КОНТРОЛНО 1

(82) дефинирайте понятията:

КДА A, изпълнение на дума над A, разширена ф-я на преходите на A, език $L(A)$, регулярен израз r, език $L(r)$

НДКА N, изпълнение на дума над N, разширена ф-я на преходите на N, език $L(N)$

КДА A е $A = \langle \Sigma, Q, q_{start}, \delta, F \rangle$ където

$\Sigma \to$ крайна азбука $Q \to$ множество от състояния
$q_0 \to$ начално състояние $F \to$ финални състояния
$\delta : Q \times \Sigma \to Q$ функция на преходите

изпълнение на дума над A:

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \xrightarrow{a_2} \dots \xrightarrow{a_{n-2}} q_{n-1} \xrightarrow{a_{n-1}} q_n \quad (q_{n-1}, a_{n-1})$$

q_{start} $\delta(q_0, a_0)$ $\delta(q_1, a_1)$ q_F

разширена ф-я на преходите на A $\delta^*(q, \alpha\beta) = \delta^*(\delta^*(q, \alpha), \beta)$

$$\delta^* : Q \times \Sigma^* \to Q$$

език $L(A) \to$ език на автомата A

$$L(A) = \{\alpha \in \Sigma^* \mid A \text{ приема } \alpha\}$$

един език L е автоматен ако $\exists$ ДКА A т.ч. $L = L(A)$

или $L(A) = \{\alpha \in \Sigma^* \mid \delta^*(q_{start}, \alpha) \in F\}$

рег. израз r над азбуката Σ е стринг над азбуката $\Sigma \cup \{\phi, \cdot, \cup, *\}$ който може да се дефинира индуктивно както следва:

1. ϕ и всеки елемент от Σ е рег. израз
2. Ако $\overset{r_1}{\alpha}$ и $\overset{r_2}{\beta}$ са рег. изрази $\alpha \cdot \beta$, $\alpha \cup \beta$, α^* са рег. изр
3. Нищо друго не е рег израз освен ако не следва от първите 2 условия.

рег. език $L(r)$ за рег. израз r. L е фун-ята която описва връзката между рег. израз и езика, който той задава.
L е дефиниран индуктивно както следва:
$L(\phi) = \phi$ $L(r) = \{r\}$ за $\forall$ $r \in \Sigma$ (дори за $r = \varepsilon$)

Ако r_1 и r_2 са рег. изрази то $L(r_1 \cdot r_2) = L(r_1) \cdot L(r_2)$
$L(r_1 \cup r_2) = L(r_1) \cup L(r_2)$, $L(r^*) = (L(r))^*$

НДКА N наричаме петорката $N = \langle Q, \Sigma, \Delta, Q_{start}, F \rangle$

където Q е крайно множество от състояния
Σ е крайна азбука
Q_{start} крайно множество от начални състояния
F множество от финални състояния
$\Delta \to Q \times \Sigma \to P(Q)$ функция на преходи

изпълнение на дума над N $\alpha = a_1 a_2 \dots a_s$

$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \dots \xrightarrow{a_s} q_s (\in F)$ ако е успешно.

$q_s \in Q_s$ $\Delta(q_0, a_1)$

$\Delta(q_{s-1}, a_s)$

разширена ф-я на преходите $\Delta^*: P(Q) \times \Sigma^* \to P(Q)$ за произволни $R \in Q$ и $\alpha \in \Sigma^*$
$\Delta^*(q, \alpha\beta) = \Delta^*(\Delta^*(q, \alpha), \beta)$.

