

Formális nyelvek és fordítóprogramok alapjai

Formális nyelvek témakör jegyzete

Készült Nagy Sára előadásai
és Dévai Gergely gyakorlatai alapján

Sárközi Gergő, 2021-22-2. félév
Nincsen lektorálva!

Tartalomjegyzék

1. Előadás 1: Alapfogalmak és jelölések	3
1.1. Szavak műveletei	3
1.1.1. Részszó, szó prefix, szó szuffix	4
1.2. Nyelvek műveletei	4
2. Előadás 2: grammatika	5
2.1. Levezetés	5
2.2. Chomsky féle grammatika típusok	6
2.2.1. Grammatika típusuk implementálási igényei	7
2.3. Példa grammatika típusára	7
2.4. Nyelvtanok ekvivalenciája	7
2.5. Nyelvek típusai	7
2.6. Kényelmi jelölés	7
2.7. Bizonyítások	7
3. Előadás 3: reguláris műveletek és nyelvek	8
3.1. ϵ -mentesítés	8
3.2. Nyelvosztályok zártsága a reguláris műveletekre	9
3.3. Grammatikákon végzett reguláris műveletek	9
3.3.1. Unió, $\mathcal{G}_{0,2,3}$ típusú nyelvek esetén	9
3.3.2. Unió, \mathcal{G}_1 típusú nyelvek esetén	9
3.3.3. Konkatenáció, \mathcal{G}_3 típusú nyelvek esetén	9
3.3.4. Lezárt, \mathcal{G}_3 típusú nyelvek esetén	9
3.4. 3-as nyelvcsalád leírásai	10
3.5. Reguláris nyelvek	10

3.6.	Reguláris kifejezések	10
3.7.	Reguláris kifejezés gyakorlati jegyzet	11
3.8.	3-as típusú grammatikák normál formája	11
3.8.1.	Normál formára alakítás algoritmus	12
4.	Előadás 4: véges (nem) determinisztikus automaták	13
4.1.	Véges determinisztikus automata (VDA)	13
4.2.	Redukció	14
4.2.1.	Közvetlen redukció	14
4.2.2.	Általános redukció	14
4.3.	Automata által elfogadott nyelv	14
4.4.	Véges nemdeterminisztikus automata (VNDA)	15
4.5.	3-as típus nyelvek kapcsolata véges automatákkal	15
4.6.	VNDA-ból VDA	16
4.6.1.	Módszer VNDA-ból VDA konstruálásra	16
4.7.	Kleene tétele: $\mathcal{L}_3 = \mathcal{L}_{reg}$	17
4.7.1.	$\mathcal{L}_{VDA} \subseteq \mathcal{L}_{reg}$ bizonyítása	17
5.	Előadás 5: minimális automata	18
5.1.	Minimális véges determinisztikus automata	18
5.1.1.	$L \in \mathcal{L}_{reg}$ -et felismerő minimális VDA izomorfizmusig egyértelmű	18
5.1.2.	Összefüggő VDA	18
5.1.3.	Ekvivalens állapotok meghatározása	19
6.	Előadás 6: \mathcal{G}_2 típusú grammatikák, veremautomata	20
6.1.	Programozási nyelvek szintaxisa	20
6.1.1.	Szóprobléma	20
6.1.2.	Backus-Naur forma (BNF)	20
6.1.3.	Szintaxisfa	20
6.2.	Chomsky normál forma (\mathcal{G}_2)	21
6.3.	Lemmák $\mathcal{L}_3, \mathcal{L}_2$ kapcsán	22
6.3.1.	Kis Bar-Hillel lemma, szükséges feltétel $L \in \mathcal{L}_3$ -ra	22
6.3.2.	Kis Bar-Hillel lemma bizonyítása	22
6.3.3.	Példa és lemma alapján $\mathcal{L}_3 \subset \mathcal{L}_2$ belátása	22
6.3.4.	Bar-Hillel lemma (pumpáló lemma)	23
6.3.5.	Bar-Hillel lemma \implies van környezetfüggő nyelv	23
6.4.	Veremautomata	24

1. Előadás 1: Alapfogalmak és jelölések

- Két halmaz akkor egyenlő, ha egymásnak részhalmazai.
- Ábécé: jelek egy nem üres véges halmaza, jele: V
- Betű: ábécé elemei ($a \in V$)
- Szó: V feletti szó a V ábécé elemeinek véges sorozata (szó=sztring)
 - u szó hossza: $\ell(u)$ ($0 \leq \ell(u) < \infty$)
 - üres szó: ϵ ($\ell(\epsilon) = 0$)
- V^* : V ábécé feletti szavak halmaza
 - $\epsilon \in V^*$, de $V^+ = V^* \setminus \{\epsilon\}$
- Nyelv, formális nyelv: V^* egy bizonyos részhalmaza, jele: L
 - Létezik véges és végtelen nyelv
 - Üres szót tartalmazó nyelv: $\{\epsilon\}$
 - Üres nyelv: \emptyset
- Nyelvosztály, nyelvcsalád: nyelvek valamely halmaza
- Lexikografikus rendezés: először hosszúság szerint, utána ábécé szerint
 - Véges és végtelen nyelveket is el lehet kezdeni felsorolni így

1.1. Szavak műveletei

- Konkaténáció: szavak egymás után leírása
 - Nincs műveleti jel: csak egymás után leírjuk a szimbólumokat
 - Egységelem: ϵ
 - Asszociatív: $(uv)w = u(vw)$ ($u, v, w \in V^*$)
 - V^* zárt a konkaténáció műveletre
- Hatványozás: szó önmagával vett n -szeres konkaténációja
 - $u^0 = \epsilon$, $u^1 = u$ és ha $n \geq 1$: $u^n = u^{n-1}u$
- Szó megfordítása: betűk hátulról előre olvasva, jele: u^{-1}

