Týden 7

Přednáška

Speciální formy bezkontextových gramatik – dokončení

Připomněli jsme si ještě, co jsou *nevypouštějící gramatiky*; tyto gramatiky nemají pravidla typu $X \longrightarrow \varepsilon$. S algoritmem převodu obecné bezkontextové gramatiky G na nevypouštějící gramatiku G' takovou, že $L(G') = L(G) - \{\varepsilon\}$, jste se seznámili na cvičení. Je nám tedy jasná následující věta.

Věta. Ke každé BG G lze sestrojit nevypouštějící gramatiku G' tž. $L(G') = L(G) - \{\varepsilon\}$.

Poté jsme si připomněli gramatiky v *Chomského normální formě*. Taková gramatika má výhradně pravidla ve tvarech $X \longrightarrow YZ$ a $X \rightarrow a$. Tedy na pravé straně každého pravidla je buď jeden terminál nebo dva neterminály. Ilustrovali jsme si konstrukci, která dokazuje následující větu.

Věta. Ke každé BG G lze sestrojit BG G' v ChNF tž. $L(G') = L(G) - \{\varepsilon\}$.

Převod gramatiky G do ChNF se dá rozdělit do čtyř kroků:

- převod na nevypouštějící gramatiku,
- odstranění pravidel typu $X \rightarrow Y$,
- pro každý terminál a přidáme nový neterminál A_a a pravidlo $A_a \rightarrow a$; na každé pravé straně delší než 1 nahradíme a neterminálem A_a,
- každé pravidlo typu $X \rightarrow Y_1Y_2 \ldots Y_n$, kde $n \geq 3$, nahradíme pravidly

  $$\begin{aligned}&X \rightarrow Y_1Z_1,\\ &Z_1 \rightarrow Y_2Z_2,\\ &\ldots\ldots\\ &Z_{n-3} \rightarrow Y_{n-2}Z_{n-2},\\ &Z_{n-2} \rightarrow Y_{n-1}Y_n\end{aligned}$$

  kde $Z_1, Z_2, \ldots, Z_{n-2}$ jsou nově přidané neterminály.

První dva kroky byly přiblíženy na minulém cvičení (druhý v referátu). Na přednášce jsme si (druhý a) zbývající kroky ilustrovali na příkladu následující gramatiky.

$$\begin{aligned}&S \longrightarrow (E) \mid E\\ &E \longrightarrow F + F \mid F \times F\\ &F \longrightarrow a \mid S\end{aligned}$$

Zásobníkové automaty

Uvedli jsme neformálně model „zásobníkový automat“ (ZA) a zkonstruovali jsme konkrétní ZA M přijímající jazyk

$$L = \{wcw^R \mid w \in \{a,b\}^*\}\,.$$

(Sestavili jsme patřičnou množinu instrukcí typu $(q,a,X) \to (q',\alpha)$.)
Zároveň jsme uvedli formální definici zásobníkového automatu jako struktury

$$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)\,,$$

kde Q je konečná množina *stavů*, Σ je konečná *vstupní abeceda*, Γ je konečná *zásobníková abeceda*, $q_0 \in Q$ je *počáteční stav*, $Z_0 \in \Gamma$ je *počáteční zásobníkový symbol* a

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to \mathcal{P}_{fin}(Q \times \Gamma^*)$$

je přechodová funkce (neboli konečná množina instrukcí).
Definovali jsme *konfiguraci ZA M* jako trojici (q,w,α), kde $q \in Q$, $w \in \Sigma^*$, $\alpha \in \Gamma^*$.
Na množině takových konfigurací přirozeně definujeme relaci $\vdash_M$ (odvození v jednom kroku) :

$$(q, aw, X\beta) \vdash_M (q', w, \alpha\beta) \Leftrightarrow \delta(q,a,X) \ni (q',\alpha)$$

(kde $a \in (\Sigma \cup \{\varepsilon\})$, $w \in \Sigma^*$, $\beta \in \Gamma^*$).
Relace $\vdash_M^*$ (odvození v konečně mnoha krocích) je reflexivním a tranzitivním uzávěrem relace $\vdash_M$.
Jazykem rozpoznávaným (nebo též *přijímaným*) zásobníkovým automatem M rozumíme jazyk

$$L(M) = \{\, w \in \Sigma^* \mid (q_0, w, Z_0) \vdash_M^* (q, \varepsilon, \varepsilon) \text{ pro nějaký } q \in Q \,\}\,.$$