$L(N)$ езика на НДКА $N = \{\omega \mid \omega \in \Sigma^* \ \omega$ се приема от $N\}$

83) формулирайте Лемата за покачване за рег. езици

за $\forall$ рег. език L, $\exists!\ p>0$ такова че за $\forall$ дума $\alpha \in L$ с дължина $|\alpha| \geq p$ $\exists$ разбиване на думата α на 3 части $\alpha = xyz$ където $|xy| \leq p$

$|y| \geq 1$

$\alpha_i = xy^iz \in L$

84) контролно 2
дефинирайте понятията:

Неограничена граматика, едностъпков и многостъпков извод, език на граматика, рег. граматика, безконтекстна граматика, стеков автомат, конфигурация на стеков автомат, едностъпков преход

Неограничена граматика - наричаме наредената четворка $G = \langle V, \Sigma, R, S \rangle$ където V е крайно множество от променливи / нетерминали, $\Sigma \rightarrow$ крайно множество от букви / терминали $S \in V$ начален нетерминал, $R \subseteq (V \cup \Sigma)^+ \times (V \cup \Sigma)^*$ крайно множество от правила за извод.

едностъпков извод $\{\alpha \underset{G}{\rightarrow} \beta,\ \lambda, \rho \in (V \cup \Sigma)^*\} \rightsquigarrow \lambda\alpha\rho \rightarrow \lambda\beta\rho$

многостъпков извод $\{\alpha \overset{\ell \text{ стъпки}}{\Rightarrow} \beta \quad \lambda, \rho \in (V \cup \Sigma)^*\}$

$$\rightsquigarrow \lambda\alpha\rho \xrightarrow{\ell} \lambda\beta\rho$$

$$\alpha \underset{G}{\Rightarrow} \alpha_1 \underset{G}{\Rightarrow} \alpha_2 \underset{G}{\Rightarrow} \dots \alpha_\ell \underset{G}{\Rightarrow} \beta$$

CamScanner

език на граматика $G = \langle V, \Sigma, R, S \rangle$ е $L(G) = \{w \in \Sigma^* \mid S \overset{*}{\underset{G}{\Rightarrow}} w\}$

или аналогично $L(G) = w \in \Sigma^* \mid S \overset{l}{\Rightarrow} w$ за някое брой l-стъпки изведеда дума w

регулярна граматика $G = \langle V, \Sigma, R, S \rangle$
се нарича ако правилото в G са от видa $A \to \varepsilon$, $A \to aB$

КСГ $\to$ $G = \langle V, \Sigma, R, S \rangle$ ако правилата в G са от вида $A \to \alpha$ $A \in V$ $\alpha \in (V \cup \Sigma)^*$

стеков автомат НСА наричаме седморка

$$P = (Q, \Sigma, \Gamma, \#, q_{start}, q_{accept}, \Delta)$$

където:

$Q \to$ крайно мн-во от състояния
$\Sigma \to$ крайна азбука
$\Gamma \to$ крайна стекова азбука
$\# \to$ символ за дъно на стека $\# \notin \Sigma \cup \Gamma$
$q_{st} \to$ начално съст.
$q_{acc} \to$ финално съст.
$\Delta \to$ релация на преходите която е подмножество на:

$$\Delta : (Q \times (\Sigma \times \{\varepsilon\}) \times \Gamma) \times (Q \times \Gamma^{\leq 2})$$

конфигурация на стеков автомат: $(q, \alpha, \gamma) \in Q \times \Sigma^* \times \Gamma^*$
автомата се намира в състояние q, остава да се прочете α от входа и съдържанието на стека е γ.

едностъпков преход $\vdash_P$ релацията за едностъпков преход

$$(q, b\alpha, A\gamma) \vdash_P (p, \alpha, \beta, \gamma)$$

(25) Контролно 1 зад 1.2.

докажете че $L(r_1) = L(r_2)$ където

$r_1 = (r.s^* + s.t^* + t.r^*)^*$

$r_2 = (r^* . (s + r^*.s^*).t^*)^*$ и r, s, t са произволни рег. изрази.

Ще докажем че $L(r_1) = L((r+s+t)^*) = L(r_2)$

$\boxed{v^*.v^* = v^* \quad (u^*)^* = v^*}$ ще използвами

1. $r_1 \subseteq (r+s+t)^*$

$r \subseteq (r+s+t)^*$, $s^* \subseteq (r+s+t)^*$

$\Rightarrow r.s^* \subseteq (r+s+t)^*.(r+s+t)^* = (r+s+t)^*$

аналогично $s.t^*$ и $t.r^*$.

