1. príklad

(a) Nájdite príklad jazykov A, B, pre ktoré platí: $A \neq B$ ale $A^2 = B^2$.

(b) Zapíšte jazyk $L = \{0^{m+4}\, 1^{2m}\, 2^{m+3} \mid m \geq 1\}$ ako prienik dvoch jazykov L_1 a L_2, ktoré spĺňajú súčasne nasledujúce podmienky:

(i) $L_1 \not\subseteq L_2$, $L_2 \not\subseteq L_1$

(ii) $L_1, L_2 \subseteq \{0\}^+\{1\}^+\{2\}^+$

(iii) $L_1 - L$ a $L_2 - L$ sú nekonečné jazyky.

Odôvodnite.

Riešenie 1. príkladu

(a) Hľadáme jazyky, ktoré sú rôzne, ale ich druhé mocniny sú rovnaké. Môžeme skúšať hľadať takéto konečné jazyky, uvažovať, že potrebujeme nejaké slovo vyskladať iným spôsobom, ako vznikne v druhom jazyku. Mohli by ste navrhnúť napr. takúto dvojicu jazykov: $A = \{a, aaa\}, B = \{\lambda, aa, aaaa\}$.

Pozrime sa na ich druhé mocniny. $A^2 = \{aa, aaaa, aaaaaa\} = \{a^2, a^4, a^6\}$. $B^2 = \{\lambda, aa, aaaa, aaaaaa\} = \{\lambda, a^2, a^4, a^6\}$. Vidíme, že sú takmer rovnaké, presnejšie $A^2 \subset B^2$, ale nie sú rovnaké, lebo v B^2 je aj prázdne slovo λ. Je častou chybou, že si neuvedomíte, že $\lambda.\lambda = \lambda$.

Z toho vyplýva, že nevieme mať dva jazyky, v ktorých v jednom prázdne slovo je a v druhom nie je. Keby sme dali do oboch prázdne slovo, tak v ich druhých mocninách by bol aj jazyk samotný (jazyk zreťazený s prázdnym slovom), čo by nám to opäť mohlo pokaziť.

Pri konečných jazykoch bude ich druhá mocnina vždy obsahovať viac slov[1] (zamyslite sa, prečo). Tak skúsme nájsť nejaké nekonečné jazyky.

Nech $A = \{\lambda, a, aa...\} = \{a^k \mid k \in N\}$.

$B = \{\lambda, a, aaa, aaaa...\} = \{a^k \mid k \in N, k \neq 2\}$

Ich druhé mocniny budú rovné $A^2 = B^2 = A = \{a^k \mid k \in N\}$, keďže slovo aa v B^2 vznikne zreťazením $a.a$. Vidíme, že tieto jazyky rovnaké nie sú, ale ich druhé mocniny áno.

[1] okrem jazyka $\{\lambda\}$

(b) Jazyk L je nekonečný. Keďže $L_1 - L$, $L_2 - L$ sú nekonečné jazyky, tak aj L_1 a L_2 budú nekonečné. Aby sme splnili podmienku (ii) a zároveň, aby sme jednoznačne vedeli povedať, že $L_1 \cap L_2 = L$, použime jazyk L s miernymi úpravami pre oba jazyky.

Nech: $L_1 = \{0^{m+k}\ 1^{2m}\ 2^{m+3} \mid m, k \geq 1\}$
$L_2 = \{0^{m+4}\ 1^{lm}\ 2^{m+3} \mid m, l \geq 1\}$.

Čo sme zmenili? V oboch jazykoch pribudol jeden parameter navyše. Čo to znamená? Skúsme si obaja jazyky napísať ako zjednotenie jazykov postupnou iteráciou týchto parametrov.

Čiže: $L_1 = \{0^{m+1}\ 1^{2m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+2}\ 1^{2m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+3}\ 1^{2m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+4}\ 1^{2m}\ 2^{m+3} \mid m \geq 1\}$...