Připomněli jsme, že tedy jako základní bereme přijímaní prázdným zásobníkem, a speciálně jsme zdůraznili, že jako základní verze se u zásobníkových automatů berou *nedeterministické* zásobníkové automaty.
Všimli jsme si, že náš výše sestrojený automat byl deterministický, což znamená

- $\delta(q,a,X)$ je vždy nejvýše jednoprvková množina (pro $a \in \Sigma \cup \{\varepsilon\}$) (tedy neexistují dvě různé instrukce se stejnou levou stranou),
- je-li $\delta(q,\varepsilon,X) \neq \emptyset$, pak $\delta(q,a,X) = \emptyset$ pro vš. $a \in \Sigma$ (může-li automat udělat ε-krok, nemůže mu jiná instrukce zároveň umožňovat přečtení vstupního symbolu).

Pak jsme náš konkrétní ZA upravili tak, aby rozpoznával jazyk

$$L' = \{ww^R \mid w \in \{a,b\}^*\}$$

a uvědomili jsme si, že vzniklý automat už není deterministický. Uvedli jsme si, že L' nelze rozpoznat žádným deterministickým ZA, tedy že L' nepatří do třídy DCFL.
Pak jsme si uvedli větu

Věta. (Nedeterministické) zásobníkové automaty rozpoznávají právě bezkontextové jazyky, tedy jazyky z třídy CFL.

Ukázali jsme si ji jen v jednom směru (k bezkontextové gramatice G lze sestrojit [dokonce jen jednostavový] ZA M tak, že $L(G) = L(M)$) a ilustrovali jsme na příkladu gramatiky

1/ $A \longrightarrow A + B$
2/ $A \longrightarrow B$
3/ $B \longrightarrow B * C$
4/ $B \longrightarrow C$
5/ $C \longrightarrow (A)$
6/ $C \longrightarrow a$

Použili jsme tedy obecný postup

Pro BG $G = (\Pi, \Sigma, S, P)$ sestrojme $M = (\{q_0\}, \Sigma, \Pi \cup \Sigma, \delta, q_0, S)$, kde
pro $X \in \Pi$: $\delta(q_0, \varepsilon, X) = \{ (q_0, \alpha) \mid (X \to \alpha) \in P \}$
a pro $a \in \Sigma$: $\delta(q_0, a, a) = \{ (q_0, \varepsilon) \}$.
(Všude jinde δ přiřazuje $\emptyset$.)

Ilustrovali jsme běh sestrojeného ('velmi' nedeterministického) ZA na slově $a * (a + a)$ a také fakt, že tento běh odpovídá konstrukci příslušného derivačního stromu metodou „shora dolů“.

Nebezkontextové jazyky

Metodou obrázků derivačních stromů z části 5.3. jsme si ukázali, proč jazyk

$$L = \{a^n b^n c^n \mid n \geq 0\}$$

není bezkontextový. Porozuměli jsme tak pumping lemmatu (neboli $uvwxy$-teorému) pro bezkontextové jazyky:

Věta. Nechť L je bezkontextový jazyk. Pak existuje přirozené číslo n tž. každé slovo $z \in L$, $|z| \geq n$, (tedy každé 'dlouhé' slovo z jazyka L) lze psát ve tvaru $z = uvwxy$, přičemž platí

- $vx \neq \varepsilon$,
- $|vwx| \leq n$,
- pro vš. $i \geq 0$ je $uv^i wx^i y \in L$.

Intuici k poznání nebezkontextových jazyků jsme si posílili zvážením jazyků $L_1 = \{ww \mid w \in \{0,1\}^*\}$, $L_2 = \{ 0^m 1^n 0^m \mid m = 2n \}$, u nichž obou jsme usoudili, že ne všechna jejich dlouhá slova lze napsat v patřičném tvaru $uvwxy$, a že tedy nejsou bezkontextové. (U jazyka L_1 jde např. o slova tvaru $0^n 1^n 0^n 1^n$.)