$\Rightarrow r.s^* + st^* + t.r^* \subseteq (r+s+t)^*$ и така

$r_1 = (r.s^* + st^* + tr^*)^* \subseteq ((r+s+t)^*)^* = (r+s+t)^*$

2. $(r+s+t)^* \subseteq r_1$

$r = r.\varepsilon \subseteq r.s^* \subseteq r.s^* + st^* + tr^*$

$s = s.\varepsilon \subseteq s.t^* \subseteq r.s^* + st^* + tr^*$

$t = t.\varepsilon \subseteq tr^* \subseteq r.s^* + st^* + tr^*$ и така

$r+s+t \subseteq rs^* + st^* + tr^*$

$(r+s+t)^* \subseteq (rs^* + st^* + tr^*)^* = r_1$.

CamScanner

3. $r_2 \subseteq (r+s+t)^*$

$r^* \subseteq (r+s+t)^*$

$s \subseteq (r+s+t)^*$

$r^* \subseteq (r+s+t)^*$ $s^* \subseteq (r+s+t)^*$ и така че

$r^*.s^* \subseteq (r+s+t)^* \cdot (r+s+t)^* = (r+s+t)^*$

и $s + r^*s^* \subseteq (r+s+t)^*$ и

$t^* \subseteq (r+s+t)^*$

по този начин $r^*.(s+r^*s^*)t^* \subseteq (r+s+t)^*(r+s+t)^*.(r+s+t)^*$

$=(r+s+t)^*$

4. $(r+s+t)^* \subseteq r_2$

$r = r.\varepsilon.\varepsilon \subseteq r^*(s+r^*s^*)t^*$

$s = \varepsilon s \varepsilon \subseteq$ –//–

$t = \varepsilon\varepsilon s \subseteq$ –//– .

(86) вариант 2 контролно 1 зад 1.2

докажете че за произволен рег. израз r с $L(r) \neq \emptyset$ съществува регулярен израз s, т.че s не съдържа символа $\emptyset$ и $L(r) = L(s)$

индукция по r.

<u>база:</u> тъй като $L(r) \neq \emptyset$, базата е $r = \varepsilon$ или $r \in \Sigma$
в такъв случай очевидно $r = s$.

<u>стъпка</u> нека твърдението е изпълнено за изразите r_1 и r_2
• да разгледаме обединението. $\boxed{r = r_1 + r_2}$

<u>1 сл</u> $L(r_1) \neq \emptyset$ и $L(r_2) \neq \emptyset$ от И.П имаме s_1 и s_2 които не съдържат $\emptyset$ и имат същите езици като r_1 и r_2 така че $s = s_1 + s_2$ върши работа.

<u>2 сл</u> $L(r_1) \neq \emptyset$ $L(r_2) = \emptyset$. от ИП. имаме $s_1 \neq \emptyset$
така че $s = s_1$ същи език като r_1 който върши работа защото
$L(r) = L(r_1)$

<u>3 сл</u> $L(r_1) = \emptyset$ $L(r_2) \neq \emptyset$. от ИП. $s_2 \neq \emptyset$ $s = s_2$
аналогично като 2 сл.