1.1.1. Részszó, szó prefix, szó szuffix

- Részszó: v az u részszava, ha létezik x, y szó, hogy $u = xvy$
- Szó prefixe: v az u prefixe, ha létezik w szó, hogy $u = vw$
- Szó szuffixe: v az u szuffixe, ha létezik w szó, hogy $u = wv$
- "Valódi" jelentése: a teljes szó és az üres szó nem számít bele
 - Azaz $v \neq \epsilon$ és $v \neq u$

1.2. Nyelvek műveletei

- Reguláris műveletek: unió, konkatenáció, lezárás
- Két nyelv uniója: ha $L_1, L_2 \subseteq V^*$: $L_1 \cup L_2 = \{u \in V^* \mid u \in L_1 \vee u \in L_2\}$
 - Ha nem mindkettő V^* feletti: ábécé unióját is vesszük
 - Kommutatív, asszociatív, egységelemes (\emptyset)
- Két nyelv metszete: $L_1 \cap L_2 = \{u \in V^* \mid u \in L_1 \wedge u \in L_2\}$
- Egy nyelv komplementere egy ábécére vonatkozóan: $\overline{L} = V^* \setminus L$
- Egy nyelv tükörképe: $L^{-1} = \{u \in V^* \mid u^{-1} \in L\}$
- Két nyelv konkatenációja: $L_1 L_2 = \{uv \mid u \in L_1 \wedge v \in L_2\}$
 - Gyakorlatilag szorzat
 - Asszociatív, egységelemes ($\{\epsilon\}$)
 - Van nullelem: $\emptyset L = \emptyset = L \emptyset$
- Nyelv hatványozása: ismételt konkatenáció önmagával
 - $L^0 = \{\epsilon\}$, $L^1 = L$ és ha $n \geq 1$: $L^n = L^{n-1} L$
 - $\emptyset^0 = \{\epsilon\}$ (\emptyset minden más hatványa önmaga)
- Nyelv lezártja (iteráltja): $L^* = L^0 \cup L^1 \cup L^2 \cup \dots$
 - $L^+ = \bigcup_{i \geq 1} L^i$ ($\epsilon \in L^+ \Leftrightarrow \epsilon \in L$)

2. Előadás 2: grammatika

- Nyelv megadható szabályrendszerrel (de nem mindegyik)
- Grammatika (nyelvtan): $G = (N, T, P, S)$
 - N : nemterminális ábécé (szimbólumok, nagybetűsek)
 - T : terminálisok ábécéje (bemeneti szöveg ábécéje, kisbetűsek)
 - P : átírási szabályok véges halmaza
 - S : kezdőszimbólum ($S \in N$)
 - N és T diszjunkt halmazok
- Átírási szabályok
 - $p \rightarrow q$ alakúak, ahol $p \in (N \cup T)^* N (N \cup T)^*$ és $q \in (N \cup T)^*$
 - * Bal oldal tartalmaz min. 1 nemterminális szimbólumot
 - * Jobb oldal lehet üres
 - Mondatforma: $(N \cup T)^*$ (vegyes karakterek)
 - * Ha nem vegyes, akkor mondatforma is, de szó is

2.1. Levezetés

- v közvetlen levezethető u -ból ($u \Rightarrow_G v$), ha: $(u, v \in (N \cup T)^*)$
 - Létezik $u_1, u_2 \in (N \cup T)^*$ és $x \rightarrow y \in P$
 - Hogy: $u = u_1 x u_2$ és $v = u_1 y u_2$
- v közvetetten levezethető u -ból ($u \Rightarrow_G^* v$), ha: $(u, v \in (N \cup T)^*)$
 - Létezik $k \in \mathbb{N}$ és $x_0, \dots, x_k \in (N \cup T)^*$
 - Hogy: $u = x_0$ és $v = x_k$ és $\forall i \in [0, k-1] : x_i \Rightarrow_G x_{i+1}$
- Grammatika által generált nyelv
 - Minden olyan szó, ami (közvetetten) levezethető S -ből
 - $L(G) = \{u \in T^* \mid S \Rightarrow_G^* u\}$
- Példa: $G = (\{S, A, B\}, \{a, b\}, P, S)$
 - $P = \{S \rightarrow ASB, S \rightarrow AB, AB \rightarrow BA, A \rightarrow a, B \rightarrow b\}$
 - $L(G) = \{u \in \{a, b\}^* \mid \ell_a(u) = \ell_b(u) \geq 1\}$
 - Példa: $S \Rightarrow_G^* A^n B^n \Rightarrow_G^* A^{n-1} B A B^{n-1} \Rightarrow_G^* B A^n B^{n-1} \Rightarrow_G^* b a^n b^{n-1}$