$L_2 = \{0^{m+4}\ 1^{m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+4}\ 1^{2m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+4}\ 1^{3m}\ 2^{m+3} \mid m \geq 1\} \cup \{0^{m+4}\ 1^{4m}\ 2^{m+3} \mid m \geq 1\}$...

Vidíme, že jazyk L je podmnožinou oboch týchto jazykov. Je aj ich prienikom? V jazyku L_1 budú rovnaké slová ako v L_2 iba ak v jazyku L_1 bude $k = 4$ a zároveň v jazyku L_2 bude $l = 2$. Pre iné hodnoty k, l slová z jedného jazyka nebudú patriť do druhého (a opačne).

Ako budú vyzerať $L_1 - L$ a $L_2 - L$? Pozrime sa vyššie. Vidíme, že ak z oboch jazykov vynecháme jazyk L, parametre k, l nemôžu nadobudnúť hodnoty 4 resp. 2. To znamená, že:

$L_1 - L = \{0^{m+k}\ 1^{2m}\ 2^{m+3} \mid m, k \geq 1\, k \neq 4\}$.
$L_2 - L = \{0^{m+4}\ 1^{lm}\ 2^{m+3} \mid m, l \geq 1, l \neq 2\}$.

Oba jazyky sú nekonečné (podmienka (iii)), keďže množina celých kladných čísel okrem 4 (resp. okrem 2) je nekonečná. Podmienku (ii) sme splnili, v jazyku L_1 by mohlo byť aj $k = 0$, keďže $m \geq 1$, čiže tam vždy bude aspoň jedna nula. V jazyku L_2 nemôže byť $l = 0$, keďže v takom prípade by v slovách neboli žiadne jednotky.

A čo s podmienkou (i)? Pozrime sa na slovo z jazyka L_1, keď $m = 1 \wedge k = 1$. $w = 0^{1+1}\ 1^2\ 2^{1+3}\ = 00112222$. Toto slovo sa určite nebude nachádzať v jazyku L_2. Prečo? Lebo pre $m = 1$ bude v slovách z jazyka L_2 päť núl, dve nuly sa v žiadnom slove z L_2 nachádzať nebudú. To znamená, že $L_1 \nsubseteq L_2$.

Skúsme to aj opačne. Pre $m = 1 \wedge l = 1$ bude slovo $w = 0^{1+4}\ 1^{1.1}\ 2^{1+3}\ =$ 0000012222. Toto slovo nepatrí do jazyka L_1, keďže v ňom budú vždy slová, ktoré majú párny počet jednotiek. Z toho vyplýva, že $L_2 \nsubseteq L_1$.

2. príklad

(a) Pozrime sa na homomorfizmy. Máme abecedy $\Sigma = \{0, 1\}$ a $\Gamma = \{a, b\}$. Homomorfizmus h_1 je definovaný takto:

$$h_1 : \Sigma^* \to \Gamma^*$$
$$h_1(0) = aa$$
$$h_1(1) = a$$

(i) $L_0 = \{0, 01, 001, 111\}$, čomu je rovné $h_1(L_0)$?

(ii) $L_a = \{aa, aaa, baa\}$, čomu je rovné $h_1^{-1}(L_a)$?

(b) Majme homomorfizmus h definovaný takto:

$$h : \Sigma^* \to \Gamma^*$$
$$h(0) = ab$$
$$h(1) = b$$

Zistite čomu je rovné:

(i) $h(001001)$

(ii) $h^{-1}(ababbb)$

(iii) $h^{-1}(baba)$

(iv) $L_1 = \{0, 1, 110, 001\}$
$h(L_1)$

(v) $L_2 = \{ab, bb, bab\}$
$h^{-1}(L_2)$

(vi) $L_3 = \{aaa, aba\}$
$h^{-1}(L_3)$

(vii) $L_4 = \{abb, baa\}$
$h^{-1}(L_4)$

Riešenie 2. príkladu

(a) Zopakujme si, ako fungujú homomorfizmy. Homomorfizmus zobrazuje znaky na slová (preto pre definovaní homomorfizmu musíme povedať, na čo sa zobrazí každý znak z abecedy). Slovo sa v homomorfizme zobrazí na slovo (pre každý vzor existuje iba jeden jeho obraz), jazyk na jazyk. Pri inverznom homomorfizme sa môže stať, že jedno slovo vzniklo z rôznych slov, čiže pre toto slovo bude viac jeho vzorov (jeden obraz mohol vzniknúť z viacerých vzorov). Inverzný homomorfizmus slova preto bude jazyk, a inverzný homomorfizmus jazyka zjednotenie jazykov inverzného homomorfizmu pre všetky slová.