<u>4 сл.</u> $L(r_1) = L(r_2) = \emptyset$. невъзможно защото $L(r) \neq \emptyset$

конкатенация $\boxed{r = r_1 \cdot r_2}$
тъй като $L(r) \neq \emptyset$ то е ясно че $L(r_1) \neq \emptyset$ и $L(r_2) \neq \emptyset$
от ИП имаме съответните s_1 и s_2 които не съдържат $\emptyset$
така че $s = s_1 s_2$

звезда на клини $\boxed{r = r_1^*}$
ако $L(r_1) = \emptyset$ то може да вземем $\emptyset^* = \{\varepsilon\}$ $\boxed{s = \varepsilon}$
ако $L(r_1) \neq \emptyset$ то от ИП. имаме s_1 който $\neq \emptyset$ така че $s = s_1^*$

CamScanner

Вариант 3

(87) Докажете че $L(r_1) = L(r_2)$ където

$r_1 = (s^*.t + t^* r + r^* s)^*$

$r_2 = ((r+\varepsilon).t^* \cdot (s+r^*))^*$

ще докажем че $L(r_1) = (t+r+s)^* = L(r_2)$

1. $r_1 \subseteq (t+r+s)^*$

$s^* \subseteq$ -||- $\quad t \subseteq$ -||- $\Rightarrow s^*.t = (t+r+s)^*.(t+r+s)^* = (t+r+s)^*$

аналогично и за $t^* r$ и $r^* s$

$\Rightarrow r_1 \subseteq ((t+r+s)^*)^* = r_4 \subseteq (t+r+s)^*$

2. $(t+r+s)^* \subseteq r_1$

$t = \varepsilon t \subseteq s^* t \subseteq s^* t + t^* r + r^* s$

$r = \varepsilon t \subseteq t^* r \subseteq$ -||-

$s = \varepsilon s \subseteq r^* s \subseteq$ -||-

$\Rightarrow (t+r+s)^* \subseteq r_1$

3. $r_2 \subseteq (t+r+s)^*$

$r \subseteq (t+r+s)^* \quad \varepsilon \subseteq (t+r+s)^* \quad t^* \subseteq (t+r+s)^*$

$\Rightarrow (r+\varepsilon)t^* \subseteq (t+r+s)^*$

$s \subseteq (t+r+s)^* \quad r^* \subseteq (t+r+s)^*$

$\Rightarrow (s+r^*) \subseteq (t+r+s)^*$

по този начин $((t+r+s)(t+r+s)^* \cdot (t+r+s))^* = ((t+r+s)^*)^* = (t+r+s)^*$

$r_2 \subseteq (t+r+s)^*$

4. $(t+r+s)^* \subseteq r_2$

$t = \varepsilon t \varepsilon \qquad s = \varepsilon\, \varepsilon s$

$r = r \varepsilon \varepsilon$ $\Big\}\in r_2$ $\quad \square$

(88) Нека L_1 и L_2 са нерегулярни езици, които удовлетворяват условията в лемата за покачване. Докажете че $L_1 \cdot L_2$ също удовлетворяват това условие.

пример. Нека $L_1 = L_2$ са нерег. езици

пример $L_1 = \{a^n b^n \mid n \geq 0\} = L_2$

$L_1 \cdot L_2$

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

$a \ldots\ldots a b \ldots\ldots b a \ldots a b \ldots\ldots b$

Нека $\exists !\ p > 0$ т.ч $w = a^p b^p a^p b^p$

$W = xyz \qquad |xy| \leq p \qquad W_i = x y^i z$

y съдържа поне 2 различни символи

$$\underbrace{a \ldots a}_{x}\underbrace{b \ldots}_{y}\underbrace{b a \ldots a b \ldots b}_{z}$$

$W_i = a^{|x|} a^{i|y|} b^{|x|} b^{i|y|} a^{p - \cancel{|x|} - |y|} b^{-p - \cancel{|x|} - |y|} a^p b^p$

$w_i = a^{(i-1)|y| + 2p} b^{(i-1)|y| + 2p} = a^{2p + (i-1)|y|} b^{2p + (i-1)|y|}$

xy съдържа само a $\qquad a^{2p} b^{2p} = a^n b^n$

$\Rightarrow$ същото както $a^p b^p$

само б. $\Rightarrow$ -||-

(89)

Контролно 2 вариант 1

Зад 1.2
При даден ДКА A опишете регулярна граматика G т.че $L(G) = L(A)$. Изисква се доказателство, че вашето описание е коректно