2.2. Chomsky féle grammatika típusok

- A $G = (N, T, P, S)$ grammatikát P alapján lehet osztályozni
- $i = 0$ típus: nincs korlátozás (azaz ennyi: grammatikával megadható)
- $i = 1$ típus, környezetfüggő: P minden szabálya $u_1Au_2 \rightarrow u_1vu_2$ alakú, ahol $u_1, u_2, v \in (N \cup T)^*$, $A \in N$ és $v \neq \epsilon$, kivéve az $S \rightarrow \epsilon$ alakú szabály (Korlátozott ϵ szabály, KES), de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.
- $i = 2$ típus, környezetfüggetlen: P minden szabálya $A \rightarrow v$ alakú, ahol $A \in N, v \in (N \cup T)^*$
- $i = 3$ típus, reguláris grammatika: P minden szabálya vagy $A \rightarrow uB$ vagy $A \rightarrow u$ alakú, ahol $A, B \in N$ és $u \in T^*$
- Jelölje \mathcal{G}_i az i típusú grammatikák halmazát
 - $\mathcal{G}_i \subseteq \mathcal{G}_0$ ahol $i \in \{1, 2, 3\}$
 - $\mathcal{G}_3 \subseteq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subseteq \mathcal{G}_0$
 - * Hiányzó tartalmazás oka: ϵ -nal kapcsolatos kikötések
 - * Epszilon mentes grammatika esetén van tartalmazás ott is
- Alábbi táblázat sorai egymásba alakítható grammatikák

Típus	Alaptípus szabályai	Speciális alakok szabályai	Normál forma szabályai
0.	Nincs korlátozás.	$p \rightarrow q$, ahol $p \in N^*, q \in (N \cup T)^*$	
1.	$u_1Au_2 \rightarrow u_1vu_2$, ahol $u_1, u_2, v \in (N \cup T)^*, A \in N$, és $v \neq \epsilon$, kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem. (környezetfüggő grammatika)	$p \rightarrow q$, ahol $l(p) \leq l(q)$ kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem. (hosszúság nemcsökkentő grammatika)	Kuroda normál forma $A \rightarrow a$ vagy $A \rightarrow B$ vagy $A \rightarrow BC$ vagy $AB \rightarrow CD$ alakúak a szabályok, ahol $a \in T$ és $A, B, C, D \in N$, kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.
2.	$A \rightarrow v$, ahol $v \in (N \cup T)^*, A \in N$ (környezetfüggetlen grammatika)	$A \rightarrow v$, ahol $v \in (N \cup T)^*, A \in N$ és $v \neq \epsilon$, kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.	Chomsky normál forma $A \rightarrow a$ vagy $A \rightarrow BC$ alakúak a szabályok, ahol $a \in T$ és $A, B, C \in N$, kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.
3.	$A \rightarrow uB$ vagy $A \rightarrow u$, ahol $u \in T^*, A, B \in N$ (reguláris grammatika)	$A \rightarrow aB$ vagy $A \rightarrow a$, ahol $a \in T$, és $A, B \in N$, kivéve az $S \rightarrow \epsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.	3-as normál forma $A \rightarrow aB$ vagy $A \rightarrow a$, ahol $a \in T$, és $A, B \in N$.

2.2.1. Grammatika típusuk implementálási igényei

- 3-as típusú grammatika verem nélkül is implementálható
- 2-es nyelvcsaládhoz már kell 1 verem
- Két veremmel már bármi lehetséges (0-as nyelvcsalád)

2.3. Példa grammatika típusára

- Kérdés $G = (\{S', S, A, B\}, \{a, b\}, P, S')$ grammatika típusa

$$\bullet P = \begin{cases} S' \rightarrow \epsilon & \text{KES} \\ S' \rightarrow S & 0,1,2,3 \text{ típusú szabály} \\ S \rightarrow ASB & 0,1,2 \text{ típusú szabály} \\ S \rightarrow AB & 0,1,2 \text{ típusú szabály} \\ AB \rightarrow BA & 0 \text{ típusú szabály (TODO miért nem 1?)} \\ A \rightarrow a & 0,1,2,3 \text{ típusú szabály} \\ B \rightarrow b & 0,1,2,3 \text{ típusú szabály} \end{cases}$$

- Tehát G grammatika 0-s típusú, $G_1 \in \mathcal{G}_0$

2.4. Nyelvtanok ekvivalenciája

- G_1 és G_2 ekvivalensek, ha $L(G_1) = L(G_2)$
- Gyengén ekvivalensek, ha $L(G_1) \setminus \{\epsilon\} = L(G_2) \setminus \{\epsilon\}$

2.5. Nyelvek típusai

- Egy L nyelv i típusú, ha $\exists i$ -típusú grammatika, ami generálja
- Nyelvcsalád: \mathcal{L}_i jelöli az i -típusú nyelvek halmazát
- Chomsky féle hierarchia: $\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0$

2.6. Kényelmi jelölés

- $S \rightarrow A \mid B$ jelentése: két szabály, $S \rightarrow A$ és $S \rightarrow B$

2.7. Bizonyítások

- Indukcióval

3. Előadás 3: reguláris műveletek és nyelvek

- Nyelvtani transzformáció: eljárás, ami G -ből G' -t csinál
 - Ekvivalens transzformáció, ha $L(G) = L(G')$

3.1. ϵ -mentesítés

- Tétel:
 - $G = (N, T, P, S)$ legyen környezetfüggetlen (2-es típusú)
 - Csinálható vele ekvivalens $G' = (N', T, P', S')$
 - * ami szintén környezetfüggetlen
 - * amiben nincs $A \rightarrow \epsilon$ alakú szabály, kivéve, ha $\epsilon \in L(G)$, ekkor $S' \rightarrow \epsilon \in P'$, de ekkor S' nem szerepelhet szabály jobboldalán
- Eljárás:
 - Első lépés: mely nemterminálisokból vezethető le az ϵ ?
 - * $H = \{A \in N \mid A \Rightarrow_G^* \epsilon\} = ?$
 - * $H_1 = \{A \in N \mid \exists A \rightarrow \epsilon \in P\}$
 - * $H_{i+1} = H_i \cup \{A \in N \mid \exists A \rightarrow w \in P \wedge w \in H_i^*\}$
 - * $H_1 \subset H_2 \subset \dots H_k = H_{k+1} \implies H = H_k$
 - * Nemterminálisok halmaza véges, tehát az eljárás véges
 - * Belátható: $\epsilon \in L(G) \Leftrightarrow S \in H$
 - Második lépés, $S \notin H$ esetén: $A \rightarrow v' \in P'$ akkor és csak akkor ha $v' \neq \epsilon$ és ha $\exists A \rightarrow v \in P$, hogy v -ből v' -t úgy kapunk, hogy elhagyunk nulla vagy több H -beli nemterminálist v -ből
 - Második lépés, $S \in H$ esetén:
 - * Előbb felsorolt szabályok közül az összes
 - * Új kezdőszimbólum: S' és $S' \notin N$
 - * Két extra szabály: $S' \rightarrow \epsilon$ és $S' \rightarrow S$