Uzávěrové vlastnosti třídy CFL

Připomněli jsme si, že snadno umíme ukázat, že CFL je uzavřena vůči sjednocení.
Pak jsme si všimli, že jazyky $L_1 = \{a^i b^j c^k \mid i = j\}$ a $L_2 = \{a^i b^j c^k \mid j = k\}$ jsou bezkontextové, ale $L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 0\}$ je jazyk, o němž víme, že bezkontextový není.
Tedy CFL není uzavřena na průnik. Díky de Morganovým pravidlům ihned vidíme, že CFL není uzavřena ani na doplněk.
(Jelikož $(L_1 \cap L_2 = \overline{(\overline{L_1} \cup \overline{L_2})})$, tak z uzavřenosti na doplněk by díky uzavřenosti na sjednocení vyplynula uzavřenost na průnik.)

Sekce pro hlubší zájemce – důkazy

Existence vnitřně nejednoznačných bezkontextových jazyků
Uvažujme jazyk

$$L = \{a^k b^k c^m d^m \mid k, m \geq 1\} \cup \{a^k b^m c^m d^k \mid k, m \geq 1\}.$$

Ten je sice bezkontextový, ale dokážeme, že jej negeneruje žádná jenoznačná bezkontextová gramatika.
Pro odvození sporu předpokládejme, že L je generován jednoznačnou bezkontextovou gramatikou $G = (\Pi, \Sigma, S, P)$.
G můžeme zredukovat (odstranit zbytečné neterminály), přičemž zůstává jednoznačná. (Množina derivačních stromů pro slova z $L = L(G)$ se nezměnila.) Lze rovněž snadno ověřit, že standardní konstrukce odstranění ε-pravidel a odstranění pravidel typu $X \longrightarrow Y$ také nemohou způsobit nejednoznačnost. Takže rovnou předpokládáme, že G nemá taková pravidla.
Nechť nyní pro neterminál $A \neq S$ neplatí $A \Rightarrow^+ u_1 A u_2$ (pro terminální slova u_1, u_2, kde tedy alespoň jedno neprázdné). Tedy na pravých stranách pravidel $A \longrightarrow \alpha_1 \mid \cdots \mid \alpha_m$ se nevyskytuje A. Nahradíme-li nyní pravé strany (jiných pravidel) obsahující A všemi možnosti, při nichž výskyty A nahrazujeme řetězci $\alpha_1, \ldots, \alpha_m$, stane se A nedosažitelným z S a je možné jej z gramatiky odstranit. Jednoznačnost opět nemohla být porušena.
Takže dále rovnou předpokládáme, že v G pro každý neterminál $A \neq S$ platí $A \Rightarrow^+ u_1 A u_2$.
Pro libovolné $A \Rightarrow^+ u_1 A u_2$ snadno vypozorujeme následující:

- u_1 obsahuje nanejvýš jeden ze symbolů a, b, c, d; podobně u_2.
- u_1 obsahuje jeden symbol a u_2 jiný.
- $|u_1| = |u_2|$
- jestliže $A \Rightarrow^+ u_3 A u_4$, pak u_1 a u_3 obsahují tentýž symbol, podobně u_2 a u_4.
- jsou jen následující možnosti:

  - $u_1 \in \{a\}^+$ a $u_2 \in \{b\}^+$,
  - $u_1 \in \{a\}^+$ a $u_2 \in \{d\}^+$,
  - $u_1 \in \{b\}^+$ a $u_2 \in \{c\}^+$,
  - $u_1 \in \{c\}^+$ a $u_2 \in \{d\}^+$,

- V množině neterminálů Π tedy lze přirozeně definovat příslušné třídy $\mathcal{C}_{ab}$, $\mathcal{C}_{ad}$, $\mathcal{C}_{bc}$, $\mathcal{C}_{cd}$. Tyto třídy jsou očividně disjunktní a S nepatří do žádné z nich. Neterminál z jedné třídy se může přepsat na neterminál z jiné třídy jedině v případě pravidla typu $X \longrightarrow v_1 Y v_2$, kde $X \in \mathcal{C}_{ad}$ a $Y \in \mathcal{C}_{bc}$.