Нека $A = \langle \Sigma, Q, q_{st}, F, \delta \rangle$ и $G = \langle V, \Sigma, R, S \rangle$

$Q = \{q_0, q_1, \dots q_k\}$ $R = \{(A_i \to x A_j \mid \delta(q_i, x) = q_j)$

$V = \{A_0, A_1, \dots, A_k\}$ $\cup (A_i \to \varepsilon \mid q_i \in F)\}$

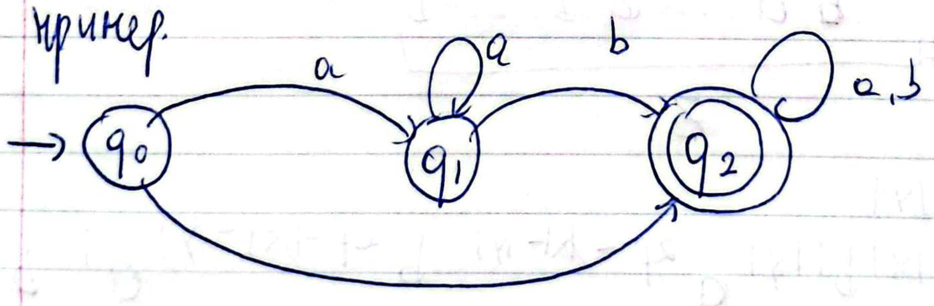
$q_0 = q_{start}$

$S \Rightarrow A_0$
$S \to A_1$ преименуващи правило

$\delta^*(q_i, \alpha) = q_j \iff A_i \overset{*}{\underset{G}{\Rightarrow}} \alpha A_j$

индукция по α $\alpha \in L(A) \iff \alpha \in L(G)$

пример.



$V = \{A_0, A_1, A_2\}$

$\Sigma = \{a, b\}$

$R: A_0 \to a A_1 \mid b A_2$

$A_1 \to a A_1 \mid b A_2$

$A_2 \to a A_1 \mid b A_2 \mid \varepsilon$

$S = A_0$

вариант 2

(90) При дадена рег. граматика G, опишете НДКА N т.че $L(N) = L(G)$ докажете го.

доказателство

$G = \langle V, \Sigma, R, S \rangle$ $N = \langle \Sigma, Q, Q_{st}, F, \Delta \rangle$

$V = \langle A_0, A_1, \dots A_n \rangle$

$S = A_0$

$Q = \langle q_0, q_1, \dots q_n \rangle$

$Q_{start} = \{q_0\}$

$\Delta = (q_i, x q_j) = \delta(q_i, x) = q_j$

$A_i \underset{G}{\longrightarrow} A_j$

$F = \{q_i \mid A \underset{G}{\longrightarrow} \varepsilon\}$

$\alpha \in L(N) \Leftrightarrow \exists q_j \in F, q_j \in \Delta^* (\{q_0\}, \alpha)$

$\Leftrightarrow \exists q_j \in F \, (A_0 \overset{*}{\underset{G}{\Rightarrow}} \alpha A_j) \Leftrightarrow \exists_j \, (A_j \underset{G}{\rightarrow} \varepsilon \text{ и } S \overset{*}{\underset{G}{\Rightarrow}} \alpha A_j)$

$\Leftrightarrow S \overset{*}{\underset{G}{\Rightarrow}} \alpha \Leftrightarrow \alpha \in L(G)$

вариант 1

(91) Нека G е зададена с правилото

$S \to ASC \mid Cb$
$A \to S \mid b \mid \varepsilon$
$C \to aAA \mid b \mid AS$

Като използвате изучения алгоритъм намерете G' в НФЧ т.че $L(G') = L(G)$.

$Gen[0] = [\phi]$
$Gen[1] = [A, C]$
$Gen[2] = [A, C]$

? намиране G' и представяне в НФЧ.