(i) $h_1(L_0) = \{aa, aaa, aaaaa\}$ - slovo aaa vzniklo aj ako $h1(01)$, aj ako $h1(111)$, ale jazyk je množina, preto ho uvádzame iba raz.

(ii) $h_1^{-1}(L_a) = \{0, 11, 01, 10, 111\}$ Prvé slovo mohlo vzniknúť tromi spôsobmi, druhé slovo dvoma, slovo baa vzniknúť nemohlo.

(b) Postupujeme rovnako ako v (a).

(i) $h(001001) = ababbababb$

(ii) $h^{-1}(ababbb) = \{0011\}$

(iii) $h^{-1}(baba) = \{\}$

(iv) $h(L_1) = \{ab, b, bbab, ababb\}$

(v) $h^{-1}(L_2) = \{0, 11, 10\}$

(vi) $h^{-1}(L_3) = \{\}$

(vii) $h^{-1}(L_4) = \{01\}$

3. príklad

Skonštruujme automat, ktorý ponúka tri nápoje (Kávu, Čaj, Mlieko), pričom každý z nich stojí 30 centov. Automat berie 10 centové a 20 centové mince, výdavok nevracia. Tlačidlo pre výber nápoja vieme stlačiť hocikedy, ale nápoj nám automat vydá jedine vtedy, ak stlačíme tlačidlo potom, ako sme už vhodili 30 centov.

	t_0	t_1	t_2	t_3	t_4
STAV	q_0	q_1	q_2	q_3	q_4
VSTUP	10c	10c	10c	K	λ
VÝSTUP	nič	nič	nič	nič	nápoj

Tabuľka 1: Správanie automatu pri vhodení 3x10c mincí a výbere kávy

(a) Nakreslite podľa tabuliek stavový diagram pre automat. Ktoré prechody tam musíme dokresliť, aby bol deterministický?

	t_0	t_1	t_2	t_3
STAV	q_0	q_2	q_3	q_4
VSTUP	20c	10c	Č	λ
VÝSTUP	nič	nič	nič	nápoj

Tabuľka 2: Správanie automatu pri vhodení 20c, 10c a výbere čaju

(b) Definujte automat pomocou pätice z prednášky.

(c) Aká bude počiatočná konfigurácia pre prípady z tabuliek?

(d) Odsimulujte výpočet pre obe konfigurácie.

(e) Odsimulujte výpočet pre nejaký ďalší prípad. Je to akceptujúci alebo zamietajúci výpočet?

(f) Čo by sme museli zmeniť v automate, aby vracal výdavok? Čo by si musel automat pamätať a ako to vieme zabezpečiť?

