23 Gramatiky, regulární gramatiky a bezkontextové gramatiky, bezkontextové jazyky. Zásobníkové automaty a jejich vztah k bezkontextovým jazykům. Vlastnosti bezkontextových gramatik, lemma o vkládání. (A4B01JAG)

23.1 Gramatiky

23.1.1 Hierarchie gramatik

23.1.1.1 Definice

Gramatika je uspořádaná čtveřice $G = (N, \Sigma, S, P)$, kde

- N je konečná množina tzv. *neterminálů;*
- Σ je konečná neprázdná množina tzv. *terminálů*, platí $N \cap \Sigma = \emptyset$;
- $S \in N$ je startovací symbol;
- P je konečná množina pravidel typu $\alpha \to \beta$, kde α a β jsou slova nad $N \cup \Sigma$ taková, že α obsahuje alespoň jeden neterminál.

23.1.1.2 Příklad

V programovacích jazycích se často vyskytují definice typu číslo v Backus-Naurově formě:

- $\langle$číslo$\rangle$ ::= $\langle$číso bez zn.$\rangle$ | + $\langle$číslo bez zn.$\rangle$ | − $\langle$číslo bez zn.$\rangle$
- $\langle$číslo bez zn.$\rangle$ ::= $\langle$číslice$\rangle$ | $\langle$číslice$\rangle$ $\langle$číslo bez zn.$\rangle$
- $\langle$číslice$\rangle$ ::= 0|1|2|3|4|5|6|7|8|9

Jedná se o speciální příklad gramatiky: Označíme $S = \langle$číslo$\rangle$, $A = \langle$číslo bez zn.$\rangle$ a $B = \langle$číslice$\rangle$. Pak se jedná o gramatiku, kde

$$N = \{S, A, B\}\,, \quad \Sigma = \{+, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

a pravidla P jsou

- $S \to A$, $S \to +A$, $S \to -A$;
- $A \to B$, $A \to BA$;
- $B \to 0$, $B \to 1, \ldots$, $B \to 9$.

23.1.1.3 Přímé odvození

Je dána gramatika $G = (N, \Sigma, S, P)$. Řekneme, že δ se *přímo odvodí* z γ, značíme $\gamma \Rightarrow_G \delta$, jestliže existuje v P pravidlo $\alpha \to \beta$ a slova $\varphi, \psi \in (N \cup \Sigma^\star)$ taková, že $\gamma = \varphi\alpha\psi$ a $\delta = \varphi\beta\psi$.

Zhruba řečeno, ve slově γ najdeme některý výskyt podslova α, které tvoří levou stranu pravidla $\alpha \to \beta$ z P . Slovo δ dostaneme tak, že zvolený výskyt α (v γ) nahradíme slovem β (tj. pravou stranou pravidla).

23.1.1.4 Odvození

Je dána gramatika $G = (N, \Sigma, S, P)$. Řekneme, že δ se *odvodí* z γ, jestliže

- buď $\gamma = \delta$,
- nebo existuje posloupnost přímých odvození
- $$\gamma \Rightarrow \gamma_1 \Rightarrow_G \gamma_2 \Rightarrow_G \ldots \Rightarrow_G \gamma_k = \delta.$$
- Tento fakt značíme $\gamma \Rightarrow_G^\star \delta$.

23.1.1.5 Jazyk generovaný gramatikou

Řekneme, že slovo $w \in \Sigma^\star$ je *generováno* gramatikou G, jestliže $S \Rightarrow_G^\star w$.

Jazyk $L(G)$ *generovaný* gramatikou G se skládá ze všech slov generovaných gramatikou G, tj.

$$L(G) = \{w \in \Sigma^\star | S \Rightarrow_G^\star w\}\,.$$

23.1.1.6 Konvence

- Neterminály značíme obvykle velkými písmeny $A, B, X, Y, \ldots$
- Terminály značíme obvykle malými písmeny ze začátku abecedy $a, b, c, d, \ldots$
- Slova z $(N \cup \Sigma)^\star$ obvykle značíme řeckými písmeny $\alpha, \beta, \ldots$
- Terminální slova, tj. slova z $\Sigma^\star$, značíme malými písmeny z konce abecedy $u, w, x, y, \ldots$

Obvykle v textu vynecháváme jméno gramatiky, je-li z kontextu jasné o jakou gramatiku se jedná. Píšeme proto $\Rightarrow$ a $\Rightarrow^\star$ místo $\Rightarrow_G$ a $\Rightarrow_G^\star$.