вариант 3

(92) Нека G е зададена с правилата

$S \to AC \mid Ca$
$A \to baA \mid b$
$C \to bCa \mid bA$

Постройте стеков автомат P т.че $L(P) = L(G)$

$P = \langle Q, \Sigma, \Gamma, \#, q_{st}, q_{acc}, \Delta \rangle$

$Q = \langle q_{st}, q_{acc}, p \rangle$ $\quad \Sigma = \{ab\}$

$\Gamma = \{a, b, A, C, \#\}$

$\Delta(q_{st}, \varepsilon, \#) = (p, S\#)$
$\Delta(p, \varepsilon, S) = \{(p, AC), (p, Ca)\}$
$\Delta(p, \varepsilon, C) = \{(p, bCa), (p, bA)\}$
$\boxed{\Delta(p, \varepsilon, \#) = (q_{acc}, \varepsilon)}$ последно.
$\Delta(p, \varepsilon, A) = \{(p, baA)\ (p, b)\}$
$\Delta(p, b, b) = \{p, \varepsilon\}$
$\Delta(p, a, a) = (p, \varepsilon)$
$\Delta(p, c, c) = (p, \varepsilon)$

$(p, bbab, \#)$
$\downarrow$
$(p, b^2ab, \varepsilon) / \phi$
$(p, bab, b\#)$
$\vdash$
$(p, ab\ b\#)$
$\vdash$
$(p, b, ab\ \#)$ *
$\vdash$
$(p, \varepsilon, \#)$
$\vdash$
$(f, \varepsilon, \#)$
успешно.

93 дефинирайте понятията:

език на състояние на A, класове, автомат на Бжозовски, приемане и отхвърляне на дума от машина на Тюринг, разрешим и полуразрешим език, изчислима ф.

език на състояние на A означава езика за ДКА A която разпознава или приема това състояние

$$L(A) = \{ q_f \mid q_f \in F \}$$

класове $\to \alpha \in L \Rightarrow [\alpha]_L \subseteq L$
клас на еквивалентност.

Нека $L \in \Sigma^*$ е рег. език, тогава $\exists$ КДА A който разпознава L с точно толкова състояния, колкото са класовете на еквивалентност относно релацията $\approx_L$
$\Rightarrow L$ е рег $\Leftrightarrow$ индексът на релацията $\approx$ е краен
тоест $|I \approx_L| = n \in \mathbb{N}$.

автомат на Бжозовски

$$B = (Q^B, \Sigma, q_{st}^B, \delta^B, F^B)$$

където Σ е крайна азбука

$$Q^B = \{ \alpha^{-1}(L) \mid \alpha \in \Sigma^* \}$$

$$q_{st}^B = L = \varepsilon^{-1}(L)$$

$$\delta^B(M, a) = a^{-1}(M)$$

$$F^B = \{ \alpha^{-1}(L) \mid \varepsilon \in \alpha^{-1}(L)$$

CamScanner

Приемане и отхвърляне на думи от машината на Тюринг
— M приема $\alpha \in \Sigma^*$ ако $(\varepsilon, q_{start}, \alpha \sqcup) \vdash^* (\lambda, q_{accept}, \rho)$
M отхвърля $\alpha \in \Sigma^*$ ако $(\varepsilon, q_{start}, \alpha \sqcup) \vdash^* (\lambda, q_{rej}, \rho)$

разрешим и полуразрешим език

Казваме че езика $L \subseteq \Sigma^*$ е разрешим ако $\exists$ M, т.че $L = L(M)$ и M е разрешител.
Казваме че $L \subseteq \Sigma^*$ е полуразрешим, ако $\exists$ M т.че $L = L(M)$. M разпознава езика L

Изчислима ф-я — една тотална ф-я $f: \Sigma^* \to \Sigma^*$ се нарича изчислима с машина на Тюринг M ако за $\forall \alpha \in \Sigma^*$
$$(\varepsilon, q_{start}, \alpha) \vdash^*_M (\varepsilon, q_{accept}, f(\alpha) \sqcup^k) \text{ за някое } k \in \mathbb{N}$$
това означава че M винаги завършва.