3.2. Nyelvosztályok zártsága a reguláris műveletekre

- Tétel: a $\mathcal{L}_{0..3}$ nyelvosztályok mindegyike zárt a reguláris műveletekre
- Emlékeztető: reguláris műveletek: unió, konkatenáció, lezárás

3.3. Grammatikákon végzett reguláris műveletek

3.3.1. Unió, $\mathcal{G}_{0,2,3}$ típusú nyelvek esetén

- Legyen $G = (N, T, P, S)$ és $G' = (N', T, P', S')$ azonos típusúak
- Legyen $N \cap N' = \emptyset$ és S_0 új szimbólum, $S_0 \notin (N \cup N')$
- $G_{\cup} = (N \cup N' \cup \{S_0\}, T, P \cup P' \cup \{S_0 \rightarrow S, S_0 \rightarrow S'\}, S_0)$
- Ekkor G_{\cup} típusa megegyzik G, G' típusával és $L(G) \cup L(G') = L(G_{\cup})$

3.3.2. Unió, \mathcal{G}_1 típusú nyelvek esetén

- Ha $\epsilon \in L(G) \cup L(G')$, akkor az előző G_{\cup} -ban nem teljesül a KES
- G_{\cup} készítésekor vegyük ki P -ből és P' -ből a $S \rightarrow \epsilon, S' \rightarrow \epsilon$ szabályokat
- G_{\cup} kezdőszimbóluma legyen az új S_1 szimbólum
- G_{\cup} szabályai közé vegyük be az alábbiakat: $S_1 \rightarrow \epsilon, S_1 \rightarrow S_0$

3.3.3. Konkatenáció, \mathcal{G}_3 típusú nyelvek esetén

- P -ből csinálunk P_1 -et: $A \rightarrow u$ alakúakat lecseréljük $A \rightarrow uS'$ -re
- $G_C = (N \cup N', T, P_1 \cup P', S)$ és $G_C \in \mathcal{G}_3$ és $L(G_C) = L(G)L(G')$

3.3.4. Lezárt, \mathcal{G}_3 típusú nyelvek esetén

- Legyen S_0 új szimbólum ($S_0 \notin N$)
- P -ből csinálunk P_1 -et: $A \rightarrow u$ alakúakat lecseréljük $A \rightarrow uS_0$ -ra
- $G_* = (N \cup \{S_0\}, T, P_1 \cup P \cup \{S_0 \rightarrow \epsilon, S_0 \rightarrow S\}, S_0)$
- $G_* \in \mathcal{G}_3$ és $L(G_*) = L^*$

3.4. 3-as nyelvcsalád leírásai

- \mathcal{G}_3 nyelvei leírhatóak:
 - 3-as típusú grammatikával
 - reguláris kifejezéssel
 - véges determinisztikus automatával
 - véges nemdeterminisztikus automatával
- Bebizonyítható: $\mathcal{L}_3 = \mathcal{L}_{reg} = \mathcal{L}_{VDA} = \mathcal{L}_{VNDA}$

3.5. Reguláris nyelvek

- Rekurzív definíció:
 - Elemi nyelvek: \emptyset , $\{\epsilon\}$ és $\{a\}$, ahol a egy tetszőleges betű
 - Reguláris nyelv: elemi nyelvekből reguláris műveletekkel létrehozható
 - Nincs más reguláris nyelv
- Tétel: $\mathcal{L}_{reg} \subseteq \mathcal{L}_3$, bizonyítás:
 - Elemi nyelvekhez megadható 3-as típusú grammatika:
 $G = (\{S\}, \{a\}, \{S \rightarrow aS\}, S)$, $G = (\{S\}, \{a\}, \{S \rightarrow \epsilon\}, S)$,
 $G = (\{S\}, \{a\}, \{S \rightarrow a\}, S)$
 - \mathcal{L}_3 zárt a reguláris műveletekre
 - Reguláris nyelv def alapján készíthető azonos 3-as nyelv is

3.6. Reguláris kifejezések

- Definíció:
 - Elemi reguláris kifejezések: \emptyset , ϵ , a , ahol a egy tetszőleges betű
 - Ha R_1 és R_2 reguláris kifejezések, akkor az alábbiak is:
 $(R_1|R_2)$, (R_1R_2) , $(R)^*$
 - A reguláris kifejezések halmaza a legszűkebb halmaz, amire a felső két pont teljesül.
- Műveletek prioritása csökkenő sorrendben: lezárás, konkatenáció, unió
 - A zárójelek ennek megfelelően elhagyhatók

