

Materiál na štátnicu

Formálne jazyky a automaty

Verzia 1

Obsah.

8.1. ÚVOD DO PROBLEMATIKY.....	3
8.1.1. ZÁKLADNÉ POJMY.....	3
8.1.2. ELEMENTÁRNE OPERÁCIE NAD SLOVAMI.....	3
8.1.3. ELEMENTÁRNE OPERÁCIE NAD JAZYKMI.....	3
8.1.4. GRAMATIKY.....	3
8.1.5. PROCEDURÁLNA REPREZENTÁCIA JAZYKA.....	4
8.2. REGULÁRNE GRAMATIKY A KONEČNÉ AUTOMATY.....	4
8.2.1. REGULÁRNE GRAMATIKY.....	4
8.2.2. KONEČNÉ AUTOMATY.....	4
8.2.3. NEDETERMINISTICKÉ AUTOMATY.....	4
8.2.4. MYHILL-NERODOVA VETA.....	5
8.2.5. APLIKÁCIE MYHILL-NERODOVEJ VETY.....	6
8.2.6. MINIMALIZÁCIA DETERMINISTICKÉHO KONEČNÉHO AUTOMATU.....	6
8.2.7. REGULÁRNE VÝRAZY.....	6
8.2.8. UZÁVEROVÉ VLASTNOSTI.....	7
8.3. BEZKONTEXTOVÉ GRAMATIKY A ZÁSOBNÍKOVÉ AUTOMATY.....	8
8.3.1. BEZKONTEXTOVÉ GRAMATIKY.....	8
8.3.2. REDUKOVANÉ GRAMATIKY.....	8
8.3.3. NAJĽAVEJŠIE ODVODENIE.....	8
8.3.4. CHOMSKÉHO A GREIBACHOVEJ NORMÁLNY TVAR.....	8
8.3.5. STROMY ODVODENIA.....	9
8.3.6. PUMPOVACIA LEMA.....	9
8.3.7. ZÁSOBNÍKOVÉ AUTOMATY.....	10
8.3.8. UZÁVEROVÉ VLASTNOSTI.....	10
8.4. FRÁZOVÉ GRAMATIKY A TURINGOVE STROJE.....	11
8.4.1. TURINGOVE STROJE.....	11
8.4.2. MODIFIKOVANÉ TURINGOVE STROJE.....	11
8.4.3. REKURZÍVNE VYPOČÍTEĽNÉ JAZYKY.....	11
8.4.4. FRÁZOVÉ GRAMATIKY.....	11
8.4.5. CHURCHOVA TÉZA.....	12
8.5. KONTEXTOVÉ GRAMATIKY A LINEÁRNE OHRANIČENÉ AUTOMATY.....	12
8.5.1. KONTEXTOVÉ GRAMATIKY.....	12
8.5.2. LINEÁRNE OHRANIČENÉ AUTOMATY.....	12
8.5.3. CHOMSKÉHO HIERARCHIA GRAMATÍK.....	13
8.5.4. UZÁVEROVÉ VLASTNOSTI.....	14
8.6. APLIKÁCIE TURINGOVÝCH STROJOV.....	15
8.6.1. TURINGOVE STROJE, POČÍTAJÚCE CELOČÍSELNÉ FUNKCIE.....	15
8.6.2. TURINGOVE STROJE AKO ENUMERÁTORY.....	15
8.7. ROZHODNUTEĽNOSŤ.....	15
8.7.1. NIEKTORÉ ŠTANDARDNÉ PROBLÉMY.....	15
8.7.2. VLASTNOSTI REKURZÍVNYCH JAZYKOV.....	17
8.7.3. POSTOV KOREŠPONDENČNÝ PROBLÉM.....	17
8.7.4. NEROZHODNUTEĽNÉ PROBLÉMY PRE KONTEXTOVÉ A BEZKONTEXTOVÉ JAZYKY.....	17

8. Formálne jazyky a automaty.

8.1. Úvod do problematiky.

8.1.1. Základné pojmy.

Definícia.

Neprázdnu konečnú množinu symbolov Σ nazývame *abeceda*. Konečná postupnosť symbolov z Σ je *slovo* nad abecedou Σ . Množina slov nad abecedou Σ je *jazyk* nad abecedou Σ . prázdne slovo označujeme symbolom ε . Mohutnosť množiny, vytvárajúcej slovo, sa nazýva *dĺžka slova*.

8.1.2. Elementárne operácie nad slovami.

Definícia.

Zreťazenie slov $v = v_1..v_n$ a $w = w_1..w_m$ je slovo $vw = v_1..v_nw_1..w_m$. *Zrkadlový obraz* slova $v = v_1..v_n$ je slovo $v^R = v_n..v_1$.

Označenie.

Množinu všetkých slov nad abecedou Σ označujeme Σ^* . Množinu všetkých neprázdnych slov nad abecedou Σ označujeme Σ^+ .

Definícia.

Nech $x, y \in \Sigma^*$. Slovo x je *podслово* slova y , ak $\exists v, w \in \Sigma^* : vxw = y$.

8.1.3. Elementárne operácie nad jazykmi.

Poznámka.

Nad jazykmi možno vykonávať elementárne množinové operácie, čiže zjednotenie, prienik, rozdiel a doplnok v Σ^* .

Definícia.

Zreťazenie jazykov L_1 a L_2 je jazyk $L_1L_2 = \{vw; v \in L_1 \text{ a } w \in L_2\}$. *n-tá mocnina* jazyka L je jazyk $L^n = \begin{cases} \{\varepsilon\} & \leftarrow n=0 \\ LL^{n-1} & \leftarrow n \geq 1 \end{cases}$. *Uzáver* jazyka L je jazyk $L^* = \bigcup_{n \in \mathbb{N}} L^n$. $L^* - \{\varepsilon\}$ je *kladný uzáver* jazyka L .

