Teoretická informatika (TIN) – 2024/2025
Úkol 1
(max. zisk 7 bodů – 10 bodů níže odpovídá 1 bodu v hodnocení předmětu)

1. Uvažujme abecedu $\Sigma = \{a, b\}$ a jazyk $L_1 = \{w \in \Sigma^* : |w| \bmod 2 = 1\}$. Sestrojte relaci pravé kongruence $\sim$, která splňuje následující tři podmínky: (1) L_1 je sjednocením některých tříd rozkladu $\Sigma^*/_{\sim}$, (2) index $\sim$ je konečný a soudělný s indexem $\sim_{L_1}$ a (3) jedna ze tříd rozkladu $\Sigma^*/_{\sim}$ má právě dva prvky.

15 bodů

2. Uvažte následující operaci na jazycích nad abecedou Σ:

$$\Box\, L = \{w \in L \mid \forall u, v \in \Sigma^* \colon w = uv \Rightarrow u \in L\},$$

Rozhodněte a dokažte, zda jsou následující třídy jazyků uzavřeny na operaci $\Box$:

   (a) třída regulárních jazyků a

   (b) třída rekurzivně vyčíslitelných jazyků.

15 bodů

3. Uvažujte následující jazyk nad abecedou $\Sigma = \{a, b\}$:

$$L_3 = \{a^k b^\ell \mid \ell = k^2\}$$

Dokažte, že jazyk L_3 není bezkontextový.

15 bodů

4. Navrhněte algoritmus, který pro bezkontextovou gramatiku $G = (N, \Sigma, P, S)$ spočítá množinu

$$N_{abc} = \{A \in N \mid \exists u, v \in \Sigma^* \colon A \Rightarrow_G^* uabcv\}.$$

V algoritmu můžete využít množiny $N_\epsilon = \{A \in N \mid A \Rightarrow_G^+ \epsilon\}$ a $N_t = \{A \in N \mid \exists w \in \Sigma^* \colon A \Rightarrow_G^+ w\}$. Doporučujeme nadefinovat si další vhodné množiny neterminálů a algoritmicky popsat jejich výpočet (u N_ϵ a N_t popis výpočtu není potřeba).

Ilustrujte použití algoritmu na příkladě gramatiky nad abecedou $\Sigma = \{a, b, c, d\}$ s pravidly

$$S \to V \mid caUca \mid RTcU \qquad V \to Vabc \qquad U \to b \mid \epsilon \qquad R \to Uca \qquad T \to W \mid aU \qquad W \to UbU$$

