1. opravná písomka z Formálnych Jazykov a Automatov

utorok 7. 1. 2003 (zimný semester 2002/03)

myšlienky riešení ©MišoF.

1 Daný je nedet. konečný automat $A = (K, \Sigma, \delta, q_0, F)$, kde $K = \{q_0, \dots, q_6\}$, $\Sigma = \{a, b\}$, $F = \{q_2, q_4\}$, δ-funkcia je daná tabuľkou:

δ:	q_0	q_1	q_2	q_3	q_4	q_5	q_6
ε	q_1, q_5	q_3		q_4		q_6	q_5
a		q_0	q_3	q_5	q_3	q_1, q_2	q_6
b	q_2	q_2			q_1		

Štandard. konštrukciou zostrojte ekvivalentný nedet. konečný automat, ktorý nepoužíva prechody na ε.

Čo dodať? Použijeme štandardnú konštrukciu z prednášky – zostrojíme ε-chvosty a pridáme príslušné hrany. Nezabudnúť na množinu akceptačných stavov.

2 Rozhodnite, či je jazyk

$$L = \{w \mid w \in \{a, b\}^* \wedge \#_a(w) = 2\#_b(w)\}$$

bezkontextový. Ak nie, dokážte. Ak áno, zostrojte bezkontextovú gramatiku, ktorá bude tento jazyk generovať. Súčasťou riešenia by mal byť aj slovný komentár k množine pravidiel a myšlienka dôkazu správnosti.

Jazyk je bezkontextový. Jedna možná konštrukcia gramatiky je zostrojiť pre L najskôr zásobníkový automat. Ten je pomerne triviálny – na zásobníku si udržuje, čoho a o koľko viac doteraz prečítal. Okrem iného táto úloha bola v jednej sade domácich úloh. No a na automat stačilo použiť štandardnú konštrukciu.

Samozrejme existuje aj viacero spôsobov, ako zostrojiť priamo gramatiku. Pravidlá hľadanej gramatiky mohli vyzerať napr. nasledovne:

$$\sigma \to a\sigma a\sigma b\sigma \mid a\sigma b\sigma a\sigma \mid b\sigma a\sigma a\sigma \mid \varepsilon$$

V každom kroku buď nevygenerujeme nič, alebo vygenerujeme dve a a jedno b, z toho je jedna inklúzia zrejmá. Na druhú treba ukázať, že keď máme ľub. slovo $w \in L$, vieme ho zapísať v jednom z tvarov $auavbx, aubvax, buavax, \varepsilon$ (kde $u, v, x \in L$).

Formálny dôkaz by mohol vyzerať približne nasledovne: Nech $w = w_1 \dots w_n$, položme $f(i) = \#_a(w_1 \dots w_i) - 2\#_b(w_1 \dots w_i)$. Zjavne $f(0) = 0$ a keďže $w \in L$, aj $f(n) = 0$. Prechádzajme teraz s i od 0 do n, v každom kroku sa hodnota f buď zväčší o 1 (ak je príslušný znak a), alebo zmenší o 2 (ak je to b). Ak je $w_1 = b$, je $f_1 = -2$. Zoberme i, kedy je prvýkrát $f(i) = 1$, $j > 0$ kedy je prvýkrát $f(j) = 0$. Zjavne takéto i, j existujú a $i < j$. (Prečo?) Potom ale $w_i = w_j = a$ a tieto dve a rozdelia naše slovo na hľadané 3 časti u, v, x. Ak $w_1 = a$, analogicky.

Trochu iné riešenie. Upravme pravidlá našej gramatiku do tvaru:

$$\sigma \to \sigma a\sigma a\sigma b\sigma \mid \sigma a\sigma b\sigma a\sigma \mid \sigma b\sigma a\sigma a\sigma \mid \varepsilon$$

Jedna inklúzia je naďalej zrejmá. K druhej: Všimnime si, že každé slovo z L obsahuje ako podslovo aspoň jedno zo slov aab, aba, baa (prečo?) a jeho vynechaním dostaneme opäť slovo z L. Indukciou od dĺžky w dokážeme, že ak $w = w_1 \dots w_n \in L$, tak $\sigma \Rightarrow^* \sigma w_1 \sigma w_2 \dots w_n \sigma$. Indukčný krok vyplýva z práve uvedeného pozorovania – keď máme w, nájdeme podslovo, vynecháme ho, z ind. predpokladu vieme odvodiť vetnú formu pre kratšie slovo, ktoré sme vynechaním dostali, no a už len v 1 kroku prepíšeme príslušnú σ.