8.1.4. Gramatiky

Definícia.

Usporiadanú štvoricu $G = (N, T, P, \sigma)$ nazývame *gramatika*. N

($\quad \quad \quad$ d' $\quad \quad$ t') T
($\quad \quad \quad$) P

$(N \cup T)^* N (N \cup T)^* \times (N \cup T)^*$ *pravidiel* na úpravu neterminálnych symbolov a σ je štartovací symbol.

Označenie.

Ak pravidlo P α t' β , tak píšeme $\alpha \rightarrow \beta$.

Definícia.

Krok odvodenia gramatiky G je relácia \Rightarrow_G $\forall v, w \in (N \cup T)^* : v \Rightarrow_G w \Leftrightarrow$
 $\exists x, y, u, z \in (N \cup T)^* : v = xuy \wedge w = xzy \wedge u \rightarrow z \in P$ H \quad í \quad w je *odvoditeľné* z v ($v \Rightarrow_G^* w$),
 x t' v do w $(N \cup T)$ Γ

štartovacieho symbola gramatiky G , nazývame jazyk *generovaný* gramatikou G $L(G)$.

8.1.5. Procedurálna reprezentácia jazyka.

Definícia.

Nech F je program, ktorý na vstup $x \in \Sigma^*$ poskytuje odpovede áno/nie. F je *procedurálna reprezentácia* jazyka L , ak na $x \in \Sigma^*$ odpovie áno práve vtedy keď $x \in L$. Vo všeobecnosti F akceptuje L , ak $L \subseteq F$.

8.2. Regulárne gramatiky a konečné automaty.

8.2.1. Regulárne gramatiky.

Definícia.

Gramatika $G = (N, T, P, \sigma)$ je *regulárna* (sprava lineárna), ak sú všetky jej pravidlá tvaru $A \rightarrow xB$, kde $A \in N$, $B \in N \cup \epsilon$ a $x \in T^*$. Jazyk L je *regulárny*, ak je generovaný regulárnou gramatikou G .

8.2.2. Konečné automaty.

Definícia.

Usporiadanú päťicu $A = (K, \Sigma, \delta, q_0, F)$, kde K je množina stavov, Σ je abeceda, δ je prechodová funkcia $K \times \Sigma \rightarrow K$, $q_0 \in K$ a $F \subseteq K$, nazývame *konečný automat*. Usporiadanú dvojicu (q, w) , kde $q \in K$ a $w \in \Sigma^*$, nazývame *konfigurácia* automatu A , ak w je prípona vstupného slova v .

Krok odvodenia $(q, v) \xrightarrow[A]{f} (p, w)$ znamená, že $\exists a \in \Sigma : v = aw$ a $\delta(q, a) = p$. *Konečný výpočet* je postupnosť konfigurácií $(q_0, v) \xrightarrow[A]^* (q, \epsilon)$. Ak výsledný stav $q \in F$, nazývame v *akceptované slovo*. Jazyk $L(A) = \{v \in \Sigma^* : \sigma \xrightarrow[A]^* v\}$ nazývame jazyk *rozpoznávaný* automatom A .

8.2.3. Nedeterministické automaty.

Definícia.

Automat A sa nazýva *deterministický*, ak $\forall q \in K \forall a \in \Sigma \exists ! p \in K : \delta(q, a) = p$. V prípade, že existuje $a \in \Sigma$ a $q \in K$ tak, že $\delta(q, a) = \emptyset$, nazývame A *nedeterministický*.

Veta 8.2.3.1.

Nech L je jazyk. Ak L je regulárny, potom existuje konečný automat A' , ktorý rozpoznáva L .

Dôkaz.

Nech $A = (K, \Sigma, \delta, q_0, F)$. Potom $K' = \mathbf{P}(K)$ (množina podmnožín $M \subseteq K$). $q_0' = \{q_0\}$, $F' = \{M \subseteq K : \exists q \in M : q \in F\}$ a $\delta' : \mathbf{P}(K) \times \Sigma \rightarrow \mathbf{P}(K)$ je daná predpisom $\delta'(M, a) = \bigcup_{q \in M} \delta(q, a)$.

tento automat akceptuje rovnaký jazyk ako pôvodný.

Lema 8.2.3.1.

Nech L je jazyk. Ak L je regulárny, potom existuje konečný automat A' bez ϵ -prechodov, ktorý rozpoznáva L .

Dôkaz.

Vo dvoch krokoch. Najprv odstránime ϵ -prechodov, ktoré sú v $\delta'(q, a) = \{p \in K : (q, a) \xrightarrow[A]^* (p, \epsilon)\}$.

8.2.3.1 k nemu existuje deterministický automat A' .

L . Z konštrukcie tohoto automatu, ktorá odstráni ϵ -prechodov.

Veta 8.2.3.2.

Nech L je jazyk. Ak L je regulárny, potom L je regulárny jazyk.

Dôkaz.

Pre $A = (K, \Sigma, \delta, q_0, F)$ zostrojíme regulárnu gramatiku $G = (N, T, P, \sigma)$, kde $N = K, T = \Sigma, P = \{q \rightarrow ap; q, p \in N, a \in \Sigma, \delta(q, a) = p\} \cup \{q \rightarrow \varepsilon; q \in F\}$ a $\sigma = q_0$

$$L(G) = L(A).$$

Lema 8.2.3.2.