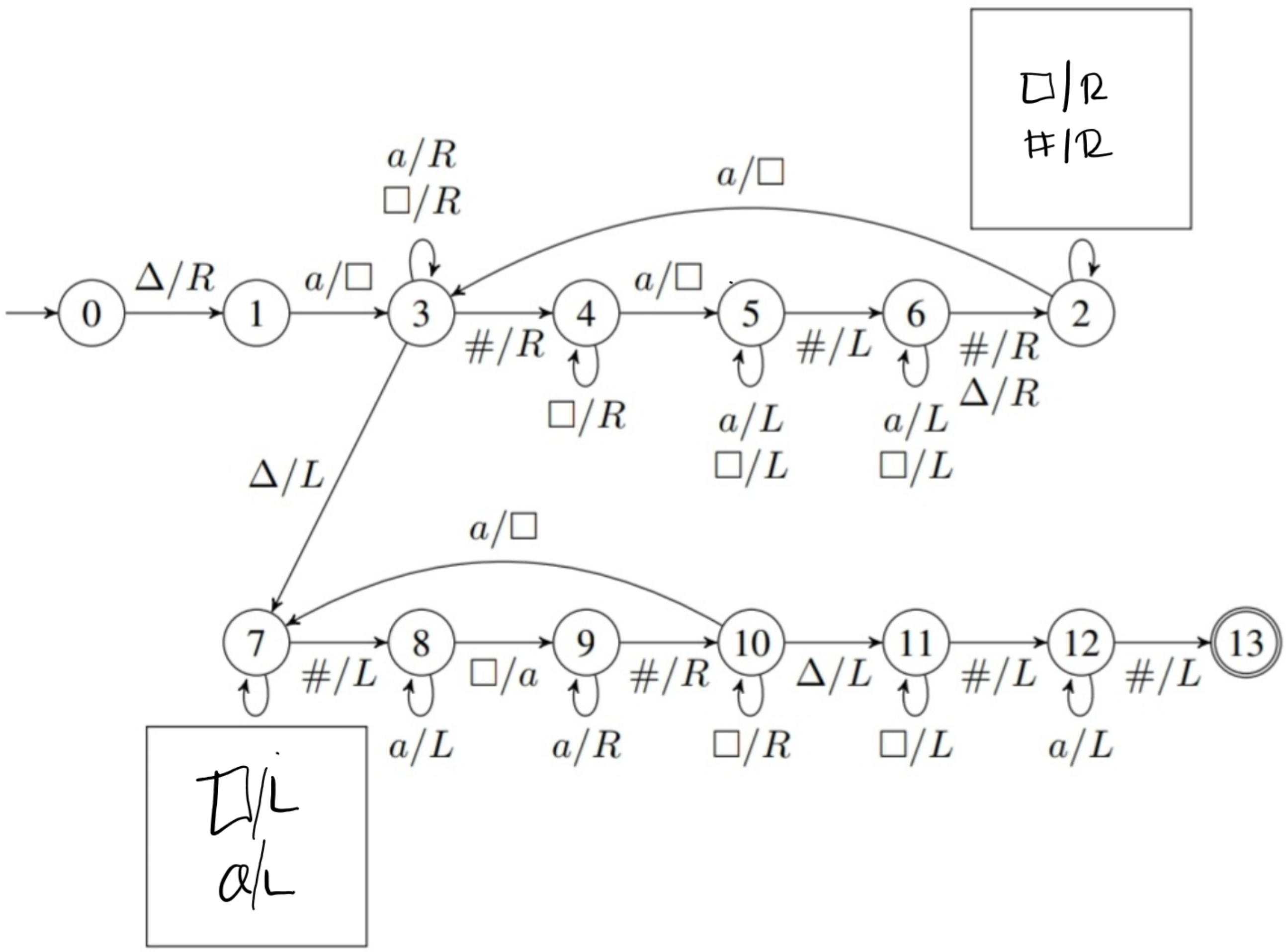
15 bodů

5. Doplňte do rámečků v přechodovém diagramu Turingova stroje M_5 s páskovou abecedou $\Gamma = \{a, \#, \Box, \Delta\}$ v Obrázku 1 chybějící popisky přechodů tak, aby platilo, že $L(M_5) = \{a^{\ell_1} \# a^{\ell_2} \# a^{\ell_3} \mid 1 \leq \ell_1 \leq \ell_2 \leq \ell_3 \wedge \ell_2 - \ell_1 = \ell_3 - \ell_2\}$. V jednom popisku může být i více operací. Nic jiného nepřidávejte.

10 bodů

5. Doplňte do rámečků v přechodovém diagramu Turingova stroje M_5 s páskovou abecedou $\Gamma = \{a, \#, \Box, \Delta\}$ v Obrázku 1 chybějící popisky přechodů tak, aby platilo, že $L(M_5) = \{a^{\ell_1}\#a^{\ell_2}\#a^{\ell_3} \mid 1 \leq \ell_1 \leq \ell_2 \leq \ell_3 \wedge \ell_2 - \ell_1 = \ell_3 - \ell_2\}$. V jednom popisku může být i více operací. Nic jiného nepřidávejte.