3.7. Reguláris kifejezés gyakorlati jegyzet

- $[a - 9]$: ha 9 előbb van, akkor üres
- $.$ (pont) az bármi, kivéve $\backslash n$, de $[\wedge a]$ match-el $\backslash n$ -re
- Be lett vezetve: $a\{5\}$ (de min, max, tól-ig változatai nem)
- Escaping: $"$ közé, vagy \backslash
- Üresség: $""$ (kettő $"$) (azaz $a? \equiv ""|a$)
- Halmazban $([...])$ escape-elni kell a $-$ karaktert, de mást nem

3.8. 3-as típusú grammatikák normál formája

- Tétel: minden 3-as típusú nyelv generálható ilyen szabályokkal:
 - $A \rightarrow aB$, ahol $A, B \in N$ és $a \in T$
 - $A \rightarrow \epsilon$ ahol $A \in N$
- Ez a normál forma
- Ebből könnyen készíthető automata

3.8.1. Normál formára alakítás algoritmusa

- $G = (N, T, P, S) \rightarrow G' = (N', T, P', S)$
 - G 3-as típusú grammatika
 - G' 3-as normál formájú grammatika
- Hosszredukció
 - $A \rightarrow a_1 \dots a_k B$ alakú szabályok helyett (ahol $k \geq 2$, B lehet ϵ is)
 - $A \rightarrow a_1 Z_1$ ahol $Z_1 \notin N$ és $Z_1 \rightarrow A_2 Z_2$ ahol $Z_2 \notin (N \cup Z_1)$, stb.
 - Azaz Z_k mindig egy új nemterminális
 - $Z_{k-1} \rightarrow a_k B$
- Befejező szabályok átalakítása
 - Legyen E egy minden szabályra közös új nemterminális
 - $A \rightarrow a$ alakú szabályok helyett (ahol $a \in T$ és $A \in N$)
legyen P' része: $A \rightarrow aE$ és $E \rightarrow \epsilon$
- Láncmentesítés
 - $A \rightarrow B$ alakú szabályok helyettesítése
 - Ismert módszerrel meghatározzuk: $H(A) = \{B \in N \mid A \Rightarrow_G^* B\}$
 - * $H_1(A) = \{A\}$
 - * $H_{i+1}(A) = H_i(A) \cup \{B \in N \mid \exists C \in H_i(A) \wedge C \rightarrow B \in P\}$
 - Majd P' -be felvesszük az $A \rightarrow X$ szabályokat,
ha $\exists B \in H(A)$, hogy $B \rightarrow X \in P$ ahol $X \in (T \cup N)^* \setminus N$
 - * Azaz ahol $X \in (T \cup N)^*$ és nem csak egyetlen nemterminális

4. Előadás 4: véges (nem) determinisztikus automaták

4.1. Véges determinisztikus automata (VDA)

- Definíció: $A = (Q, T, \delta, q_0, F)$ a VDA, ahol
 - Q : állapotok nemüres VÉGES halmaza
 - T : input szimbólumok ábécéje (véges, nemüres)
 - $\delta: Q \times T \rightarrow Q$ leképezés: állapot-átmenti függvény
 - * Determinisztikus, egyértelmű
 - * Minden $q, a \in Q \times T$ párra értelmezett (különben parciális)
 - * Érdemes táblázattal megadni: bemenetek az oszlopok
 - $q_0 \in Q$: kezdőállapot
 - $F \subseteq Q$: elfogadóállapotok (lehet üres)
 - * Ha nem ilyen állapotnál fog el a bemenet, akkor rossz szó
- Irányított gráffal megadható: csúcspont az állapot, δ alapján élek
 - Kezdőállapot jelölése: semmiből induló nyíl vezet bele
 - Elfogadóállapotok (F) jelölése: dupla karika
- Az egész automata megadható δ -t leíró táblázattal, plusz két extra:
 - Befelé nyíl a kezdőállapotnál
 - Kifelé nyíl az elfogadóállapotoknál
- Beolvasáskor elég a bemenet (szalag) és 1 változó (aktuális állapot)
 - Azaz itt még nincs verem
- Alternatív jelölés állapot-átmenetekre: $\delta(q, a) = p$ helyett $qa \rightarrow p$
- Konfiguráció: aktuális állapot és a maradék bemenet
- Automata mindig terminál, mert a bemenet véges
- Gyakorlaton konvenció: betűvel nem lehet lépni \implies a betűvel a hibaállapota jutunk
 - Hibaállapot: onnan nem mehetünk másikba és nem elfogadó állapot
- Gyakorlat: nem muszáj az állapotokat elnevezni

4.2. Redukció

4.2.1. Közvetlen redukció

- Legyen $A = (Q, T, \delta, Q_0, F)$ egy véges automata
- A az $u \in QT^*$ konfigurációt a $v \in QT^*$ konfigurációra redukálja közvetlenül...
 - Jelölés: $u \Rightarrow_A v$
- ... ha létezik $qa \rightarrow p$ szabály (azaz $\delta(q, a) = p$)
- ... és ha $\exists w \in T^* : u = qaw \wedge v = pw$

4.2.2. Általános redukció

- Legyen $A = (Q, T, \delta, Q_0, F)$ egy véges automata
- A az $u \in QT^*$ konfigurációt a $v \in QT^*$ konfigurációra redukálja...
 - Jelölés: $u \Rightarrow_A^* v$
- ... ha $u = v$ vagy $\exists z \in QT^* : u \Rightarrow_A^* z \wedge z \Rightarrow_A v$

4.3. Automata által elfogadott nyelv

- Legyen $A = (Q, T, \delta, q_0, F)$
- A által elfogadott nyelv: $L(A) = \{u \in T^* \mid \exists p \in F : q_0u \Rightarrow_A^* p\}$
- VNDA esetén: $q_0 \in Q_0$ a halmaz definícióban