Nech L je regulárny jazyk. Potom existuje regulárna gramatika G s pravidlami tvaru $X \rightarrow aY, X \rightarrow Y, X \rightarrow \varepsilon$, ktorá generuje L .

Dôkaz.

Jednoduché.

Veta 8.2.3.3.

Nech L je regulárny jazyk. Potom existuje regulárna gramatika A taká, že $L = L(A)$.

Dôkaz.

Nech $G = (N, T, P, \sigma)$ je „jednokroková“ regulárna gramatika, generujúca L . Zostrojíme $A = (K, \Sigma, \delta, q_0, F)$, kde $K = N, \Sigma = T, q_0 = \sigma, F = \{X \in N; X \rightarrow \varepsilon \in P\}$ a $\delta(X, a) = \{Y \in N; X \rightarrow aY \in P\}$ a $\delta(X, \varepsilon) = \{Y \in N; X \rightarrow Y \in P\}$. Gramatika A vygeneruje terminálne slovo.

Dôsledok 8.2.3.1.

Dôkaz.

Tvrdenie vyplýva priamo z predchádzajúcich viet.

8.24. Myhill-Nerodova veta.

Definícia.

Nech R je relácia ekvivalencie na množine Σ^* . Triedu ekvivalencie, do ktorej patrí prvok w nazývame *index* R prvku w a označujeme ju $[w]$.

Poznámka.

Pre $L \subseteq \Sigma^*$ definujeme reláciu ekvivalencie \equiv_L na množine Σ^* tak, že $v \equiv_L w \Leftrightarrow \forall x \in \Sigma^* : vx \in L \Leftrightarrow wx \in L$. Pritom ak $\alpha_1.. \alpha_n..$ je lexikografické usporiadanie Σ , potom $T_2[\alpha_1.. \alpha_n..]$ predpisom $T_2[\alpha_i, \alpha_j] = 1 \Leftrightarrow \alpha_i \alpha_j \in L$, potom $v \equiv_L w \Leftrightarrow$ prvky v a w majú rovnaké rady, prislúchajúce v a w sú rovnaké.

Definícia.

Nech $A = (K, \Sigma, \delta, q_0, F)$ je regulárna gramatika. Ak δ je definovaná $\forall q \in K \forall a \in \Sigma, \delta(q, a)$ je úplne definovaná.

Označenie.

Pre regulárnu gramatiku A o najviac n stavoch s úplne definovanou prechodovou funkciou, akceptujúci L definujeme $DKA_{\Sigma, n} = \{L \in R; \exists A \text{ o najviac } n \text{ stavoch s úplne definovanou prechodovou funkciou, akceptujúci } L\}$.

Veta 8.2.4.1 (Myhill-Nerode).

- Jazyk L je regulárny $\Leftrightarrow \exists L \in DKA_{\Sigma, n}$
- $L \in DKA_{\Sigma, n} \Leftrightarrow$ relácia \equiv_L má index $\leq n$.

Dôkaz.

Nech L je regulárny jazyk. Potom $\exists w_1..w_{|K|+1} \forall i \neq j \in 1..|K|+1 : \neg w_i \equiv_L w_j$. Ak A má iba $|K|$ stavov, potom $\exists i \neq j \in 1..|K|+1 : w_i \equiv_L w_j$.

Obrátene nech index $\equiv_L = n$. Definujme $A' = (K', \Sigma, \delta', q_0', F')$, kde $K' = \{[w]; w \in \Sigma^*\}, q_0' = [\varepsilon], F' = \{[w]; w \in L\}$ a $\delta'([w], a) = [wa]$. A' má zjavne n stavov.

do L (práve vtedy patrí trieda, ktorú reprezentuje, do F') $T \quad L \in DKA_{\Sigma, n}$.

Označenie.

$$mindet_{\Sigma}(L) = \min\{n; L \in DKA_{\Sigma, n}\}.$$

Dôsledok 8.2.4.1.

$$mindet_{\Sigma}(L) = \text{index relácie } \equiv_L.$$

Dôkaz.

Tvrdenie vyplýva priamo z Myhill-Nerodovej vety (veta 8.2.4.1).

8.2.5. Aplikácie Myhill-Nerodovej vety.

Poznámka.

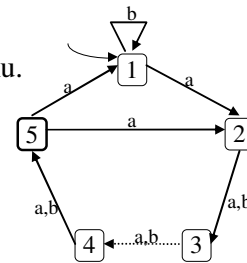
Myhill-indexu relácie \equiv_L nedeterministickému, ako ukazuje nasledujúca veta.

Veta 8.2.5.1.

$$\forall n \geq 1 \exists L \subseteq \{a,b\}^* : mindet_{\{a,b\}^*}(L) = 2^n.$$

Dôkaz.

P $L(A)$, kde automat A je zobrazený na obrázku.



8.2.6. Minimalizácia deterministického konečného automatu.

Definícia.

D $A_1 = (K_1, \Sigma, \delta_1, q_1, F_1)$ a $A_2 = (K_2, \Sigma, \delta_2, q_2, F_2)$ sú **izomorfné**, ak existuje bijekcia $h : K_1 \rightarrow K_2$ $h(q_1) = q_2, F_2 = \{h(q); q \in F_1\}$ a $\forall q \in K_1 \forall a \in \Sigma : \delta_2(h(q), a) = h(\delta_1(q), a)$.

Veta 8.2.6.1.

Jediné minimálne deterministické automaty, definované v dôkaze Myhill-Nerodovej vety, L sú automaty izomorfné s

Dôkaz.

Nedokazujeme.