- Pravidla $S \longrightarrow \alpha$ jsou jedině v některém z tvarů

  - α je terminální řetězec ($a^k b^k c^m d^m$ nebo $a^k b^m c^m d^k$),
  - α obsahuje jen jeden neterminál a ten je z třídy $\mathcal{C}_{ab}$ či $\mathcal{C}_{cd}$, nebo dva neterminály, kde levější je z třídy $\mathcal{C}_{ab}$ a ten druhý z $\mathcal{C}_{cd}$; v tom případě každé slovo generované z α musí být z jazyka $\{a^k b^k c^m d^m \mid k, m \geq 1\}$.
  - α obsahuje jen jeden neterminál a ten je z třídy $\mathcal{C}_{bc}$ či $\mathcal{C}_{ad}$; v druhém případě pak později může dojít k přepsání neterminálu z třídy $\mathcal{C}_{ad}$ na neterminál z třídy $\mathcal{C}_{bc}$. Zde každé slovo generované z α musí být z jazyka $\{a^k b^m c^m d^k \mid k, m \geq 1\}$.

Vypusťme teď z G pravidla typu $S \longrightarrow a^n b^n c^n d^n$ a rozdělme pravidla (disjunktně) do dvou gramatik G_1, G_2 tak, že G_1 generuje všechna $a^k b^k c^m d^m$ pro $k \neq m$, plus případně $a^n b^n c^n d^n$ pro nějaká n

a G_2 generuje všechna $a^k b^m c^m d^k$ pro $k \neq m$, plus případně $a^n b^n c^n d^n$ pro nějaká n.

Ukážeme, že ve skutečnosti G_1 generuje $a^n b^n c^n d^n$ pro skoro všechna n (tedy až na konečně mnoho výjimek). Podobně ukážeme, že G_2 generuje $a^n b^n c^n d^n$ pro skoro všechna n. To ovšem bude spor s jednoznačností G.

Předpokládejme tedy, že $J = \{n \mid G_1 \text{ negeneruje } a^n b^n c^n d^n\}$ je nekonečná. Očíslujme si pravidla $S \longrightarrow \alpha$ z G_1 jako $r_1, r_2, \ldots, r_s$. Pro každé $n_0 \in J$ odvození v G_1 podle schématu $S \Rightarrow \alpha \Rightarrow^* a^k b^k c^m d^m$, kde $S \longrightarrow \alpha$ je pravidlo r_1

neodvodí buď slovo typu $a^{n_0} b^{n_0} c^m d^m$ nebo slovo typu $a^k b^k c^{n_0} d^{n_0}$ (jinak by se dalo odvodit i $a^{n_0} b^{n_0} c^{n_0} d^{n_0}$, jak plyne z možných tvarů pravidla r_1, které jsme zmínili výše).

Existuje tedy nekonečná množina $J_1 \subseteq J$ taková, že pro každé $k, m \in J_1$ neexistuje odvození $S \Rightarrow^* a^k b^k c^m d^m$ začínající pravidlem r_1 (tedy odvození podle schématu 1). Dále ovšem pro každé $n_0 \in J_1$ odvození v G_1 začínající pravidlem r_2 neodvodí buď slovo typu $a^{n_0} b^{n_0} c^m d^m$ nebo slovo typu $a^k b^k c^{n_0} d^{n_0}$ (jinak by se dalo odvodit i $a^{n_0} b^{n_0} c^{n_0} d^{n_0}$).

Tedy existuje nekonečná množina $J_2 \subseteq J_1$ taková, že pro každé $k, m \in J_2$ neexistuje odvození $S \Rightarrow^* a^k b^k c^m d^m$ začínající (pravidlem r_1 nebo) pravidlem r_2.

Když takto konstruujeme $J \supseteq J_1 \supseteq J_2 \supseteq \cdots \supseteq J_s$, zjistíme, že pro každé $k \neq m$, kde $k, m \in J_s$, platí, že G_1 nevygeneruje $a^k b^k c^m d^m$, což je spor.