23.1.1.7 Chomského hierarchie

Podle podmínek, které klademe na pravidla dané gramatiky rozlišujeme gramatiky a jimi generované jazyky na:

- *Gramatiky typu 0* jsou gramatiky tak, jak jsme je zavedli v odstavci 23.1.1.1. Jazyky generované gramatikami typu 0 se nazývají *jazyky typu 0.*
- *Gramatiky typu 1* též *kontextové gramatiky* jsou takové gramatiky, kde každé pravidlo v P je tvaru

  $$\alpha A \beta \rightarrow \alpha \gamma \beta,$$

  kde $\alpha, \beta, \gamma \in (N \cup \Sigma)^\star$, A je neterminál a $\gamma \neq \epsilon$. Jedinou výjimku tvoří pravidlo $S \rightarrow \epsilon$, pak se S nevyskytuje na pravé straně žádného pravidla.

  Jazyky generované gramatikami typu 1 se nazývají *jazyky typu 1*, též *kontextové jazyky.*
- *Gramatiky typu 2* též *bezkontextové gramatiky* (což zkracujeme na CFG) jsou takové gramatiky, kde každé pravidlo v P je tvaru

  $$A \rightarrow \gamma,$$

  kde $\gamma \in (N \cup \Sigma)^\star$ a A je neterminál.

  Jazyky generované gramatikami typu 2 se nazývají *bezkontextové jazyky* nebo j*azyky typu 2.*
- *Gramatiky typu 3* neboli *regulární gramatiky* (též *pravé lineární gramatiky*) jsou takové gramatiky, kde každé pravidlo v P je tvaru

  $$A \rightarrow wB,\ A \rightarrow w,$$

  kde A, B jsou neterminály a w je terminální slovo.

  Jazyky generované gramatikami typu 3 se nazývají *regulární jazyky* nebo *jazyky typu 3.*

Poznamenejme, že regulární jazyky již byly definovány jako ty jazyky, které jsou přijímány konečnými automaty — později ukážeme, že je to správně, totiž, že každý jazyk typu 3 je přijímán konečným automatem.

23.1.1.8 Nevypouštěcí gramatiky

Gramatiku $G = (N, \Sigma, S, P)$ nazveme *nevypouštěcí*, jestliže neobsahuje žádné pravidlo typu $A \to \epsilon$.

Tvrzení: Ke každé bezkontextové gramatice G existuje nevypouštěcí gramatika G_1 taková, že

$$L(G) = L(G_1) - \{\epsilon\}.$$

Důsledek: Označme L_i třídu jazyků typu i. Pak platí:

$$L_3 \subseteq L_2 \subseteq L_1 \subseteq L_0.$$

23.2 Regulární jazyky a regulární gramatiky

23.2.1 Tvrzení

Ke každému regulárnímu jazyku L existuje regulární gramatika G, která ho generuje.

23.2.2 Lemma

Ke každé gramatice G typu 3 existuje gramatika G_1 typu 3 generující stejný jazyk a taková, že má pravidla pouze tvaru

$$A \to aB,\ A \to \epsilon.$$

Přidáním nových neterminálů je možné pravidlo $A \to wB$, kde $w = a_1 a_2 \ldots a_k$, nahradit posloupností pravidel $A \to a_1 X_1$, $X_1 \to a_2 X_2, \ldots, X_{k-1} \to a_k B$.

23.2.3 Tvrzení

Ke každé gramatice G typu 3 existuje konečný automat M takový, že

$$L(G) = L(M).$$

23.2.4 Věta

Gramatiky typu 3 generují právě třídu regulárních jazyků.