3 Pre dané jazyky L_1, L_2 nad abecedou Σ definujeme operáciu Shuffle nasledovne:

$$\text{Shuffle}(L_1, L_2) = \Big\{ w \;\Big|\; \exists n \in N, u_1, \dots, u_n, v_1, \dots, v_n \in \Sigma^*; \; w = u_1 v_1 u_2 v_2 \dots u_n v_n \;\wedge\; u_1 u_2 \dots u_n \in L_1 \;\wedge\; v_1 v_2 \dots v_n \in L_2 \Big\}$$

Neformálne povedané, jazyk Shuffle(L_1, L_2) obsahuje všetky slová, ktoré vzniknú zmiešaním slova z L_1 a slova z L_2. Rozhodnite a dokážte, či je trieda regulárnych jazykov uzavretá na operáciu Shuffle. (Teda či $\forall L_1, L_2 \in \mathcal{R};\ \text{Shuffle}(L_1, L_2) \in \mathcal{R}$.)

Správna odpoveď je áno. Majme deterministické automaty A_1, A_2 také, že $L(A_i) = L_i$.
Označme $A_1 = (K_1, \Sigma_1, \delta_1, q_{01}, F_1)$, $A_2 = (K_2, \Sigma_2, \delta_2, q_{02}, F_2)$. Zostrojme automat A pre Shuffle(L_1, L_2).
$A = (K, \Sigma, \delta, [q_{01}, q_{02}], F)$
$K = K_1 \times K_2, \quad F = F_1 \times F_2, \quad \Sigma = \Sigma_1 \cup \Sigma_2$
$\forall [q_a, q_b] \in K; \forall x \in \Sigma; \quad \delta([q_a, q_b], x) = \Big\{[\delta_1(q_a, x), q_b], [q_a, \delta_2(q_b, x)]\Big\}$
Neformálna myšlienka konštrukcie: Náš automat si v stave pamätá stavy oboch automatov A_1, A_2. Vždy, keď mu príde písmeno, nedeterministicky sa rozhodne, v ktorom z automatov spraví krok. Ak sa mu podarí natipovať také rozdelenie slova, že A_1 aj A_2 skončí v akceptačnom stave, akceptuje. (Tie písmená, na ktoré robil kroky v A_1 tvoria u, ostatné tvoria v.)
Formálny dôkaz by mal vyzerať tak, že dokážeme obe inklúzie medzi $L(A)$ a Shuffle(L_1, L_2). Myšlienka dôkazu jednej inklúzie je v predchádzajúcom odseku, druhú inklúziu dokážeme tak, že zoberieme slovo z Shuffle(L_1, L_2), to sa musí dať rozdeliť na u a v, pre ktoré existujú akc. výpočty v A_1, resp. A_2, z tých zostrojíme akc. výpočet A na w.

$\boxed{4}$ Odpovede vyznačte krížikmi do príslušných políčok, nemusíte ich zdôvodňovať. Bodovanie: správna odpoveď +1, nesprávna -1, bez odpovede 0. Ak budete mať záporný súčet, získavate za túto úlohu 0 bodov.

	áno	**nie**
Homomorfizmov z $\{a,b,c,d\}^*$ do $\{a,b\}^*$, ktoré zobrazia *bacaab* na *abbab* je viac ako 5.	x	
Existuje homomorfizmus h taký, že $h^{-1}(\varepsilon) = \varepsilon$.		x
Pre každý homomorfizmus h a jazyk L je $h^{-1}(h(L)) = L$.		x
Ak L spĺňa pumpovaciu lemu pre bezkontextové jazyky, tak je bezkontextový.		x
Ku každému zásob. automatu akcept. prázdnou pamäťou existuje ekviv. s 1 stavom	x	
Ku každej bezkontextovej gramatike existuje ekvivalentná s 1 neterminálom.		x
Nech $L_1, L_2, \ldots \in \mathcal{R}$, $L = \bigcap_{i=1}^{\infty} L_i$. Potom aj $L \in \mathcal{R}$.		x
Všetky nekonečné podmnožiny nekonečného regulárneho jazyka sú regulárne.		x
Trieda rekurzívnych jazykov je uzavretá na komplement.	x	
Možných konfigurácií deterministického konečného automatu je nekonečne veľa.	x	

- d nemá predpísaný obraz, takže takých homomorfizmov je nekonečne veľa
- ...ale existuje taký, že $h^{-1}(\varepsilon) = \{\varepsilon\}$
- stačí vziať $L = \{a\}$ a h taký, že $h(a) = \varepsilon$
- protipríklad bol aj je na webe, pumpovacia lema je opačná implikácia
- stačí spraviť štandardnú konštrukciu na gramatiku a späť
- pre jazyk $\{a^n \mid n \geq 0\} \cup \{b^n \mid n \geq 0\}$ nám jeden neterminál stačiť nebude (prečo?)
- Regulárne jazyky sú uzavreté na operáciu prieniku **dvoch** (prípadne konečne), nie nekonečne veľa jazykov! Nekonečným prienikom vieme dostať ľubovoľný jazyk L: Nech $L^C = \{w_1, w_2, \ldots\}$. Položme $L_i = \Sigma^* \setminus \{w_i\}$. Každý L_i je zjavne regulárny. Keď ale spravíme ich prienik, L_i z neho vyškrtne slovo w_i. V prieniku teda zostanú práve slová (ľubovoľného, nie nutne regulárneho) jazyka L.
- TS A' pre L^C simuluje TS A pre L, keď zastane A, zastane aj A' a akc. práve vtedy, keď A neakc.
- konfigurácia je dvojica (*stav*, *slovo*), takých je ∞ (výpočet je relácia na konfiguráciách, pamätáte?)