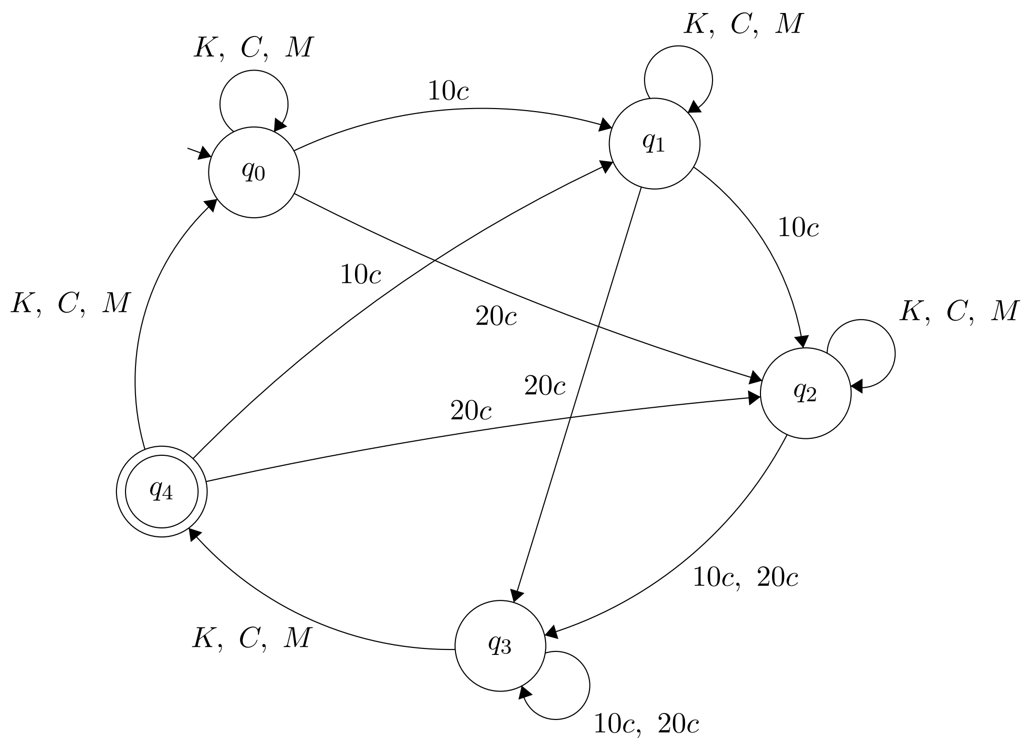
Riešenie 3. príkladu

(a) Pozrime sa najprv na tabuľky. V prvej vidíme, že sme začali v stave q_0, vhodili 10c, automat ešte nič nevrátil, ale posunul sa do stavu q_1. Tam sme opäť vhodili 10c, posunuli sme sa do stavu q_2. Ešte raz sme vhodili 10c, posunuli sa do stavu q_3. Tam sme stačili tlačidlo Káva, automat sa posunul do stavu q_4 a vydal nám nápoj. V druhej tabuľke sme preskočili stav q_1, lebo sme rovno hodili 20c. Keď sa nad tým zamyslíme, zistíme, čo znamenajú jednotlivé stavy.

Stav q_0 je vstupný (začiatočný), v ktorom ešte automat neeviduje žiadne vhodené mince. V stavoch q_1, q_2, q_3 sa nachádza, ak sme doň už hodili 10, 20 alebo 30 centov. Nápoj vydáva v stave q_4, ak sme už hodili 30 centov (a teda boli v stave q_3) a vybrali si nápoj. Ak tlačidlo pre ľubovoľný nápoj stlačíme v inom stave, automat sa neposunie do iného stavu, keďže suma vhodených peňazí sa nezmení. Čo ak hodíme viac ako 30 centov? Podľa zadania automat čaká, kým stlačíme tlačidlo s voľbou nápoja, čiže hocijakú sumu nad 30 centov budeme brať ako 30 centov a stav automatu sa meniť nebude.

Čo sa stane, keď už vydáme nápoj a stlačíme opäť nejaké iné tlačidlo? Ak by sa stav automatu nezmenil, stačilo by raz vhodiť 30 centov a mali by sme nekonečné nápoje[2], preto sa musíme niekam posunúť.

[2]čomu by sa asi prevádzkovateľ automatu nepotešil



Obr. 1: Riešenie príkladu 3

Keďže aktuálny stav vhodených mincí je 0 centov, posunieme sa do q_0. Čo ale ak sme v stave q_4, automat nám vydal nápoj a my vhodíme 10 centov? Mal by to zaevidovať a teda nás posunúť do stavu q_1. Rovnako tak, ak vhodíme 20 centov, tak nás posunie do q_2.

Výsledný automat vidíte na obrázku 1.

(b) Definujeme: $\Sigma = \{10c, 20c, K, C, M\}$, $Q = \{q_0, q_1, q_2, q_3, q_4\}$, počiatočný stav je q_0, akceptačný stav je iba jeden, preto $F = \{q_4\}$ a prechodovú funkciu si môžeme skúsiť zapísať aj do tabuľky.