23.3 Bezkontextové gramatiky

Připomeňme, že bezkontextová gramatika (CFG) je gramatika $G = (N, \Sigma, S, P)$, která obsahuje pouze pravidla typu

$$A \to \gamma, \quad \text{kde } \gamma \in (N \cup \Sigma)^\star \text{ a } A \text{ je neterminál.}$$

Dále připomeňme, že ke každé CFG gramatice G existuje nevypouštěcí CFG gramatika G_1 taková, že

$$L(G_1) = L(G) - \{\epsilon\}.$$

23.3.1 Tvrzení

Máme dánu bezkontextovou gramatiku $G = (N, \Sigma, S, P)$ a v ní derivaci

$$S \Rightarrow_G^\star \alpha A \beta \Rightarrow_G^\star w,$$

pro $\alpha, \beta \in (\Sigma \cup N)^\star$, $A \in N$ a $w \in \Sigma^\star$.
Pak existují slova $u, x, v \in \Sigma^\star$ taková, že

$$w = uxv \quad \text{a} \quad \alpha \Rightarrow_G^\star u, \quad A \Rightarrow_G^\star x, \quad \beta \Rightarrow_G^\star v.$$

23.3.2 Redukovaná bezkontextová gramatika

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$, pro kterou $L(G) \neq \emptyset$. Řekneme, že G je *redukovaná*, jestliže splňuje tyto dvě podmínky:

1. Ke každému neterminálu A existuje aspoň jedno terminální slovo w takové, že $A \Rightarrow_G^\star w$.
2. Ke každému neterminálu A existují slova $\alpha, \beta \in (N \cup \Sigma)^\star$ tak, že $S \Rightarrow_G^\star \alpha A \beta$.

23.3.3 Tvrzení

Ke každé bezkontextové gramatice $G = (N, \Sigma, S, P)$, pro kterou $L(G) \neq \emptyset$, existuje redukovaná gramatika G_1 taková, že $L(G_1) = L(G)$.

23.3.4 Algoritmus redukce CFG

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$.

1. Sestrojíme množinu $V = \{A | A \in N, A \Rightarrow_G^\star w, w \in \Sigma^\star\}$:

$$V_0 = \Sigma,$$

$$V_{i+1} = V_i \cup \{A| \text{ existuje } \alpha \in V_i^\star \text{ takové, že } A \Rightarrow_G^\star \alpha\}.$$

Platí

$$V_O \subseteq V_1 \subseteq V_2 \subseteq \ldots \subseteq (N \cup \Sigma) .$$

Proto existuje n takové, že $V_n = V_{n+1}$. Položíme $V = V_n - \Sigma$.

Jestliže $S \notin V$, pak $L(G) = \emptyset$ a redukovaná gramatika ke gramatice G neexistuje.

Definujeme $G' = (V, \Sigma, S, P')$: do P' dáme pouze ta pravidla z P, která obsahují neterminály z množiny V.

2. Pro gramatiku $G' = (V, \Sigma, S, P')$ zkonstruujeme množinu

$$U = \{A | A \in V, \text{ existují } \alpha, \beta \in (V \cup \Sigma)^\star \text{ tak, že } S \Rightarrow_G^\star \alpha A \beta\} .$$

$$U_0 = \{S\} ,$$

$$U_{i+1} = U_i \cup \{A | \text{existují } B \in U_i, \alpha, \beta \in (V \cup \Sigma)^\star \text{ tak, že } B \Rightarrow_G^\star \alpha A \beta\} .$$

Platí

$$U_0 \subseteq U_1 \subseteq U_2 \subseteq \ldots \subseteq V.$$

Proto existuje n takové, že $U_n = U_{n+1}$. Položíme $U = U_n$.

Hledaná gramatika je gramatika $G_1 = (U, \Sigma, S, P_1)$, kde P_1 je množina všech pravidel z P' (a tedy i z P), které obsahují neterminály pouze z množiny U.

Platí: gramatika $G_1 = (U, \Sigma, S, P_1)$ je redukovaná a generuje stejný jazyk jako původní gramatika $G = (N, \Sigma, S, P)$.

23.3.5 Poznámky

- Uvědomte si, že redukovaná CFG gramatika „nemá zbytečné neterminály".
- Je obtížné ke dvěma bezkontextovým gramatikám zjistit, zda generují stejný jazyk. Redukce gramatik nám k rozhodnutí nepomůže.
- Kroky předchozího postupu nelze zaměnit. Kdybychom nejprve hledali množinu neterminálů U a pak teprve z ní vybírali ty neterminály, ze kterých je možné odvodit terminální slovo, výsledná gramatika by nemusela splňovat druhou podmínku z 23.3.2.