8.2.7. Regulárne výrazy.

Definícia.

Nech Σ \oplus, \otimes a $*$ $RV(\Sigma)$ **regulárnych výrazov** na Σ definujeme rekurzívne

1. $\emptyset \in RV(\Sigma), \varepsilon \in RV(\Sigma)$
2. $\forall a \in \Sigma : a \in RV(\Sigma)$
3. $\forall \alpha, \beta \in RV(\Sigma) : \alpha \oplus \beta \in RV(\Sigma), \alpha \otimes \beta \in RV(\Sigma), \alpha^* \in RV(\Sigma)$.

Definícia.

Jazyk $L[\alpha]$, reprezentovaný regulárnym výrazom α definujeme nasledovne :

1. $[\emptyset] = \emptyset, [\varepsilon] = \{\varepsilon\}$
2. $\forall a \in \Sigma : [a] = \{a\}$
3. $\forall \alpha, \beta \in RV(\Sigma) : L[\alpha] = L_{\alpha} \wedge L[\beta] = L_{\beta} \Rightarrow L[\alpha \oplus \beta] = L_{\alpha} \cup L_{\beta} \wedge L[\alpha \otimes \beta] = L_{\alpha} L_{\beta} \wedge L[\alpha^*] = L_{\alpha}^*$.

Označenie.

H i $t' P$ O $Min(\Sigma)$
 $t' P$ a $T = \{[\alpha]; \alpha \in RV(\Sigma)\}$.

Veta 8.2.7.1.

$Min(\Sigma) = T$.

Dôkaz.

Tvrdenie vyplýva z definície regulárnych výrazov.

- a) $t' P$ T
- b) T $t' P$.

8.2.8. Uzáverové vlastnosti.

Veta 8.2.8.1.

R je uzavretá na zjednotenie a prienik.

Dôkaz.

Nech L_1 a L_2 sú regulárne jazyky; $L_1 = (K_1, \Sigma, \delta_1, q_1, F_1)$, $L_2 = (K_2, \Sigma, \delta_2, q_2, F_2)$. definujme automat $A = (K_1 \times K_2, \Sigma, \delta, (q_1, q_2), F)$, kde $(p, q) \in F \Leftrightarrow p \in F_1 \vee q \in F_2$ a $\delta((p, q), a) = (\delta_1(p, a), \delta_2(q, a))$. Tento automat vykonáva paralelne L_1 a L_2 ,
 $t' L(A) = L_1 \cup L_2$.

O A predpisom $(p, q) \in F \Leftrightarrow p \in F_1 \wedge q \in F_2$ dostaneme automat, akceptujúci $L_1 \cap L_2$.

Veta 8.2.8.2.

R je uzavretá na doplnok.

Dôkaz.

Nech L je regulárny jazyk, akceptovaný automatom $A = (K, \Sigma, \delta, q_0, F)$. Zámenou F a $K-F$ dostávame automat pre doplnok.

Veta 8.2.8.3.

R t'

Dôkaz.

Nech L_1 a L_2 sú regulárne jazyky; $L_1 = (K_1, \Sigma, \delta_1, q_1, F_1)$, $L_2 = (K_2, \Sigma, \delta_2, q_2, F_2)$. definujme automat $A = (K_1 \cup K_2, \Sigma, \delta, q_1, F_2)$, kde $\delta = \delta_1 \cup \delta_2$ plus ε -
 $q \in F_1$ do q_2 . Automat A t' $L_1 L_2$.

Veta 8.2.8.4.

R je uzavretá na uzáver.

Dôkaz.

K automatu pre L pridáme ε -prechody z F do q_0 .

Veta 8.2.8.5 (Kleene).

$R = Min(\Sigma)$.

Dôkaz.

R $t' P$ (z predchádzajúcich viet) $\Rightarrow Min(\Sigma) \subseteq R$. Obrátene nech L je regulárny jazyk, akceptovaný automatom $A = (K, \Sigma, \delta, q_0, F)$ D f $R_{i,j} = \{w \in \Sigma^*; \delta(q_i, w) \rightarrow q_j\}$. Zrejme $L = \bigcup_{i,j=1}^{|K|} R_{i,j}$ O $R_{i,j}^k = \{w \in R_{i,j}; q_1..q_k\}$. $\forall i, j \in 1..|K| : R_{i,j}^0 \subseteq \Sigma \cup \{\varepsilon\}$ a to $t' g$ A $\forall i, j \in 1..|K| \forall k \geq 1 : R_{i,j}^k = R_{i,j}^{k-1} \cup R_{i,k}^{k-1} \cdot (R_{k,k}^{k-1})^* \cdot R_{k,j}^{k-1}$ a d' RV t' $R \subseteq Min(\Sigma)$.

8.3. Bezkontextové gramatiky a zásobníkové automaty.

8.3.1. Bezkontextové gramatiky.

Definícia.

Gramatika $G = (N, T, P, \sigma)$ je *bezkontextová*, ak sú všetky jej pravidlá tvaru $A \rightarrow x$, kde $A \in N$, a $x \in (N \cup T)^*$. Jazyk L je *bezkontextový*, ak je generovaný bezkontextovou gramatikou.

Definícia.

Bezkontextová gramatika G je *nevypúšťajúca*, ak neobsahuje pravidlá tvaru $X \rightarrow \varepsilon$.

Veta 8.3.1.1.

Nech L je bezkontextový jazyk, generovaný gramatikou G a neobsahujúci ε . Potom existuje G' , generujúca L .

Dôkaz.