4.4. Véges nemdeterminisztikus automata (VNDA)

- Determinisztikus autotává lehet alakítani (de annak általánosítása is)
- Különbség: $q_0 \in Q$ helyett $Q_0 \subseteq Q$
- Különbség: $\delta : Q \times T \rightarrow Q$ helyett $\delta : Q \times T \rightarrow \mathcal{P}(Q)$
 - $\mathcal{P}(Q)$ jelentése: Q részhalmazainak a halmaza (azaz Q hatványhalmaza)
 - Azaz $\mathcal{P}(Q)$ az a Q hatványhalmaza, ami egy véges halmaz
 - Q hatványhalmazába képez, ez egy véges halmaz
 - Táblázat celláiban lehet 0, 1, vagy több állapot is
- Egy szó akkor jó, ha létezik olyan lefutás, amikor el van fogadva
 - Azaz ha nem determinisztikus esetben "rossz" döntést hoztunk, de más döntés esetén jó lett volna a szó, akkor a szó az jó
- Redukciók, elfogadott nyelv definíció gyakorlatilag megegyezik

4.5. 3-as típus nyelvek kapcsolata véges automatákkal

- Tétel: $\mathcal{L}_3 \subseteq \mathcal{L}_{VNDA}$ és $\mathcal{L}_{VNDA} \subseteq \mathcal{L}_3$
- Bizonyítás: $L \in \mathcal{L}_3$ megadása $A = (Q, T, \delta, Q_0, F)$ VNDA-val
 - Tudjuk, hogy L megadható $G = (N, T, P, S) \in \mathcal{G}_3$ grammatikával
 - Minden nemterminális legyen egy állapot: $Q = \{q_X \mid X \in N\}$
 - Kezdőállapot: $Q_0 = q_S = \{S\}$
 - Legyen $\delta(q_a, a) = q_B \Leftrightarrow (A \rightarrow aB) \in P$
 - Legyen $q_A \in F \Leftrightarrow (A \rightarrow \epsilon) \in P$
 - Megjegyzés: eredmény nem feltétlenül determinisztikus automata
 - Megjegyzés: $S \Rightarrow_G^* u \Leftrightarrow q_S \Rightarrow_A^* u$
- A másik irány bizonyítása hasonló

4.6. VNDA-ból VDA

- Tétel: $\mathcal{L}_{VNDA} \subseteq \mathcal{L}_{VDA}$
- Másik irány triviális definíció alapján

4.6.1. Módszer VNDA-ból VDA konstruálásra

- Alapötlet: legyenek az állapotok maguk is halmazok
 - Üres halmaz is egy állapot
- Legyen $Q' = \mathcal{P}(Q)$, azaz Q részhalmazainak a halmaza (hatványhalmaz)
- Legyen $\delta' : Q' \times T \rightarrow Q'$ így definiálva:
 - $\delta'(q', a) = \bigcup_{q \in q'} \delta(q, a)$ ahol $q' \in Q'$ és $a \in T$
- Legyen $q'_0 = Q_0$
- Legyen $F' = \{q' \in Q' \mid q' \cap F \neq \emptyset\}$
- Bizonyítás: $L(A) \subseteq L(A')$
 - Lemma 1: $\forall q, p \in Q; q' \in Q'; u, v \in T^*$ esetén
ha $qu \Rightarrow_A^* pv \wedge q \in q'$ akkor $\exists p' \in Q' : q'u \Rightarrow_A^* p'v \wedge p \in p'$
 - Maradék bizonyítás: fny_4.pdf, 28. oldal
- Bizonyítás: $L(A') \subseteq L(A)$
 - fny_4.pdf, 29. oldal

4.7. Kleene tétele: $\mathcal{L}_3 = \mathcal{L}_{reg}$

- Bizonyítás vázlat:
 - $\mathcal{L}_{reg} \subseteq \mathcal{L}_3$ (már beláttuk)
 - $\mathcal{L}_3 = \mathcal{L}_{VDA}$ (már beláttuk)
 - $\mathcal{L}_{VDA} \subseteq \mathcal{L}_{reg}$

4.7.1. $\mathcal{L}_{VDA} \subseteq \mathcal{L}_{reg}$ bizonyítása

- Legyen A egy n állapotú VDA: $Q = \{q_1, q_2, \dots, q_n\}$ és q_1 a kezdőállapot
- $q_i u \Rightarrow_A^* q_j$ redukció
 - Érinti a q_m állapotot, ha q_m egy közbülső lépés része
 - k -megszorított, ha csak q_1 és q_k közötti állapotokat érint (i és j nincs megszorítva, csak a közbülső lépések)
- Milyen szavakba lehet eljutni k megszorítással?
 - $0 \leq k \leq n$ és $1 \leq i, j \leq n$
 - $E_{i,i}^0 = \{\epsilon\} \cup \{a \in T \mid \exists q_i a \rightarrow q_i\}$
 - $E_{i,j}^0 = \{a \in T \mid \exists q_i a \rightarrow q_j\}$ ahol $i \neq j$
 - $E_{i,j}^k = \{u \in T^* \mid \exists q_i u \Rightarrow_A^* q_j \text{ k-megszorított redukció}\}$
 - $E_{i,j}^k = E_{i,j}^{k-1} \cup E_{i,k}^{k-1} (E_{k,k}^{k-1})^* E_{k,j}^{k-1}$
- Automata által elfogadott állapotok: $L(A) = \bigcup_{i \in I} E_{1,i}^n$
 - Ahol I az elfogadó állapotok indexe
 - Ez csak reguláris műveleteket használ: unió, konkatenáció, lezárás