23.3.6 Levá derivace, levé odvození

Přímé odvození se nazývá *levé*, jestliže se přepisuje ten neterminál, který je nejvíc „vlevo”, tj. $uA\beta \Rightarrow_G u\delta\beta$, kde $u \in \Sigma^\star$ a $A \to \delta$ je pravidlo gramatiky.

Derivace (odvození) se nazývá *levá*, jestliže se skládá pouze z levých přímých odvození.

Obdobně definujeme pravé přímé odvození a pravou derivaci.

23.3.7 Tvrzení

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$. Pak pro každou derivaci $S \Rightarrow_G^\star w$ existuje levá derivace terminálního w z S taková, že používá stejná pravidla jako původní derivace (pouze možná v jiném pořadí).

23.3.8 Derivační strom (parse tree)

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$. *Derivační strom* (anglicky *parse tree*) je kořenový strom, takový, že:

1. Každý vrchol, který není list, je ohodnocen neterminálem.

2. Každý list je ohodnocen terminálem nebo prázdným slovem ϵ. V případě, že je list ohodnocen prázdným slovem ϵ, jedná se o jediný následník (svého předchůdce).

3. Jestliže některý vrchol, který není list, je ohodnocen neterminálem A a má následníky (v pořadí od leva do prava) $X_1, X_2, \ldots, X_k$, $X_i \in N \cup \Sigma$, pak $A \to X_1 X_2 \ldots X_k$ je pravidlo gramatiky G.

Řekneme, že derivační strom *dává*, nebo *má za výsledek* slovo w, jestliže w je ohodnocení listů derivačního stromu (čteno od leva do prava).

23.3.9 Tvrzení

1. Pro každou derivaci $S \Rightarrow_G^\star w$ existuje derivační strom s výsledkem w.

2. Ke každému derivačnímu stromu s výsledkem w existuje aspoň jedna derivance $S \Rightarrow_G^\star w$ (takových derivací může být více).

3. Ke každému derivačnímu stromu s výsledkem w existuje právě jedna levá (právě jedna pravá) derivace w z S.

23.3.10 Jednoznačné a víceznačné bezkontextové gramatiky

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$. Řekneme, že G je *jednoznačná*, jestliže pro každé slovo w generované gramatikou G existuje jediný derivační strom s výsledkem w (tj. existuje jediná levá derivace w z S).

V opačném případě mluvíme o *víceznačné* gramatice.

23.3.11 Víceznačný jazyk

Bezkontextový jazyk L se nazývá *víceznačný* (též *podstatně víceznačný*), jestliže každá bezkontextová gramatika, která ho generuje, je víceznačná.

Například jazyk $L = \left\{a^i b^j c^k d^l | i = j \text{ nebo } k = l\right\}$ je podstatně víceznačný.

23.4 Zásobníkové automaty

Zhruba řečeno, zásobníkový automat se skládá z řídící jednotky, která je v jednom z možných stavů, ze vstupní pásky se čtecí hlavou a ze zásobníku. Na základě toho, v jakém stavu se automat nachází, co hlava čte na vstupní pásce a jaký symbol je na vrcholu zásobníku, automat udělá akci: přejde do nového stavu, posune čtecí hlavu o jedno políčko doprava nebo stojí (to v případě, že automat reagoval na prázdné slovo) a vrchol zásobníku nahradí zásobníkovým slovem.

23.4.1 Definice

Zásobníkový automat je sedmice $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, kde

- Q je konečná množina stavů,
- Σ je konečná množina vstupních symbolů,
- Γ je konečná množina zásobníkových symbolů,
- δ přiřazuje každé trojici (q, a, X), $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, $X \in \Gamma$, konečnou množinu dvojic (p, α), kde $p \in Q$ a $\alpha \in \Gamma^\star$. Formálně:

  $$\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \to P_f\left(Q \times \Gamma^\star\right).$$

  ($P_f\left(A\right)$ značí množinu všech konečných podmnožin množiny A.)
- $q_0 \in Q$ je počáteční stav,
- $Z_0 \in \Gamma$ je počáteční zásobníkový symbol a
- $F \subseteq Q$ je množina koncových stavů.

Uvědomte si, že zásobníkový automat tak, jak byl definován, je nedeterministický.