$\boxed{5}$ Daný je zásobníkový automat A, akceptujúci prázdnou pamäťou. K nemu zostrojte štandardnou konštrukciou bezkontextovú gramatiku G takú, aby $L(G) = N(A)$. Stručne popíšte, prečo štandardná konštrukcia funguje. (Napr. čo znamenajú neterminály ňou vytvorenej gramatiky.) *Pozn.: Vrch zásobníka je napravo.*
$A = (K, \Sigma, \Gamma, \delta, q_0, C, F), \quad \Sigma = \{a, b\},\ \Gamma = \{A, B, C\},\ K = \{q_0, \ldots, q_4\},\ F = \emptyset$
$\delta(q_0, a, C) = \{(q_1, ACA), (q_2, \varepsilon)\} \qquad \delta(q_1, b, A) = \{(q_2, BB)\}$
$\delta(q_2, \varepsilon, B) = \{(q_0, C), (q_2, B)\} \qquad \forall q \in K;\ \delta(q, a, C) = \{(q, AA), (q, CB)\}$

Gramatika má neterminály tvaru $[q, Z, q']$. Tento neterminál predstavuje kus výpočtu, ktorý vyzerá nasledovne: *sme v stave q_0, na vrchu zásobníka je Z a v okamihu, keď bude prvýkrát zásobník o 1 nižší ako teraz budeme v stave q'.* (Slovu *predstavuje* treba rozumieť tak, že z tohto neterminálu sa budú dať vygenerovať práve také slová, na ktoré mohla táto časť výpočtu prebehnúť.)
Na začiatku sa musí naša gramatika rozhodúť, v akom stave simulovaný automat skončí. Preto počiatočný neterminál bude σ a pravidlá z neho $\forall q; (\sigma \to [q_0, Z_0, q]) \in P$.

$\boxed{6}$

Rozhodnite, či existuje deterministický Turingov stroj, ktorý na každom vstupe zastaví a ak na vstupe dostane trojicu $\Big[$kód (ľubovoľného deterministického) Turingovho stroja $\langle A\rangle$, kód vstupného slova w, kód znaku $x \in \Gamma_A\Big]$, tak akceptuje práve vtedy, ak TS A počas výpočtu na slove w niekedy zapíše na pásku znak x. Ak takýto stroj existuje, stručne popíšte jeho konštrukciu. Ak neexistuje, dokážte. Môžete sa odvolať na vety dokázané na prednáške.

Taký TS neexistuje, inými slovami tento problém nie je rozhodnuteľný.
Sporom. Nech taký TS M existuje. Zostrojme TS M', ktorý bude využívať M, vždy zastaví a bude akceptovať univerzálny jazyk $L_U = \{\langle A\rangle \#w \mid w \in L(A)\}$. To bude spor so skutočnosťou, že $L_U \notin \mathcal{L}_{REC}$, ktorá bola dokázaná na prednáške.
(Dohodnime sa, že TS akceptuje, keď zastane v akc. stave. Ak by sme chceli, nech akceptuje akonáhle sa dostane do akc. stavu, nasledujúca konštrukcia by vyzerala trochu ináč.) Náš stroj M' má o stroji A rozhodnúť, či akceptuje slovo w. Preto upraví kód stroja A nasledovne:

- do pracovnej abecedy pridá nový symbol $\heartsuit$
- do množiny stavov pridá nový stav $q_\heartsuit$, množinu akc. stavov zmení na $\{q_\heartsuit\}$
- do δ-fcie z každého bývalého akc. stavu na každé chýbajúce písmeno (t.j. na tie, na ktoré by bol zastal) pridá pravidlo, v ktorom stroj prejde do $q_\heartsuit$, zapíše na pásku $\heartsuit$ a hlavou zostane stáť na mieste

Takto dostane kód nejakého stroja A'. Zjavne $L(A) = L(A')$. Takisto zjavne vie M' urobiť všetky tieto zmeny v kóde A v konečnom čase.
Všimnime si, že A' akceptuje w práve vtedy, keď sa počas výpočtu A' na w objaví na páske $\heartsuit$. Ak teda M' chce povedať, či A akceptuje w, stačí mu vyššie popísaným spôsobom vyrobiť kód A' a opýtať sa M na trojicu $[\langle A'\rangle, w, \heartsuit]$.