Ak $\varepsilon \notin L$ $\sigma = \alpha_0 \xRightarrow{G} \alpha_1 \xRightarrow{G^*} \alpha_n = w \in T^* \exists i \in 1..n : |\sigma| \leq |\alpha_i|$. Ak v G zameníme všetky pravidlá $\alpha_k \xRightarrow{G} \alpha_{k+1}$, kde $|\alpha_k| > |\alpha_{k+1}|$ pravidlom $\alpha_k \xRightarrow{G} \alpha_{k+j}$, kde $j = \min\{i \in k..n; |\alpha_k| \leq |\alpha_{k+i}|\}$, dostávame

Veta 8.3.1.2.

Nech L je bezkontextový jazyk, generovaný gramatikou G . Potom existuje bezkontextová gramatika G' bez pravidiel tvaru $X \Rightarrow Y$, generujúca L .

Dôkaz.

Triviálne.

8.3.2. Redukované gramatiky.

Definícia.

Bezkontextová gramatika G je *redukovaná*, ak neobsahuje pravidlá tvaru $A \rightarrow \varepsilon$ a pravidlá tvaru $A \rightarrow BC$, kde $A \in N$ a $B, C \in N$.

Veta 8.3.2.1.

Pre každý bezkontextový jazyk L existuje redukovaná gramatika, ktorá ho generuje.

Dôkaz.

Triviálne.

8.3.3. Najľavejšie odvodenie.

Definícia.

Odvodenie v bezkontextovej gramatike G je *najľavejšie*, ak je prvým pravidlom odvodenia pravidlo tvaru $A \rightarrow x$, kde A je prvým znakom v w .

Veta 8.3.3.1.

Nech L je bezkontextový jazyk generovaný gramatikou G a nech $w \in L$. Potom v G existuje najľavejšie odvodenie w .

Dôkaz.

Triviálne.

8.3.4. Chomského a Greibachovej normálny tvar.

Definícia.

Bezkontextová gramatika G je v *chomského normálnom tvare*, ak všetky jej pravidlá sú tvaru $A \Rightarrow a$ a $A \Rightarrow BC$, kde $a \in T$ a $B, C \in N$.

Veta 8.3.4.1.

Nech L je bezkontextový jazyk bez ε , generovaný gramatikou G . Potom existuje bezkontextová gramatika G' v chomského normálnom tvare, generujúca L .

Dôkaz.

Pravidlo tvaru $A \Rightarrow Xx$, kde $x \in (N \cup T)$ a $X \in (N \cup T)^*$, nahradíme pravidlom $A \Rightarrow YZ$, kde y a Z sú nové netermy. Ďalej nahradíme pravidlo $Z \Rightarrow x$ a pravidlo $Y \Rightarrow X$ istým spôsobom. Výsledkom je gramatika v chomského normálnom tvare.

Definícia.

Bezkontextová gramatika G je v *greibachovej normálnom tvare*, ak všetky jej pravidlá sú tvaru $A \Rightarrow a\alpha$, kde $a \in T$ a $\alpha \in N^*$.

Veta 8.3.4.1.

Nech L je bezkontextový jazyk, generovaný gramatikou G . Potom existuje bezkontextová gramatika G' v greibachovej normálnom tvare, generujúca L .

Dôkaz.

Gramatiku v chomského normálnom tvare upravíme na greibachovej normálny tvar ABC nahradíme pravidlom $A \Rightarrow aB_n \dots B_1 C$, kde $aB_n \dots B_1 C$ prerušené v okamihu $B_{n+1} \Rightarrow a$.

8.3.5. Stromy odvodenia.**Definícia.**

Nech G je bezkontextová gramatika. Každý strom s vrcholmi z $(N \cup T)^*$ nazývame *strom odvodenia* v G , ak jeho koreň σ a β je synom α a $\alpha \xRightarrow{G} \beta$. Listov stromu odvodenia T nazývame *výsledok* stromu odvodenia T a označujeme ho w_T .

Veta 8.3.5.1.

Nech G je bezkontextová gramatika. $\forall w \in (N \cup T)^* : w \in L(G) \Leftrightarrow \exists T : w_T = w$.

Dôkaz.

Triviálne.

8.3.6. Pumpovacia lema.**Veta 8.3.6.1 (Pumpovacia).**

Nech L je bezkontextový jazyk. Potom $\exists p, q \in \mathbb{N} \forall z \in L : |z| > p \Rightarrow \exists u, v, w, x, y \in \Sigma^* : z = uvwxy \wedge |vwx| \leq q \wedge vx \neq \epsilon \wedge \forall i \geq 0 : uv^iwx^iy \in L$.

Dôkaz.

Nech G je bezkontextová gramatika v Greibachovej normálnom tvare, generujúca $L - \{\epsilon\}$. Nech $|M| = k$. Nech P je cesta v T v G , ktorého výsledok t je dlhší ako p , obsahuje vetvu dlhšiu ako k . Nech P je najdlhšia cesta v T . Potom v G existuje neterminál, ktorý sa na P nachádza. Nech A je neterminál. Potom $A \xRightarrow{G} vAx$ (predposledný výskyt) a $A \xRightarrow{G} w$ (posledný výskyt), kde $v, w, x \in T^*$. Nech u je úsek „napravo“ od P a y výsledok „napravo“ od P . Potom $t = uvwxy$ a $|vwx| \leq 2^k$, $|u| \leq k+1$, $vx \neq \epsilon$ a úsek $A \xRightarrow{G} vAx$ sa opakuje i krát. Potom $uv^iwx^iy \in L$.

Definícia.

Nech G je bezkontextová gramatika, $A \in N$. Pravidlo tvaru $A \Rightarrow \alpha$, kde $\alpha \in (N \cup T)^*$, nazývame *A-pravidlo*.