23.4.2 Situace zásobníkového automatu

Je dán zásobníkový automat $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$. *Situace* zásobníkového automatu je trojice (q, u, γ), kde q je stav, u je vstupní slovo a γ je zásobníkové slovo.

Znamená to, že zásobníkový automat je ve stavu q, na vstupní pásce má slovo u a v zásobníku slovo γ s tím, že první písmeno γ je na vrcholu zásobníku.

23.4.3 Jeden krok práce zásobníkového automatu - relace $\vdash_A$

Je dán zásobníkový automat $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, který je v situaci $(q, au, X\gamma)$, kde $a \in \Sigma \cup \{\epsilon\}$, $X \in \Gamma$. Pak A *přejde* do situace $(p, u, \alpha\gamma)$ pro $(p, \alpha) \in \delta(q, a, X)$. Značíme

$$(q, au, X\gamma) \vdash_A (p, u, \alpha\gamma) \quad \text{iff} \quad (p, \alpha) \in \delta(q, a, X).$$

23.4.4 Relace $\vdash_A^\star$

Jeden krok zásobníkového automatu rozšíříme na konečný počet. Automat A *přejde ze situace S do situace S'*, píšeme $S \vdash^\star S'$, právě tehdy, když buď $S = S'$ nebo existuje konečný počet situací $S_1, S_2, \ldots, S_n$ takových, že

$$S \vdash_A S_1, S_1 \vdash_A S_2, \ldots, S_n \vdash_A S'.$$

23.4.5 Jazyk přijímaný prázdným zásobníkem

Je dán zásobníkový automat $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$. *Jazyk přijímaný prázdným zásobníkem* $N(A)$ je definován takto:

$$N(A) = \{u \in \Sigma^\star |\, (q_0, u, Z_0) \vdash_A^\star (p, \epsilon, \epsilon), p \in Q\}.$$

Zhruba řečeno, zásobníkový automat začne v počátečním stavu q_0, na vstupní pásce má slovo u a na zásobníku pouze počáteční zásobníkový symbol Z_0. Slovo je přijato, když po jeho přečtení je (může být) zásobník vyprázdněn.

23.4.6 Jazyk přijímaný koncovým stavem

Je dán zásobníkový automat $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$. *Jazyk přijímaný koncovým stavem* $L(A)$ je definován takto:

$$L(A) = \{u \in \Sigma^\star |\, (q_0, u, Z_0) \vdash_A^\star (p, \epsilon, \gamma), p \in F\}.$$

Zhruba řečeno, zásobníkový automat začne v počátečním stavu q_0, na vstupní pásce má slovo u a na zásobníku pouze počáteční zásobníkový symbol Z_0. Slovo je přijato, když po jeho přečtení je (může být) automat v některém koncovém stavu. To, zda je současně vyprázdněn zásobník nebo není, nehraje roli.

23.4.7 Tvrzení

Ke každému zásobníkovému automatu A existuje zásobníkový automat B takový, že

$$N(A) = L(B).$$

23.4.8 Tvrzení

Ke každému zásobníkovému automatu A existuje zásobníkový automat B takový, že

$$L(A) = N(B).$$

23.4.9 Věta

Ke každé bezkontextové gramatice $G = (N, \Sigma, S, P)$ existuje zásobníkový automat A takový, že

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

Nástin důkazu: Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$. Zkonstruujeme zásobníkový automat s jedním stavem q takto:

- $Q_A = \{q\}, q_0 = q$,
- $\Gamma_A = N \cup \Sigma$,
- $Z_0 = S$,
- $\delta_A(q, \epsilon, X) = \{(q, \alpha) \,|\, X \to \alpha \in P, X \in N\}$,
- $\delta_A(q, a, a) = \{(q, \epsilon)\}$, pro $a \in \Sigma$.

Zhruba řečeno, je-li na vrcholu zásobníku automatu A neterminál X, nahradíme ho v zásobníku některým pravidlem gramatiky G. Je-li na vrcholu zásobníku terminál $a \in \Sigma$, tak v případě, že a je též čten čtecí hlavou, odstraníme ho z vrcholu zásobníku a hlavu posuneme o jedno políčko doprava. Jestliže se terminální písmeno na vrcholu zásobníku nerovná prvnímu čtenému symbolu, automat se neúspěšně zastaví.

Dá se dokázat, že zásobníkový automat A přijme slovo $u \in \Sigma^\star$ prázdným zásobníkem právě tehdy, když je slovo u vygenerováno gramatikou G.

