ha nem a szó felénél lép át q_2-be, akkor vagy még a szó vége előtt kiürül a verem, és a számítás elakad, vagy nem ürül ki a verem és így egyik esetben sem fogad el. Szintén elakad ha ugyan jókor vált, de a szó nem palindrom.