Lema 8.3.6.1.

Nech G je bezkontextová gramatika bez ϵ -pravidiel, nech $A \Rightarrow \alpha\beta\gamma$ je jej A -pravidlo a nech $\{\beta \Rightarrow \beta_i\}_{i=1..n}$ sú všetky β -pravidlá. Potom nahradením pravidla $A \Rightarrow \alpha\beta\gamma$ pravidlami $\{\beta \Rightarrow \alpha\beta_i\gamma\}_{i=1..n}$ generujem rovnaký jazyk.

Dôkaz.

Zrejme.

Lema 8.3.6.2.

Nech G je bezkontextová gramatika bez ϵ -pravidiel, nech $\{A \Rightarrow A\alpha_i\}_{i=1..n}$ sú všetky A -pravidlá s A na ľavom boku a $\{A \Rightarrow \beta_i\}_{i=1..m}$ sú všetky ostatné A -pravidlá. Potom nahradením A -pravidiel pravidlami $\{A \Rightarrow \beta_i, A \Rightarrow \beta_i\beta, \beta \Rightarrow \alpha_i, \beta \Rightarrow \alpha_i\beta\}_{i=1..n.m}$ generujem rovnaký jazyk.

Dôkaz.

Zrejme.

8.3.7. Zásobníkové automaty.**Definícia.**

Usporiadanú sedmiticu $A = (K, \Sigma, T, \delta, q_0, Z_0, F)$ nazývame *zásobníkový automat*. K, Σ, δ, q_0 a F sú štandardne definované, T je zásobník a Z_0 je začiatok zásobníka.

Definícia.

Konfigurácia zásobníkového automatu je usporiadaná trojica (q, w, γ) , kde $q \in K$, w je slovo nad Σ a γ je stav zásobníka.

Poznámka.

Zásobníkový automat je deterministický práve vtedy, keď pre každú konfiguráciu (q, w, γ) existuje najviac jedna konfigurácia (q', w', γ') taká, že $(q, w, \gamma) \Rightarrow (q', w', \gamma')$.

Veta 8.3.7.1.

Rozoznávanie koncovým stavom a prázdny zásobníkom sú rovnako silné.

Dôkaz.

Triviálne.

Veta 8.3.7.2.

Pre každý bezkontextový jazyk L existuje zásobníkový automat A , ktorý ho akceptuje.

Dôkaz.

Nech $G = (N, T_G, P, \sigma)$ je bezkontextová gramatika v greibachovej normálnom tvare, generujúca L . Definujme zásobníkový automat $A = (K, \Sigma, T, \delta, q, Z_0, F)$, kde $K = \{q\}$, $\Sigma = N \cup T_G$, $T = N$, $Z_0 = \sigma$, $F = \emptyset$ a $\delta(q, aw, A\gamma) = (q, w, B_1..B_n\gamma) \Leftrightarrow A \xrightarrow{G} aB_1..B_n \in P$. Ak automat A akceptuje vstupné slová prázdny zásobníkom, tak $L \subseteq L(A)$.

Naopak, ak $L \subseteq L(A)$, potom L je prázdny zásobníkom akceptovaný. Pretože A je deterministický, tak $L(A) \subseteq L$ a teda zjavne $L(A) \subseteq L$. Pretože A je deterministický, tak $L(A) \subseteq L$ a teda zjavne $L(A) \subseteq L$.

Veta 8.3.7.3.

Pre každý bezkontextový jazyk L existuje zásobníkový automat A , ktorý ho akceptuje.

Dôkaz.

Nech $A = (K, \Sigma, T, \delta, q_0, Z_0, F)$ je zásobníkový automat, akceptujúci vstupné slová prázdny zásobníkom. Definujme gramatiku $G = (N, T_G, P, \sigma)$, kde $N = \{[q, B, p]; p, q \in K, B \in T\} \cup \{\sigma\}$, $T_G = \Sigma$ a $P = \{\sigma \Rightarrow [q_0, Z_0, q]; q \in K\} \cup \{[q, B, q_{n+1}] \Rightarrow a[q_1, B_1, q_2]..[q_n, B_n, q_{n+1}]; (q_1, w, B_1..B_n\gamma) \in \delta(q, aw, B\gamma)\}$. Táto gramatika sleduje $L(A)$ a teda $L(G) = L(A)$.

8.3.8. Uzáverové vlastnosti.**Veta 8.3.8.1.**

L_{CF} nie je uzavretá na prienik a doplnok.

Dôkaz.

Ukážeme, že L_{CF} nie je uzavretá na prienik. Navyše $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}} \Rightarrow$ ak by L_{CF} bolo uzavretá na prienik, tak by L_{CF} bolo uzavretá na doplnok.

Veta 8.3.8.2.

L_{CF} je uzavretá na prienik s R .

Dôkaz.

Nech $A = (K, \Sigma, T, \delta, q_0, Z_0, F)$ je zásobníkový automat a $A' = (K', \Sigma, \delta', q_0', F')$

Potom zásobníkový automat $A'' = (K \times K', \Sigma, T, \delta'', (q_0, q_0', Z_0, F \times F'))$, kde $\delta''((p, q), aw, \gamma) = \{(p', q'), w, \gamma\}$; $\delta(p, a) = p' \wedge (q', w, \gamma) \in \delta'(q, aw, \gamma)$ rozpoznáva jazyk $L(A) \cap L(A')$.