23.4.10 Věta

Ke každému zásobníkovému automatu A existuje bezkontextová gramatika G taková, že

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

Důkaz přechozí věty (jedná se o opačnou implikaci k větě 23.4.9) je obtížnější. Je třeba ho rozdělit do dvou kroků. Nejprve se dokáže, že pro každý zásobníkový automat A existuje zásobníkový automat B s jedním stavem takový, že $N(A) = N(B)$.

Pak už je jednoduché pro zásobníkový automat B s jedním stavem vytvořit bezkontextovou gramatiku G, která generuje stejná slova jako zásobníkový automat B přijal prázdným zásobníkem. Jedná se vlastně o opačný postup jako v důkazu věty 23.4.9.

23.4.11 Deterministický zásobníkový automat

O zásobníkovém automatu $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ řekneme, že je *deterministický*, jestliže splňuje následující dvě podmínky:

- Pro každé $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$ a $X \in \Gamma$ je $\delta(q, a, X)$ nejvýše jednoprvková (tj. $|\delta(q, a, X)| \leq 1$).
- Jestliže pro nějaké $q \in Q$ a $X \in \Gamma$ je $\delta(q, \epsilon, X)$ neprázdné, pak pro každé $a \in \Sigma$ je $\delta(q, a, X)$ prázdná množina.

Uvědomte si, že předchozí dvě podmínky zajišťují, že v každém okamžiku máme vždy nejvýše jednu možnost, jak pokračovat.

23.4.12 Jazyky přijímané deterministickým zásobníkovým automatem

Stejně jako u (nedeterministických) zásobníkových automatů rozlišujeme i u deterministických zásobníkových automatů příjímání koncovým stavem a přijímání prázdným zásobníkem. Tj. pro daný deterministický zasobníkový automat $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ je

$$L(A) = \{u | (q_0, u, Z_0) \vdash_A^\star (p, \epsilon, \gamma), p \in F\}$$

$$N(A) = \{u | (q_0, u, Z_0) \vdash_A^\star (p, \epsilon, \epsilon)\}.$$

23.4.13 Tvrzení

Pro každý deterministický zásobníkový automat A existuje deterministický zásobníkový automat B takový, že

$$N(A) = L(B).$$

Jinými slovy, každý jazyk přijímaný deterministickým zásobníkovým automatem prázdným zásobníkem je také přijímán (nějakým) deterministickým zásobníkovým automatem koncovým stavem.

23.4.14 Bezprefixový jazyk

je jazyk, který je příjímán nějakým deterministickým zásobníkovým automatem prázdným zásobníkem.

Obdoba tvrzení 23.4.8 pro deterministické zásobníkové automaty neplatí. Jestliže totiž deterministický zásobníkový automat A přijme slovo u prázdným zásobníkem, pak nemůže prázdným zásobníkem přijmout žádné slovo uv, kde $v \neq \epsilon$.

23.4.15 Deterministický jazyk

je jazyk, který je přijímán některým deterministickým zásobníkovým automatem koncovým stavem.

23.5 Vlastnosti bezkontextových jazyků

23.5.1 Chomského normální tvar

Je dána bezkontextová gramatika $G = (N, \Sigma, S, P)$. Řekneme, že gramatika G je v *Chomském normálním tvaru*, jestliže má pouze pravidla tvaru

$$A \to BC, A \to a \quad \text{pro } A, B, C \in N, a \in \Sigma.$$

23.5.2 Věta

Pro každou bezkontextovou gramatiku G existuje bezkontextová gramatika $G^{'}$ v Chomského normálním tvaru taková, že

$$L\left(G^{'}\right) = L\left(G\right) - \{\epsilon\}.$$

23.5.3 CYK

Jedná se o algoritmus, který pro danou bezkontextovou gramatiku G v Chomského normálním tvaru a pro dané terminální slovo w rozhodne, zda $w \in L\left(G\right)$.