(c) Počiatočná konfigurácia pre prvú tabuľku bude $(q_0, 10c.10c.10c.K)$ a pre druhú tabuľku $(q_0, 20c.10c.C)$.

(d) Simuláciu výpočtu môžeme spraviť dvomi spôsobmi, buď použijeme prechodovú funkciu, alebo reflexívno tranzitívny uzáver relácie krok stroja. Pre každú konfiguráciu použijem iný, aby sme si ukázali oba.

δ	$10c$	$20c$	K	C	M
q_0	q_1	q_2	q_0	q_0	q_0
q_1	q_2	q_3	q_1	q_1	q_1
q_2	q_3	q_3	q_2	q_2	q_2
q_3	q_3	q_3	q_4	q_4	q_4
q_4	q_1	q_2	q_0	q_0	q_0

Tabuľka 3: Prechodová funkcia

$\hat{\delta}(q_0, 10c.10c.10c.K) = \hat{\delta}(q_1, 10c.10c.K) = \hat{\delta}(q_2, 10c.K) = \hat{\delta}(q_3, K) = \delta(q_4, \lambda)$ Skončili sme v stave q_4, preto je tento výpočet akceptujúci.

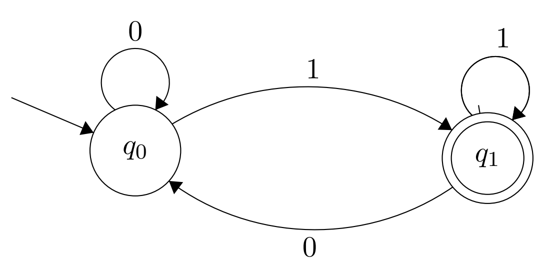
$(q_0, 20c.10c.C) \vdash_M (q_2, 10c.C) \vdash_M= (q_3, C) \vdash_M (q_4, \lambda)$. Opäť sme skončili v stave q_4, čiže tento výpočet je akceptujúci.

(e) Skúsme vymyslieť zamietajúci výpočet a ten odsimulovať. Čo ak si vypýtame nápoj skôr, ako hodíme dosť peňazí? Napr. $(q_0, 20c.M) \vdash_M (q_2, M) \vdash_M (q_2, \lambda)$. Ďalším zamietajúcim výpočtom bude aj situácia, ak tam vhodíme napr. 50 centov, ale nestlačíme žiadne tlačidlo. Napr. $(q_0, 20c.10c.10c.10c) \vdash_M (q_2, 10c.10c.10c) \vdash_M= (q_3, 10c.10c) \vdash_M (q_3, 10c) \vdash_M (q_3, \lambda)$.

(f) Nad touto úlohou sa zamyslite. Koľko stavov by sme museli pridať, aby si automat vedel zapamätať, že má vydať 10, 20, 30... centov? Kedy by automat mince vydával?

4. príklad

Pre automat na obrázku 2 spravte formálny zápis (pätica z prednášky) a spravte výpočet na slovách 110101 a 0010110.



Obr. 2: Automat zo 4. príkladu

(a) Aké slová automat akceptuje?

(b) Keby sme vymenili akceptačný a vstupný stav, aké slová by akceptoval?

(c) Navrhnite automat (DKA), ktorý akceptuje všetky slová nad abecedou Σ_{bool}, ktoré končia 10, a neakceptuje žiadne iné.

(d) Vieme vytvoriť automat, ktorý bude akceptovať iba slová končiace 1^k, (pre nami zvolené $k \in N$)? Koľko stavov bude takýto automat mať (v závislosti od k)? Ako sa vytvorí výsledný automat pre zadané k?

Riešenie 4. príkladu

Formálny zápis automatu je: $M = \{Q, \Sigma, \delta, q_0, F\}$, kde $Q = \{q_0, q_1\}$, $\Sigma = \{0, 1\}$, $F = \{q_1\}$ a $\{\delta(q_0, 0) = q_0; \delta(q_0, 1) = q_1; \delta(q_1, 0) = q_0; \delta(q_1, 1) = q_1\}$.