8.4. Frázové gramatiky a Turingove stroje.**8.4.1. Turingove stroje.****Definícia.**

Usporiadanú šesticu $A = (K, \Sigma, T, \delta, q_0, F)$ nazývame *Turingov stroj*. K stavov, Σ je vstupná abeceda T je pásková abeceda (a teda $\Sigma \subseteq T$), $\delta : K \times T \rightarrow K \times T - \{B\} \times \pm 1$ je prechodová funkcia, q_0 F

Definícia.

Konfigurácia Turingovho stroja je usporiadaná trojica (q, w, i) , kde $q \in K$, $w \in T^*$ a $i \in N$ je pozícia hlavy na páske.

Poznámka.

P T g

8.4.2. Modifikované Turingove stroje.**Veta 8.4.2.1.**

Viacstopové Turingove stroje sú ekvivalentné štandardným.

Dôkaz.

Na pásku sa nezapisujú symboly, ale vektory symbolov.

Veta 8.4.2.2.

T g

Dôkaz.

P í

Veta 8.4.2.3.

Viacpáskové Turingove stroje sú ekvivalentné štandardným.

Dôkaz.

k pásov = $2k \hat{\delta}$

Definícia.

Stupeň nedeterministickosti Turingovho stroja je $k = \max\{|\delta(q, w, i)|; q \in K, w \in T^*, i \in N\}$.

Veta 8.4.2.4.

Nedeterministické Turingove stroje sú ekvivalentné deterministickým.

Dôkaz.

Nech k ň T g T. Deterministický T'
t' l' í í

8.4.3. Rekurzívne vypočítateľné jazyky.**Definícia.**

Jazyk, rozpoznávaný Turingovým strojom, nazývame *rekurzívne vypočítateľný*. Ak $L \in RE$, L je *rekurzívny*.

8.4.4. Frázové gramatiky.**Definícia.**

Gramatika $G = (N, T, P, \sigma)$ je *frázová*
x f

Veta 8.4.4.1.

$f \quad g \quad g \quad í \quad í \quad í'$

Dôkaz.

Dvojpáskový nedeterministický TS $g \quad t'$
 g

Veta 8.4.4.2.

$P \quad T \quad g \quad A$ existuje gramatika $G \quad L(A) = L(G)$.

Dôkaz.

Gramatika postupuje v troch krokoch. Najprv vygeneruje

$í \quad P \quad g$
 $t' \quad T \quad vqaw \rightarrow vbqw$, ktoré
 simulujú posun hlavy automatu v stave $q \quad í \quad í \quad a \quad P \quad í'$
 $t' \quad v \quad t' \quad g \quad ä \quad í \quad T \quad t'$
 $ó \quad í \quad p$ je akceptovací stav. Aby sme tera $t' \quad g \quad í \quad ,, \quad "$
 $ó \quad t' \quad P \quad ,, \quad "$

Definícia.

Frázová gramatika je v *normálnom tvare*, ak má všetky pravidlá tvaru $\alpha \rightarrow \beta$ a $A \rightarrow a$, kde $\alpha, \beta \in N^*$, $A \in N$ a $a \in T$.

Lema 8.4.4.1.

P ý rekurzívny jazyk existuje frázová gramatika v normálnom tvare, ktorá ho generuje.

Dôkaz.

Triviálne.

8.4.5. Churchova téza.

Hypotéza.

$g \quad í' \quad T \quad g$

8.5. Kontextové gramatiky a lineárne ohraničené automaty.

8.5.1. Kontextové gramatiky.

Definícia.

Gramatika $G = (N, T, P, \sigma)$ je *kontextová*, ak pre všetky jej pravidlá $x \Rightarrow y$, kde $x, y \in (N \cup T)^*$ platí $|x| \leq |y|$. Jazyk L je *kontextový*, ak je generovaný kontextovou gramatikou.
 $T \quad x \quad L_{CS}$.

Definícia.

Kontextová gramatika G je v *normálnom stave*, ak má všetky pravidlá tvaru $\alpha \rightarrow \beta$ a $A \rightarrow a$, kde $\alpha, \beta \in N^*$, $|\alpha| \leq |\beta|$, $A \in N$ a $a \in T$.

Lema 8.5.1.1.

$P \quad x \quad L$ existuje kontextová gramatika G v normálnom tvare, ktorá ho generuje.

Dôkaz.

Triviálne.

8.5.2. Lineárne ohraničené automaty.

Definícia.

Turingov stroj, pracujúci v priestore, vyhradenom vstupom (alebo iným spôsobom fixne) *lineárne ohraničený* automat.

Veta 8.5.2.1.

$P \quad A$ existuje kontextová gramatika $G \quad L(A) = L(G)$.

Lema 8.5.3.6.

O T g t' g x g
 touto terminálnou abecedou.

Dôkaz.

$ó$ f $í$

Definícia.

Kanonická t' g $í$ T t'
 gramatik nad terminálnou abecedou T $ó$ t'
 binárnou abecedou.

Lema 8.5.3.7.

Ex g g t' x g $í$
 abecedou T .

Dôkaz.

Algoritmus generuje kanonickú postup t'
 kóduje kontextovú gramatiku, a ak áno, zaradí ju do postupnosti.

Definícia.

Nech $\{G_i\}_{i \in \mathbb{N}}$ t' g $í$ T . Jazyk
 $L_{DIAG} = \{w_i; w_i \notin G_i\}$ nazývame *diagonálny* jazyk nad abecedou T .

Lema 8.5.3.8.

$\exists L \in L_{REC} \setminus L_{ECS}$.

Dôkaz.