Označíme $G = (N, \Sigma, S, P)$ a $w = a_1 a_2 \ldots a_k$. Postupně vytváříme množiny $X_{i,j}$ pro $1 \leq i \leq j \leq k$, kde

$$X_{i,j} = \{A \in N | A \Rightarrow^{\star}_{G} a_i a_{i+1} \ldots a_j\}.$$

Platí

$$A \in X_{i,i} \quad \text{iff} \quad A \to a_i \in P.$$

Navíc

$$X_{1,k} = \{A \in N | A \Rightarrow^{\star}_{G} a_1 a_2 \ldots a_k = w\}.$$

Dále si uvědomte, že $A \Rightarrow^{\star}_{G} a_i a_{i+1} \ldots a_j$ iff existují neterminály B, C takové, že

$$A \to BC \in P, \text{ kde} \quad \text{buď} \quad B \Rightarrow^{\star}_{G} a_i \quad \text{a} \quad C \Rightarrow^{\star}_{G} a_{i+1} \ldots a_j$$

$$\text{nebo } B \Rightarrow^{\star}_{G} a_i a_{i+1} \quad \text{a} \quad C \Rightarrow^{\star}_{G} a_{i+2} \ldots a_j$$

$$\text{nebo } B \Rightarrow^{\star}_{G} a_i a_{i+1} a_{i+2} \quad \text{a} \quad C \Rightarrow^{\star}_{G} a_{i+3} \ldots a_j$$

$$\ldots$$

$$\text{nebo } B \Rightarrow^{\star}_{G} a_i \ldots a_{i+j-1} \quad \text{a} \quad C \Rightarrow^{\star}_{G} a_{j,j}.$$

Předpokládejme, že máme zkonstruovány všechny množiny $X_{p,q}$, kde $q-p<n$. Pak množiny $X_{i,j}$ pro $j-i=n$ utvoříme takto:

$$A \in X_{i,j} \text{ iff } \exists A \to BC \in P \text{ tak, že} \quad \text{buď} \quad B \in X_{i,i} \quad \text{a} \quad C \in X_{i+1,j}$$

$$\text{nebo} \quad B \in X_{ii+1} \quad a \quad C \in X_{i+2,j}$$

$$\text{nebo} \quad B \in X_{i,i+2} \quad a \quad C \in X_{i+3,j}$$

$$\dots$$

$$\text{nebo} \quad B \in X_{i,j-1} \quad a \quad C \in X_{j,j}$$

Začínám tedy konstrukcí množin $X_{i,i}$, $i=1,2,\dots,k$, následuje pak $k-1$ množin $X_{i,i+1}$, $i=1,2,\dots,k-1$, atd. dvě množiny $X_{1,k.-1}$, $X_{2,,k}$ a nakonec jednu množinu $X_{1,k}$ a to podle následujícího postupu:

$$X_{i,j} = \{A \in N | \exists A \to BC \in P \text{ tak, že } B \in X_{i,i+m}, C \in X_{i+m+1,j}\}.$$

Platí $w \in L(G)$ právě tehdy, když $S \in X_{1,k}$.

23.5.4 Příklad

Je dána gramatika G pravidly

$$S \to AB|BC$$

$$A \to BA|a$$

$$B \to CC|b$$

$$C \to AB|a$$

Pomocí algoritmu CYK rozhodněte, zda slovo *aabab* je generováno bezkontextovou gramatikou G.

Řešení: Konstrukci množin $X_{i,j}$ pro $1 \le i \le j \le 5$ si znázorníme do tabulky. Tabulka bude mít 5 řádků a 5 sloupců, kde vyplněných bude jen ta část, která se nachází „pod diagonálou". Poslední řádek obsahuje pět množin $X_{1,1}$, $X_{2,2}$, $X_{3,3}$, $X_{4,4}$ a $X_{5,5}$. Předposlední řádek obsahuje čtyři množiny $X_{1,2}$, $X_{2,3}$, $X_{3,4}$ a $X_{4,5}$. Řádek, který je třetí od spodu (a také shora) obsahuje tři množiny $X_{1,3}$, $X_{2,4}$ a $X_{3,5}$. Řádek, který je čtvrtý od spodu (a druhý shora) obsahuje dvě množiny $X_{1,4}$ a $X_{2,5}$. Nejvyšší řádek obsahuje jednu množinu $X_{1,5}$.