5. Előadás 5: minimális automata

5.1. Minimális véges determinisztikus automata

- Egy A VDA minimális, ha állapotainak száma minimális: nincs olyan A' VDA, hogy $L(A) = L(A')$, de A' állapotainak száma kevesebb

5.1.1. $L \in \mathcal{L}_{reg}$ -et felismerő minimális VDA izomorfizmusig egyértelmű

- L reguláris nyelvet felismerő minimális VDA: izomorfizmus erejéig egyértelmű
- Bizonyítás lépései:
 - Automata összefüggővé tétele: nem elérhető állapotok elhagyása
 - Ekvivalens állapotok meghatározása
- Elkészített véges automata: $A' = (Q', T, \delta', q'_0, F')$
 - Az állapotok partíciókra cseréljük le (partíciókról később)
 - $Q' = B_i$ partíciók
 - $q'_0 = q_0$ -t tartalmazó partíció
 - $F' = F$ -ből keletkezett partíciók
 - $\delta'(B_i, a) = B_j \iff \delta(q, a) = p \wedge q \in B_i \wedge p \in B_j$

5.1.2. Összefüggő VDA

- $A = (Q, T, \delta, q_0, F)$ VDA q állapota elérhető, ha $\exists u \in T^* : q_0 u \Rightarrow_A^* q$
- Egy VDA összefüggő, ha minden állapota elérhető q_0 -ból
- Elérhető állapotok meghatározása:
 - $H_0 = \{q_0\}$
 - $H_{i+1} = H_i \cup \{r \in Q \mid \delta(q, a) = r; q \in H_i; a \in T\}$
 - $H = H_k$, ahol $H_k = H_{k+1}$
- Tehát a $Q \setminus H$ állapotok elhagyhatóak: nem elérhetőek

5.1.3. Ekvivalens állapotok meghatározása

- q és p ekvivalens ($q \sim p$), ha minden szóra egyformán működnek
 - Azaz: $\forall u \in T^* : (qu \Rightarrow_A^* r \wedge pu \Rightarrow_A^* r') \implies (r \in F \Leftrightarrow r' \in F)$
- Állítás: $(q \sim p \wedge qa \rightarrow s \wedge pa \rightarrow t) \implies s \sim t$
 - Azaz ha két ekvivalens állapotból azonos betűvel léptünk, akkor ekvivalens állapotokba jutunk.
- q és p i -ekvivalens állapotok ($q \sim^i p$), ha minden max i hosszú szóra egyformán működnek
 - $\forall u \in T^*, \ell(u) \leq i : (qu \Rightarrow_A^* r \wedge pu \Rightarrow_A^* r') \implies (r \in F \Leftrightarrow r' \in F)$
 - Lemma: $q \sim^{i+1} p \Leftrightarrow (\forall a \in T : (qa \rightarrow r \wedge pa \rightarrow t) \implies r \sim^i t)$
 - $q \sim^0 p \Leftrightarrow (q, p \in F \vee q, p \in Q \setminus F)$
- Particionálás folyamata
 - Folyamatosan egyre hosszabb szavak alapján particionálunk
 - ϵ , először két rész: $Q = B_1 \cup B_2$, $B_1 = F$, $B_2 = Q \setminus F$
 - Finomítás:
 - * Legyen $q \sim^i p$ és a partíciók száma $k \geq 2$ ($Q = B_1 \cup \dots \cup B_k$)
 - * $p, q \in B_i$ pontosan akkor maradnak egy partícióban, ha:
 $\forall a \in T : (qa \rightarrow r \wedge pa \rightarrow t) \implies (r, t \in B_i)$
 azaz ha r, t azonos partícióba tartoznak, ilyenkor $q \sim^{i+1} p$
 - * Ellenkező esetben a B_i partíciót szét kell bontani
 - * Ezt az eljárást addig ismételjük, amíg változás van

6. Előadás 6: \mathcal{G}_2 típusú grammatikák, veremautomata

6.1. Programozási nyelvek szintaxisa

- Programozási nyelvek szintaxisa: környezetfüggetlen (\mathcal{G}_2)

6.1.1. Szóprobléma

- Tétel: $\forall G \in \mathcal{G}_2$ eldönthető, hogy $u \in T^*$ esetén $u \in L(G)$ igaz-e
 - Nem bizonyítjuk

6.1.2. Backus-Naur forma (BNF)

- Szabály bal-jobb oldal elválasztó: $::=$
- Nemterminális: $<$ és $>$ jelek között (minden más terminális)
- $\langle \text{kifejezés} \rangle ::= \langle \text{tag} \rangle \mid \langle \text{tag} \rangle + \langle \text{kifejezés} \rangle$
 $\langle \text{tag} \rangle ::= \langle \text{faktor} \rangle \mid \langle \text{faktor} \rangle * \langle \text{tag} \rangle$
 $\langle \text{faktor} \rangle ::= i \mid (\langle \text{kifejezés} \rangle)$
 - Pár helyes kifejezés: $i + i * i$, $(i + i) * i$, $i * i * i * i$, $((i))$, i

6.1.3. Szintaxisfa

- Legyen $G = (N, T, P, S) \in \mathcal{G}_2$
- A t nemüres fa szintaxisfa, ha...:
 - Belső pontjai N elemeivel vannak címkézve
 - Levelei $T \cup \{\epsilon\}$ elemeivel vannak címkézve
 - * ϵ -nal címkézett pontoknak nincs testvére
 - Ha egy belső pont címkéje A és közvetlen gyerekei X_1, X_2, \dots, X_n , akkor $A \rightarrow X_1 X_2 \dots X_n \in P$
- Szóhoz megadható szintaxisfa \Leftrightarrow létezik levezetés
- Egyértelmű grammatika: $\forall u \in L(G)$ szóhoz egyetlen szintaxisfa tartozik
 - Nem egyértelmű példa: $S \rightarrow a \mid S + S$ az $u = a + a + a$ esetben