10 bodů

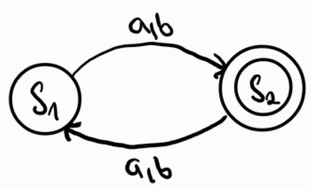


Obrázek 1: Přechodový diagram Turingova stroje M_5

1. Uvažujme abecedu $\Sigma = \{a, b\}$ a jazyk $L_1 = \{w \in \Sigma^* : |w| \bmod 2 = 1\}$. Sestrojte relaci pravé kongruence $\sim$, která splňuje následující tři podmínky: (1) L_1 je sjednocením některých tříd rozkladu $\Sigma^*_{/\sim}$, (2) index $\sim$ je konečný a soudělný s indexem $\sim_{L_1}$ a (3) jedna ze tříd rozkladu $\Sigma^*_{/\sim}$ má právě dva prvky.

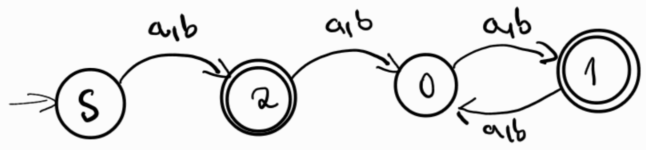
15 bodů

Minimální konečný automat jazyka L_1:



(minimalizace je níže)

A_2:



$L(S) = \emptyset$

$L(2) = \{a, b\}$

aaa

$L(0) = L((aaa + aab + aba + baa + abb + bba + bbb)^*)$

$L(1) = L\{ww \mid w \in \{aa, ab, ba, bb\}\}^*$

Relace:

$\forall u,v : u \sim v = ((u \in L((aaa + aab + aba + baa + abb + bba + bbb)^*) \wedge$

$v \in L((aaa + aab + aba + baa + abb + bba + bbb)^*)) \vee$

$(u \in \{ww \mid w \in \{aa, ab, ba, bb\}\}^* \wedge v \in \{ww \mid w \in \{aa, ab, ba, bb\}\}^*) \vee$

$(u \in \{a, b\} \wedge v \in \{a, b\}) \vee (u = \varepsilon \wedge v = \varepsilon))$

(1) Platí: $L_1 = L(1) \cup L(2)$.

(2) Index $\sim_L$ je právě 2 což lze vidět z minimálního konečného automatu). Index vytvořené relace je právě 4 což je soudělné s 2.

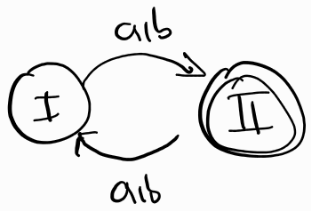
(3) Třída daná jazykem $L(2)$ má dva prvky

Minimalizace KA:

	a	b
S_1	S_2	S_2
S_2	S_1	S_1

	$\equiv_0$	a	b
I	1	II	II
II	2	I	I

$\Rightarrow$



$\Rightarrow$ automat je minimální

2. Uvažte následující operaci na jazycích nad abecedou Σ:

$$\Box L = \{w \in L \mid \forall u, v \in \Sigma^* : w = uv \Rightarrow u \in L\},$$

Rozhodněte a dokažte, zda jsou následující třídy jazyků uzavřeny na operaci $\Box$:

(a) třída regulárních jazyků a ANO

(b) třída rekurzivně vyčíslitelných jazyků.

(a) Pro realizaci takové operace je možné pouze modifikace KA ⇒ Operace je uzavřena

Zdůvodnění

KA lze modifikovat následovně:

(a) přidání stavu Sink a přechodů které dosud nikam nevedli

(b) Všechny koneční stavy zůstanou konečné.

(c) Všechny nekončující stavy jsou spojeny do jednoho "sink" stavu a jsou do něj nasměrovány všechny přechody co dosud nikam nevedou

(d) neexistuje přechod ze sink stavu jinde než do sink stavu

(e) jsou odstraněny nedosažitelné stavy

lze sestrojit KA pro operaci $\Box$ pro libovolný jazyk $L \in \mathcal{L}_3$

$\Box$ je uzavřena pro $\mathcal{L}_3$.