S,C				
S,A,C	B			
B	B	S,C		
B	S,C	S,A	S,C	
A,C	A,C	B	A,C	B
a	a	b	a	b

Z předchozí tabulky také můžeme odvodit derivace slova *aabab* gramatikou G. Jedna z takových derivací je např. tato:

$$S \Rightarrow AB \Rightarrow aB \Rightarrow aCC \Rightarrow aaC \Rightarrow aaAB \Rightarrow aaBAB \Rightarrow aabAB \Rightarrow aabaB \Rightarrow aabab.$$

23.5.5 Pumping lemma pro bezkontextové gramatiky

Pro každý bezkontextový jazyk L existuje kladné přirozené číslo m takové, že jestliže některé slovo z obsažené v jazyce L má délku alespoň m, pak z lze psát ve tvaru $z = uvwxy$, kde

- $|vwx| \leq m$, (tj. prostřední část není příliš dlouhá),
- $vx \neq \epsilon$ (tj. aspoň jedno ze slov v, x není prázdné),
- pro všechna $i \geq 0$ platí $uv^iwx^iy \in L$, (tj. v a x se dají do slova z „napumpovat" a stále dostaneme slovo z jazyka L).

23.5.6 Využití Pumping lemmatu pro bezkontextové gramatiky

Ukážeme, že jazyk $L = \{0^n 1^n 2^n | n \geq 0\}$ není bezkontextový.

Zdůvodnění: Předpokládejme, že jazyk L je bezkontextový. Pak existuje kladné číslo m z Pumping lemmatu. V jazyce L leží slovo $z = 0^m 1^m 2^m$. Podle Pumping lemmatu existují slova u, vw, x, y taková, že

$$0^m 1^m 2^m = uvwxy,\ |vx| > 0,\ |vwx| \leq m \text{ a } uv^iwx^iy \in L \text{ pro } i \geq 0.$$

Ukážeme, že slovo $uv^0wx^0y = uwy \notin L$. To bude hledaný spor.

Podmínka $|vwx| \leq m$ znamená, že slovo vwx buď neobsahuje písmeno 2 nebo neobsahuje písmeno 0.

Jestliže vwx neobsahuje písmeno 2, pak slova v, x obsahují pouze 0 nebo 1, to je nejvýše dva ze tří písmen 0,1,2.

Jestliže vwx neobsahuje písmeno 0, pak slova v, x obsahují pouze 1 nebo 2, to je nejvýše dva ze tří písmen 0,1,2.

To ale znamená, že v obou případech nemůže slovo uwy (tj. slovo z, ze kterého jsme vypustili slova v a x) obsahovat stejný počet všech tří písmen 0,1,2.

23.6 Uzávěrkové vlastnosti bezkontextových jazyků

23.6.1 Tvrzení

Bezkontextové jazyky jsou uzavřeny na sjednocení.

To znamená, jsou-li L_1 a L_2 dva bezkontextové jazyky, pak také jazyk $L_1 \cup L_2$ je bezkontextový.

23.6.2 Tvrzení

Bezkontextové jazyky jsou uzavřeny na zřetězení.

To znamená, jsou-li L_1 a L_2 dva bezkontextové jazyky, pak také jazyk $L_1 L_2$ je bezkontextový.

23.6.3 Tvrzení

Bezkontextové jazyky jsou uzavřeny na Kleeneho operaci $\star$.

To znamená, je-li L bezkontextový jazyk, pak také jazyk $L\star$ je bezkontextový.

23.6.4 Tvrzení

Bezkontextové jazyky jsou uzavřeny na reverzi.

To znamená, je-li L bezkontextový jazyk, pak také jazyk L^R je bezkontextový.

23.6.5 Tvrzení

Bezkontextové jazyky nejsou uzavřeny na průnik.

To znamená, jsou-li L_1 a L_2 dva bezkontextové jazyky, pak jazyk $L_1 \cap L_2$ nemusí být bezkontextový.

23.6.6 Tvrzení

Bezkontextové jazyky nejsou uzavřeny na doplněk.

To znamená, je-li L bezkontextový jazyk, pak jeho doplněk $\bar{L}$ nemusí být bezkontextový.

23.6.7 Tvrzení

Třída bezkontextových jazyků je uzavřena na průniky s regulárními jazyky.

To znamená, je-li L bezkontextový jazyk a R regulární jazyk, pak jazyk $L \cap R$ je bezkontextový.

23.6.8 Tvrzení

Třída bezkontextových jazyků je uzavřena na substituce.