6.2. Chomsky normál forma (\mathcal{G}_2)

- Emlékeztető: $A \rightarrow a$ vagy $A \rightarrow BC$ a szabályok ($a \in T$; $A, B, C \in N$)
 - Vagy $S \rightarrow \epsilon$, de ekkor S csak bal oldalon van
- Bináris szintaxisfák jönnek ki ilyen grammatikákból
- Aktív nemterminálisok halmaza
 - Jelentése: valahány lépésben csupa terminális csinálható belőle
 - $A = \{X \in N \mid X \Rightarrow_G^* u \wedge u \in T^*\}$
 - Inaktív nemterminálisok halmaza: $N \setminus A$
 - $A = \emptyset \implies L(G) = \emptyset$
- Elérhető nemterminálisok halmaza:
 - Jelentése: kezdőállapotból el lehet bele jutni
 - $R = \{X \in N \mid S \Rightarrow_G^* uXw \wedge u, w \in (T \cup N)^*\}$
 - * Nem lehet üres: S mindig benne van
 - Nem elérhető nemterminálisok halmaza: $N \setminus R$
- Hasznos nemterminálisok: aktív és elérhető ($A \cap R$)
- Redukált grammatika: minden nemterminális hasznos ($N = A \cap R$)

6.3. Lemmák \mathcal{L}_3 , \mathcal{L}_2 kapcsán

6.3.1. Kis Bar-Hillel lemma, szükséges feltétel $L \in \mathcal{L}_3$ -ra

- Minden $L \in \mathcal{L}_3$ nyelvhez van $n \geq 1$ nyelvfüggő konstans
- Hogy $\forall u \in L$ ahol $\ell(u) \geq n$ van olyan $u = xyz$ felbontás
 - $\ell(xy) \leq n$
 - $y \neq \epsilon$
 - $\forall i \geq 0 : xy^iz \in L$

6.3.2. Kis Bar-Hillel lemma bizonyítása

- $L \in \mathcal{L}_3 \implies$ adható minimális VDA
- Tegyük fel, hogy az automatának n állapota van
- Szó legalább n hosszú \implies legalább n hosszú prefixének olvasásakor legalább egy állapot kétszer volt érintve, azaz tettünk egy kört
 - Ezt a kört kihagyhatjuk ($i=0$) vagy akárhányszor ismételtethetjük

6.3.3. Példa és lemma alapján $\mathcal{L}_3 \subset \mathcal{L}_2$ belátása

- $L = \{a^k b^k \mid k \geq 0\} \notin \mathcal{L}_3$ bizonyítása
- Tegyük fel indirekt, hogy $\exists n \geq 1$ a lemma szerint
- Legyen $u = a^k b^k$ ahol $k > n$
- Ekkor léteznie kéne egy $u = xyz$ felbontásnak ahol $\ell(xy) \leq n$ és $y \neq \epsilon$
- De $k > n$ miatt y csak 'a' betűket tartalmazhat
- Lemma szerint $a^{k+j} b^k$ szónak is $\in L$, de ez nem igaz

6.3.4. Bar-Hillel lemma (pumpáló lemma)

- Minden környezetfüggetlen nyelvhez megadható p, q
- $\forall u \in L : \ell(u) > p \implies u = vxyz$
 - $v, x, w, y, z \in T^*$
 - $\ell(xwy) \leq q$
 - $xy \neq \epsilon$
 - $\forall i \geq 0 : vx^iwy^iz \in L$
- Párhuzamos pumpálás szemléletesen
- Nem bizonyítjuk (de Chomsky-normálforma, bináris fa alapján kell)

6.3.5. Bar-Hillel lemma \implies van környezetfüggő nyelv

- Példa: $L = \{a^n b^n c^n \mid n > 0\} \notin \mathcal{L}_2$
- Bizonyítás: indirekt, tegyük fel, hogy létezik p és q
- Legyen $k > p$ és $k > q$, ekkor $u = a^k b^k c^k$ és $\ell(u) > p$
- Tétel szerint x és y párhuzamosan iterálható, de xy -ban nem lehet mindhárom betűből

6.4. Veremautomata

- $A = (Z, Q, T, \delta, z_0, q_0, F)$
 - Z : verem szimbólumok ábécéje (nem üres)
 - Q : állapotok halmaza (véges és nem üres)
 - T : bemeneti szimbólumok ábécéje
 - $\delta : Z \times Q \times (T \cup \{\epsilon\}) \rightarrow \mathcal{P}(Z^* \times Q)$
 - * Lehet determinisztikus vagy nem determinisztikus
 - * Verem tetejét mindig kiveszi (de vissza is rakhatja, akár többet)
 - * Bár Z^* lehet végtelen, mi csak véges sok szabályt engedünk
 - $z_0 \in Z$: kezdő veremszimbólum (verem nem üresen kezd)
 - $q_0 \in Q$: kezdőállapot
 - $F \subseteq Q$: elfogadó állapotok halmaza
- VDA-tól eltérés: Z , z_0 és δ
- Elfogadó állapotba kerüléskor nem feltétlenül üres a verem
- Verem üres \implies (szó helyes \Leftrightarrow elfogadóállapotban vagyunk)
 - Bemenetből olvasható ϵ , de veremből nem!
- Nem feltétlenül terminál (ϵ -t olvashat bemenetről)