(b) Operace je uzavřena i nad $\mathcal{L}_{RE}$

(a) TS musí ověřit zda každý prefix slova se nachází v $\Box L$

(a) každé slovo má konečný počet prefixů (každé slovo je konečné)

(b) u každého prefixu dokáže TS rozhodnout zda náleží do jazyka L. Je dostatečná odpověď ano.

(c) je možné vykonat operaci $\Box$ pomocí RE TS.

$\Box$ je uzavřeno nad $\mathcal{L}_{RE}$

3. Uvažujte následující jazyk nad abecedou $\Sigma = \{a, b\}$:

$$L_3 = \{a^k b^\ell \mid \ell = k^2\}$$

Dokažte, že jazyk L_3 není bezkontextový. NENÍ BKJ.

aaaaa bbbbb

V	W	X

① $\forall k > 0$ si zvolím slovo $z = a^k b^{k^2}$. Zjevně platí že $z \in L_3 \wedge |z| > k$

② Uvažujeme libovolné rozdělení $z = uvwxy$ kde $u, v, w, x, y \in \Sigma^*$ kde $vx \neq \varepsilon \wedge |vwx| \leq K$

ⓐ $u = a^{d_1}$ $v = a^{d_2}$ $w = a^{d_3}$ $x = a^{d_4}$ $y = a^{d_5} b^{k^2}$

$d_1 + d_2 + d_3 + d_4 + d_5 = k$

pro $i=0$ platí že slovo w_i $(w_i = uv^iwx^iy)$

$\#_a(w_i) = d_1 + d_3 + d_5 = K - (d_2 + d_4)$ což pro nenulové $d_2 + d_4$ zcela jistě není k

$\#_b(w_i) = k^2 \neq (k - (d_2 + d_4))^2$

w_i

ⓑ $u = a^{d_1}$ $v = a^{d_2} b^{\beta_1}$ $wxy = b^{\beta_2}$: $d_1 + d_2 = k \wedge \beta_1 + \beta_2 = k^2$

a zároveň

$uvw = a^{d_1}$ $x = a^{d_2} b^{\beta_1}$ $y = b^{\beta_2}$: $d_1 + d_2 = K \wedge \beta_1 + \beta_2 = k^2$

slovo $w_i = uv^iwx^iy$ dokážu najít $i = 2$, bude porušeno pořadí a

ⓒ V obsahuje pouze "a" $\wedge$ X obsahuje pouze "b"

$u = a^{d_1}$ $v = a^{d_2}$ $w = a^{d_3} b^{\beta_1}$ $x = b^{\beta_2}$ $y = b^{\beta_3}$

---

pro $i = 2$ slovo $w_{i2} = uv^2wx^2y$

$i = 3$ slovo w_{i3}

$\#_a(w_{i2}) = k + d_2$

$\#_b(w_{i3}) = k^2 + \beta_2$

$\#_a(w_{i3}) = k + 2d_2$

$\#_b(w_{i3}) = k^2 + 2\beta_2$

$(k + d_2)^2 = k^2 + \beta_2$

$(k + 2d_2)^2 = k^2 + 2\beta_2$

(Vyřešil jsem pomocí wolframu)

$d_2 = 0 \wedge \beta_2 = 0$

což je v rozporu s PL.

Pro všechny případy lze najít i takové že slovo $w_i \notin L_3$ tímpádem $L_3 \notin \mathcal{L}_3$

4. Navrhněte algoritmus, který pro bezkontextovou gramatiku $G = (N, \Sigma, P, S)$ spočítá množinu

$$N_{abc} = \{A \in N \mid \exists u, v \in \Sigma^* : A \Rightarrow^*_G uabcv\}.$$

V algoritmu můžete využít množiny $N_\epsilon = \{A \in N \mid A \Rightarrow^+_G \epsilon\}$ a $N_t = \{A \in N \mid \exists w \in \Sigma^* : A \Rightarrow^+_G w\}$. Doporučujeme nadefinovat si další vhodné množiny neterminálů a algoritmicky popsat jejich výpočet (u N_ϵ a N_t popis výpočtu není potřeba).