Pro G_2 postupujeme obdobně, ale číslujeme jinak. Nejprve očíslujeme pravidla typu $S \longrightarrow \alpha$, kde α je terminální řetězec nebo obsahuje neterminál z $\mathcal{C}_{bc}$. Pravidlu r_i tohoto typu odpovídá schéma
$S \Rightarrow \alpha \Rightarrow^* a^k b^m c^m d^k$.
Další pořadová čísla přiřadíme (konečně mnoha) schématům
$S \Rightarrow \alpha \Rightarrow^* u_1 A u_2 \Rightarrow u_1 v_1 B v_2 u_2 \Rightarrow a^k b^m c^m d^k$
kde $A \in \mathcal{C}_{ad}$ a $B \in \mathcal{C}_{bc}$
(toto schéma je určeno dvěma pravidly, tedy $S \longrightarrow \alpha$ a $A \longrightarrow v_1 B v_2$).
Opět předpokládejme, že $J = \{n \mid G_2 \text{ negeneruje } a^n b^n c^n d^n\}$ je nekonečná. Pak pro každé $n_0 \in J$ odvození v G_1 podle prvního schématu neodvodí buď slovo typu $a^{n_0} b^{n_0} c^m d^m$ nebo slovo typu $a^k b^k c^{n_0} d^{n_0}$ (jinak by se očividně dalo odvodit i $a^{n_0} b^{n_0} c^{n_0} d^{n_0}$).
Existuje tedy nekonečná množina $J_1 \subseteq J$ taková, že pro každé $k, m \in J_1$ neexistuje odvození $S \Rightarrow^* a^k b^m c^m d^k$ podle prvního schématu. Atd.

Partie textu k prostudování

Zásobníkové automaty (část 4.5.) a jejich varianty, uzávěrové vlastnosti CFL (část 5.2.). Nebezkontextové jazyky (část 5.3.).
(Máte si udělat přinejmenším dobrou první představu a zamyslet se nad příklady, speciálně těmi plánovanými na cvičení, ať se můžete na cvičení aktivně účastnit a případné problémy si tam objasnit.)

Cvičení

Prezentace referátů

Referát č. 13 (Pumping lemma pro regulární jazyky)
Vysvětlete tzv. pumping lemma pro regulární jazyky
(můžete vyjít např. z `http://www.cs.vsb.cz/jancar/TEORET-INF/teoret-inf.pdf`)
a naznačte na příkladu, jak jej lze využít pro důkaz neregularity nějakého jazyka.
Referát č. 14 (Pumping lemma pro bezkontextové jazyky)
Připomeňte pumping lemma pro bezkontextové jazyky (např. ze studijního textu) a vysvětlete souvislost se hrou dvou hráčů popsanou např. v
`http://www.cs.vsb.cz/jancar/TEORET-INF/teoret-inf.pdf`.
Referát č. 15 (Zásobníkové automaty)
Tvrzení „Ke každému ZA M lze sestrojit ZA M' s jedním stavem tž. $L(M) = L(M')$“
lze dokázat pomocí následující obecně popsané konstrukce.
Jednostavový ZA M', se stavem označeným s, bude mít zásobníkové symboly typu $\langle p, X, q\rangle$, kde p, q jsou stavy a X je zásobníkový symbol automatu M, a speciální počáteční zásobníkový symbol R.

Konkrétně pro $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ konstruujeme $M' = (\{s\}, \Sigma, \Gamma', \delta', s, R)$, kde $\Gamma' = (Q \times \Gamma \times Q) \cup \{R\}$ a δ' je určena následovně:

- $\delta'(s, \varepsilon, R) = \{(s, \langle q_0, Z_0, q\rangle) \mid q \in Q\}$,
- pro $(q', \varepsilon) \in \delta(q, a, X)$ $(a \in (\Sigma \cup \{\varepsilon\}))$ zařadíme do $\delta'(s, a, \langle q, X, q'\rangle)$ prvek (s, ε),
- pro $(q', A_1A_2 \dots A_n) \in \delta(q, a, X)$ $(n \geq 1)$ zařadíme do $\delta'(s, a, \langle q, X, \overline{q}\rangle)$ prvek $(s, \langle q', A_1, q_1\rangle\langle q_1, A_2, q_2\rangle \dots \langle q_{n-1}, A_n, \overline{q}\rangle)$ pro každé $\overline{q}, q_1, q_2, \dots, q_{n-1} \in Q$.