Spravme výpočet na slove 110101. $\hat{\delta}(q_0, 110101) = \hat{\delta}(q_1, 10101) = \hat{\delta}(q_1, 0101) = \hat{\delta}(q_0, 101) = \hat{\delta}(q_1, 01) = \hat{\delta}(q_0, 1) = \hat{\delta}(q_1, \lambda)$.

Pre slovo 0010110 bez všetkých krokov: $\hat{\delta}(q_0, 0010110) = \hat{\delta}(q_0, \lambda)$.

Prvý výpočet je akceptačný, druhý je zamietajúci.

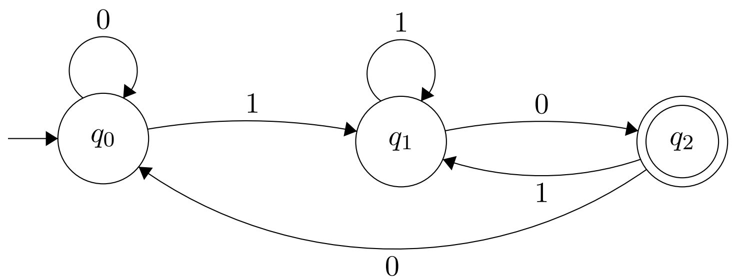
(a) Aj z dvoch výpočtov vyššie vidíme, že automat akceptuje slová, ktoré končia 1, čo vieme zapísať ako $L(M) = \{w1 \mid w \in \Sigma^*\}$.

(b) Ak by sme vymenili akceptačný a vstupný stav, automat by akceptoval slová končiace na 0.

(c) Automat bude musieť mať aspoň tri stavy, aby sme vedeli zaručiť, že v slove je určite 1 a po nej nasleduje 0. Na obrázku 4 vidíme, že v stave q_0 môže byť ľubovoľne veľa 0, lebo čakáme na 1. V stave q_1 sme už prečítali jednu jednotku, preto za ňou môže nasledovať ľubovoľne veľa ďalších jednotiek, a do akceptačného stavu sa presunieme iba ak príde 0.

Čo ak by bolo slovo 1010? Mali by sme ho akceptovať, preto ak po 10 príde jednotka, presunieme sa do stavu q_1 (čiže tam budú končiť slová končiace jednotkou) a po nule sa zas presunieme do q_1.

Čo v prípade slov, ktoré obsahujú 10, ale nie ako sufix? Napr. $1011, 11101, 100000$? Ani jedno z týchto slov nemôže skončiť v stave q_2. Prvé dve slová skončia v stave q_1, keďže končia jednotkou, tretie slovo by malo skončiť v stave q_0. Prečo? Ak by sme k nemu prireťazili 10, bolo by akceptované, pričom k prvým dvom slovám stačí prireťaziť iba 0. Takéto slová preto nemôžu skončiť v rovnakom stave.



Obr. 3: Riešenie príkladu 4c

(d) Zamyslite sa, koľko stavov (a prechodov) by ste museli vytvoriť v takomto automate? Viete použiť automat z (a) a nejako ho upraviť?

5. príklad*

Nakreslite automat $M = \{Q, \Sigma, \delta, q_0, F\}$, ak poznáte $F = \{q_0\}$ a δ (tabuľka 3).

(a) Viete zo zadaných informácií zistiť Σ, Q?

(b) Aké slová tento automat rozoznáva?

δ	q_0	q_1	q_2
a	q_1	q_2	q_0
b	q_0	q_1	q_2

Tabuľka 4: Prechodová funkcia δ

6. príklad*

Nakreslite a zadefinujte deterministický konečný automat rozpoznávajúci jazyk $L = \{ab, bba, bab\}$, $L \subseteq \{a, b\}^*$. Koľko stavov musí mať minimálne? Akceptuje iba slová z jazyka L? Vieme pre každé slovo nad vstupnou abecedou povedať, či ho náš automat akceptuje alebo nie?