Ilustrujte použití algoritmu na příkladě gramatiky nad abecedou $\Sigma = \{a, b, c, d\}$ s pravidly

$S \to V \mid caUca \mid RTcU$ $\quad V \to Vabc$ $\quad U \to b \mid \epsilon$ $\quad R \to Uca$ $\quad T \to W \mid aU$ $\quad W \to UbU$

Algoritmus pro Nabc

Vstup: $G=(N,\Sigma,P,S)$ Výstup N_{abc}

1) $N^0_{abc} = \emptyset$, $i=0$
2) $N^{i+1}_{abc} = \{A \mid A \to B_1 \alpha_{ab} B_2 \alpha_c B_3 : B_1, B_3 \in (N_\epsilon \cup N_T)^* \wedge B_2 \in N_\epsilon^* \wedge \alpha_{ab} \in (N_{ab*} \cup \{ab\}) \wedge \alpha_c \in (N_{c*} \cup \{c\})\} \cup$

   $\{A \mid A \to B_1 \alpha_a B_2 \alpha_{bc} B_3 : B_1, B_3 \in (N_\epsilon \cup N_T)^* \wedge B_2 \in N_\epsilon^* \wedge \alpha_a \in (N_{a*} \cup \{a\}) \wedge \alpha_{bc} \in (N_{bc*} \cup \{bc\})\} \cup$

   $\{A \mid A \to B_1 \alpha_a B_2 \alpha_b B_3 \alpha_c : B_1, B_3 \in (N_\epsilon \cup N_T)^* \wedge B_2 \in N_\epsilon^* \wedge \alpha_a \in (N_{*a} \cup \{a\}) \wedge \alpha_b \in (N_b^* \cup \{b\}) \wedge \alpha_c \in (N_c^* \cup \{c\})$

   $\cup N^i_{abc}$
3) Pokud je N^{i+1}_{abc} rovno N^i_{abc} je N^i_{abc} vráceno jako N_{abc} jinak inkrementuj i a (2)

Algoritmus pro N_α : kde $\alpha \in \Sigma$, $N_\alpha = \{A \Rightarrow^*_G \alpha\}$ - obecný popis pro libovolný symbol nad Σ

Vstup: $G=(N,\Sigma,P,S)$ Výstup N_α

1) $N^0_\alpha = \emptyset$, $i=0$
2) $N^{i+1}_\alpha = \{A \mid A \to B_1 w B_2 : B_1 \in N_\epsilon^* \wedge B_2 \in N_\epsilon^* \wedge w \in (N_\alpha \cup \{\alpha\})\} \cup N^i_\alpha$
3) Pokud se N^i_α rovná N^{i+1}_α tak vrať N^i_α jinak zhaov bod (2) a inkrementuj i

(Další algoritmy jsou na další straně)

Ilustrujte použití algoritmu na příkladě gramatiky nad abecedou $\Sigma = \{a, b, c, d\}$ s pravidly

$S \to V \mid caUca \mid RTcU$ $\quad V \to Vabc$ $\quad U \to b \mid \epsilon$ $\quad R \to Uca$ $\quad T \to W \mid aU$ $\quad W \to UbU$

Množiny definované zadáním

$N_t = \{U, R, W, T, S\} = N \setminus \{V\}$

$N_\epsilon = \{U\}$

1)	$N^0_a = \emptyset$	$N^0_b = \emptyset$	$N^0_c = \emptyset$	$N^0_{*a} = \emptyset$	$N^0_{c*} = \emptyset$
2)	$N^1_a = \{T\}$	$N^1_b = \{b\}$	$N^1_c = \emptyset = N_c$	$N^1_{*a} = \{T\}$	$N^1_{c*} = \{R\}$
3)	$N^2_a = \{T\} = N_a$	$N^2_b = \{b\} = N_b$		$N_{*a} = \{T\} = N_{*a}$	$N^2_{c*} = \{R, S\}$
					$N^3_{c*} = \{R, S\}$