(Chápeme-li δ' jako množinu 'instrukcí', pak lze říci, že δ' je minimální množina instrukcí splňující výše uvedené podmínky.)
(Dá se ověřit, že

$$(s, w, \langle p, X, q\rangle) \vdash^*_{M'} (s, \varepsilon, \varepsilon) \Longleftrightarrow (p, w, X) \vdash^*_{M} (q, \varepsilon, \varepsilon)\,, \tag{1}$$

a tedy každému přijímajícímu výpočtu automatu M nad slovem w odpovídá přijímající výpočet automatu M' nad w a naopak.)
Vaším úkolem je předvést aplikaci této konstrukce na zásobníkový automat M se vstupní abecedou $\{a, b\}$, zásobníkovou abecedou $\{A, B\}$, počátečním zásobníkovým symbolem A, množinou stavů $\{p, q, r\}$, počátečním stavem p a přechodovou funkcí δ definovanou následovně
$\delta(p, a, A) = \{(q, AA), (p, B)\}$,
$\delta(q, b, A) = \{(q, AA)\}$,
$\delta(p, \varepsilon, B) = \{(q, A)\}$,
$\delta(q, \varepsilon, A) = \{(r, \varepsilon)\}$,
$\delta(r, a, A) = \{(r, A)\}$,
$\delta(r, b, A) = \{(r, \varepsilon)\}$
(pro ostatní prvky def. oboru je funkční hodnota rovna $\emptyset$).
Přitom se snažte alespoň intuitivně přiblížit platnost uvedeného vztahu (1) v onom konkrétním případě.

Příklady

Příklad 7.1

Na přednášce jsme mj. ukázali, že třída CFL není uzavřena na doplněk. Existuje tedy jazyk $L \subseteq \Sigma^*$ (pro nějakou abecedu Σ), který je bezkontextový, přičemž jeho doplněk $\overline{L} = \Sigma^* - L$ bezkontextový není.
Víme, že jazyk $L_1 = \{ww \mid w \in \{a, b\}^*\}$ bezkontextový není. Zkuste prokázat, že $L_2 = \overline{L_1}$ bezkontextový je a představuje tak konkrétní příklad jazyka z CFL, jehož doplněk v CFL není.

Nápověda. Vystihněte co nejjednodušeji, jak vypadá slovo z L_2, které má sudou délku, tedy $2d$ pro nějaké $d \geq 1$. Pak se zkuste zamyslet nad následujícím vztahem:

$$2d = d_1 + 1 + d_2 + d_1 + 1 + d_2 = d_1 + 1 + d_1 + d_2 + 1 + d_2\,.$$

Nakonec si tipněte, jestli se může podařit navrhnout *deterministický* zásobníkový automat přijímající L_2.
(Pomůže vám nějak u předchozí otázky, když se dozvíte, že třída DCFL je uzavřena vůči doplňku?)

Příklad 7.2

Připomeňte si, jak se ukáže uzavřenost CFL vůči sjednocení, zřetězení a iteraci. Pak si promyslete a ukažte, jak se prokáže uzavřenost vůči zrcadlovému obrazu. (Pro $L \in CFL$ platí i $L^R \in CFL$.)

Příklad 7.3

(Na základě alespoň intuitivních argumentů) zjistěte, které z daných jazyků
jsou regulární:
jsou bezkontextové, ale ne regulární:
nejsou bezkontextové:

$L_1 = \{w \in \{a,b\}^* \mid |w|_a = |w|_b\}$
$L_2 = \{w \in \{a,b\}^* \mid |w|_a \text{ je sudé }\}$
$L_3 = \{w \in \{a,b\}^* \mid w \text{ obsahuje podslovo } abba\}$
$L_4 = \{w \in \{a,b,c\}^* \mid |w|_a = |w|_b = |w|_c\}$
$L_5 = \{w \in \{a,b\}^* \mid |w|_a \text{ je prvočíslo}\}$
$L_6 = \{\, 0^m 1^n \mid m \leq 2n \,\}$
$L_7 = \{\, 0^m 1^n 0^m \mid m = 2n \,\}$

Příklad 7.4

V příkladu 4.3. jste zkonstruovali gramatiku generující jazyk

$$L = \{\, w \in \{a,b\}^* \mid |w| \geq 1 \text{ a } |w|_a = |w|_b \}\,.$$

Připomeňte si ji a pak na ni aplikujte obecný postup, který k bezkontextové gramatice sestrojí ekvivalentní (jednostavový) zásobníkový automat. Takto sestrojíte (nedeterministický) ZA M.
Pak zkuste (co nejefektivnější) přímý návrh ZA M'. Podaří se vám navrhnout M' deterministický? Jestli ano, podaří se vám takový deterministický automat s jedním stavem?