L_{DIAG} nad kontextovými gramatikami zjavne nie je kontextový. Ale pre dané $w \in T$ $í$ t'
 index v v kanonickej postupnosti (nech je to i) g t' $ó$ i -tej gramatiky G_i postupnosť
 kontextových gramatik nad terminálnou abecedou T t' $í$ $t'w$ do $L(G_i)$ a tým aj do $L \Rightarrow$
 L je rekurzívny.

Lema 8.5.3.9.

$L_{REC} \subseteq L_{RE}$.

Dôkaz.

Triviálne.

Lema 8.5.3.10.

$\exists L \in L_{RE} \setminus L_{REC}$.

Dôkaz.

\forall $í$ T g f'

Dôsledok 8.5.3.1 (Chomského hierarchia).

$R \subseteq L_{CF} \subseteq L_{ECS} \subseteq L_{REC} \subseteq L_{RE}$.

Dôkaz.

Tvrdenie je priamym dôsledkom predchádzajúcich liem.

8.5.4. Uzáverové vlastnosti.

Veta 8.5.4.1.

L_{CS}, L_{REC}, L_{RE} t' $(L_{CS}$ iba na kladný uzáver) a zrkadlový
 obraz.

Dôkaz.

- Zjednotenie : Nech $G_1 = (N_1, T_1, P_1, S_1)$ a $G_2 = (N_2, T_2, P_2, S_2)$ sú kontextové (rekurzívne, frázové) gramatiky a nech $N_1 \cap N_2 = T_1 \cap T_2 = \emptyset$. Definujme gramatiku $G = (N, T, P, S)$, kde $S \notin N_1 \cup N_2 \cup T_1 \cup T_2$, $N = N_1 \cup N_2 \cup S$, $T = T_1 \cup T_2$ a $P = P_1 \cup P_2 \cup \{S \rightarrow S_1 S_2\}$. G patrí do rovnakej triedy chomského hierarchie $L(G_1) \cup L(G_2)$.

- **Z t' :** $G_1 = (N_1, T_1, P_1, S_1)$ a $G_2 = (N_2, T_2, P_2, S_2)$ sú kontextové (rekurzívne, frázové) gramatiky a nech $N_1 \cap N_2 = T_1 \cap T_2 = \emptyset$. Definujme gramatiku $G = (N, T, P, S)$, kde $S \notin N_1 \cup N_2 \cup T_1 \cup T_2$, $N = N_1 \cup N_2 \cup S$, $T = T_1 \cup T_2$ a $P = P_1 \cup P_2 \cup \{S \rightarrow S_1 S_2\}$. G patrí do rovnakej triedy chomského hierarchie \hat{o} g l' t' g t' $L(G_1)$ a $L(G_2)$.
- **Uzáver :** Nech $G = (N, T, P, S)$ je kontextová (rekurzívna, frázová) gramatika. Pridaním pravidla $S \rightarrow SS$ vytvoríme gramatiku, prislúchajúcu do rovnakej triedy chomského hierarchie ako pôvodná gramatika, ak generujúcu uzáver $L(G)$.
- **Zrkadlový obraz :** Nech $G = (N, T, P, S)$ je kontextová (rekurzívna, frázová) gramatika. Potom gramatika $G' = (N, T, P', S)$, kde $P' = \{\alpha^R \rightarrow \beta^R; \alpha \rightarrow \beta \in P\}$, patrí do rovnakej triedy chomského hierarchie ako pôvodná gramatika a generuje zrkadlový obraz $L(G)$.

Veta 8.5.4.2.

L_{CS} a L_{REC} sú uzavreté na komplement.

Dôkaz.

Szelepcsenyho veta dokazuje tvrdenie pre L_{CS} . Pre L_{REC} je tvrdenie triviálne.

Veta 8.5.4.3.

L_{CS} sú uzavreté na prienik.

Dôkaz.

Triviálne.

8.6. Aplikácie Turingových strojov.**8.6.1. Turingove stroje, počítajúce celočíselné funkcie.**

- $f(k_1, \dots, k_n)$ reprezentujeme vstupom $0^{k_1} 1 \Lambda 0^{k_2} 1 \Lambda \dots \Lambda 0^{k_n} 1$ f 0^m , ak $f(k_1, \dots, k_n) = m$.

8.6.2. Turingove stroje ako enumerátory.**Definícia.**

Enumerátor je Turingov stroj bez vstupu so špeciálnou výstupnou páskou, na ktorej sa t' V ú slová oddelené špeciálnym symbolom l' (#). Všetky slová, ktoré takýto enumerátor A vyprodukuje, tvoria $L(A)$.

Veta 8.6.2.1.

Jazyk L í í l' d' x A , ktorý ho generuje.

Dôkaz.

Nech L je akceptovaný $TS A'$. Ak ekvivalentný DTS (konštrukcia v dôkaze vety 8.4.2.4) upravíme na enumerátor A , produkujúci všetky slová akceptované A' , potom $L(A) = L(A')$. Obrátene TS enumerátora a porovnáva slová so vstupom.

8.7. Rozhodnuteľnosť.**8.7.1. Niektoré štandardné problémy.****Definícia.**

Problém **Find** – pre danú rekurzívnu gramatiku a vstup w nad jej terminálnou abecedou t' l' $w \in L(G)$ a NIE, ak $w \notin L(G)$.

Problém **Inclusion** – pre danú rekurzívnu gramatiku a vstup w nad jej terminálnou t' Á O/ IE l' $w \in / \notin L(G)$.

Veta 8.7.1.1.

Find l' \Leftrightarrow **Inclusion** l'

Dôkaz.

Triviálne.

Veta 8.7.1.2 (Post).

L í d' L í í l'