1)	$N^0_{*ab} = \emptyset$	$N^0_{bc*} = \emptyset$	$N^0_{abc} = \emptyset$
2)	$N_{*ab} = \{T\}$	$N^1_{bc*} = \{R\}$	$N^1_{abc} = \{S\}$
3)	$N_{*ab} = \{T\} = N_{*ab}$	$N^2_{bc*} = \{R\} = N_{bc*}$	$N^2_{abc} = \{S\} = N_{abc}$

Algoritmus pro N_{c*}, $N_1 = \{A \Rightarrow_G^* cw : w \in \Sigma^*\}$

Vstup: $G = (N, \Sigma, P, S)$ Výstup N_{c*}

1) $N_c^0 := \emptyset$, $i = 0$
2) $N_c^{i+1} = \{A \mid A \to B_1 w B_2 : B_1 \in N_\varepsilon^* \wedge B_2 \in (N_e \cup N_T \cup \Sigma)^* w \in (N_c \cup \{c\} \cup N_{c*}^i)$
3) Pokud se N_a^i rovná N_{c*}^{i+1} tak vrať N_{c*}^i jinak znovu bod 2 a inkrementuj i

Algoritmus pro N_{*a}:, $N_{*a} = \{A \Rightarrow_G^* wa : w \in \Sigma^*\}$

Vstup: $G = (N, \Sigma, P, S)$ Výstup N_{*a}

1) $N_\alpha^0 := \emptyset$, $i = 0$
2) $N_\alpha^{i+1} = \{A \mid A \to B_1 w B_2 : B_2 \in N_\varepsilon^* \wedge B_1 \in (N_e \cup N_t \cup \Sigma)^* w \in (N_a \cup \{a\} \cup N_{*a}^i)$
3) Pokud se N_a^i rovná N_{c*}^{i+1} tak vrať N_{c*}^i jinak znovu bod 2 a inkrementuj i

Algoritmus pro N_{*ab}:, $N_{*a} = \{A \Rightarrow_G^* wab : w \in \Sigma^*\}$

Vstup: $G = (N, \Sigma, P, S)$ Výstup N_{*ab}

1) $N_\alpha^0 := \emptyset$, $i = 0$
2) $N_\alpha^{i+1} = \{A \mid A \to B_1 \alpha_1 w \alpha_2 B_2 : B_2 \in N_\varepsilon^* \wedge B_1 \in (N_e \cup N_t \cup \Sigma)^* \wedge \alpha_1 \in (N_{*a} \cup \{a\}) \wedge \alpha_2 \in (N_b \cup \{b\})$
3) Pokud se N_a^i rovná N_{*ab}^{i+1} tak vrať N_{*ab}^i jinak znovu bod 2 a inkrementuj i

Algoritmus pro N_{bc*}, $N_{*a} = \{A \Rightarrow_G^* bcw : w \in \Sigma^*\}$

Vstup: $G = (N, \Sigma, P, S)$ Výstup N_{bc*}

1) $N_\alpha^0 := \emptyset$, $i = 0$
2) $N_\alpha^{i+1} = \{A \mid A \to B_1 \alpha_1 w \alpha_2 B_2 : B_2 \in N_\varepsilon^* \wedge B_1 \in (N_e \cup N_t \cup \Sigma)^* \wedge \alpha_1 \in (N_b \cup \{a\}) \wedge \alpha_2 \in (N_{c*} \cup \{c\})$
3) Pokud se N_{bc*}^i rovná N_{*ab}^{i+1} tak vrať N_{bc*}^i jinak znovu bod 2 a inkrementuj i