синтактично дърво P → Нека G е КСГ съвместимо с G. Наричаме $P = (T, \lambda)$ където
- T е кореново дърво с корен r за $\forall v \in T$ $succ_v = \{w_1, w_2 \dots w_k\}$
- $\lambda: T \to V \cup \Sigma \cup \{\varepsilon\}$ $\lambda(v)$ етикет на v $\lambda(r) \in V \cup \Sigma$
- за $\forall$ връх $v \in T$ $succ_v = \{w_1, \dots w_k\}$ $k \geq 1$
трябва $\lambda(v)_G \to \lambda(w_1)\lambda(w_2) \dots \lambda(w_k)$
- Ако $v \in T$ и $succ_v = \{w\}$ допускаме $\lambda(w) = \varepsilon$
трябва $\lambda(v)_G \to \varepsilon$ частен случай.

Нека $l_1, l_2, \dots l_k$ е обхождане на T в дълбочина на принципа най-ляв необходен тогава казваме че дървото извежда думата $\lambda(l_1)\lambda(l_2) \dots \lambda(l_k)$
$$yield(P) = \lambda(l_1)\lambda(l_2) \dots \lambda(l_k).$$

CamScanner

Изпит 1

(94) Опишете изучената конструкция с която се показва, че ако L_1 и L_2 са автоматни езици над Σ, то $L_1 \cdot L_2$, също е автоматни езици над Σ. Освен описането се изисква и доказателство.

Нека $L_1 = L(A_1)$ $A_1 = (\Sigma, Q_1, q_s^1, \delta_1, F_1)$

Нека $L_2 = L(A_2)$ $A_2 = (\Sigma, Q_2, q_s^2, \delta_2, F_2)$

A_1 и A_2 са детерминирани и $Q_1 \cap Q_2 = \emptyset$

строим $N = (\Sigma, Q_1 \cup Q_2, Q_{st}, \Delta, F)$

$$\Delta(q, a) \begin{cases} \{\delta_1(q, a)\} & q \in Q_1 \setminus F_1 \\ \{\delta_1(q, a)\ \delta_2(q^1_{st}, a)\} & q \in F \\ \{\delta_2(q, a)\} & q \in Q_2 \end{cases}$$

$L(N) = L(A_1) L(A_2)$

$(\subseteq)$ Нека $\alpha \in L(A_1)$ и $\beta \in L(A_2)$

$\delta_1^*(q^1_{st}, \alpha) = f_1 \in F_1$

N съдържа A_1 $(q^1_{st}, \alpha) \xrightarrow[N]{*} (f_1 \varepsilon)$

$\beta \neq \varepsilon$ $\beta = bp$

$q''_{st} \xrightarrow{b} \underset{\in Q_2}{q_2} \xrightarrow{p} f_2 \in F_2$ N съдържа A_2 $(q_2, p) \xrightarrow[N]{*} (f_2, \varepsilon)$

$\therefore f_1 \in F_1 \Rightarrow \delta_2(q''_{st}, b) \in \Delta(f_1 b)$

$(q^1_{st}, \alpha\beta) \xrightarrow[N]{*} (f_2, \beta) = (f_1, bp) \longrightarrow (q_2, p) \xrightarrow[N]{*} (f_2, \varepsilon)$

$\Rightarrow \alpha\beta \in L(N)$

(2) Нека $w \in L(N)$
има успешно изпълнение $q'_{st} \xrightarrow{w} q_2 \in F_2$
тъй като изпълнението започва в Q_1 и свършва
в Q_2

$$q'_{st} \xrightarrow{\alpha} f_1 \xrightarrow{b \in \Sigma} \underset{\delta(q''_{st}, b)}{\overset{\|}{q_2}} \xrightarrow{\beta} f_2 \in F_2$$

в първата част са само преходи в A_1
$\Rightarrow \delta_1^*(q'_{st}, \alpha) = f_1 \in F_1$

в 2° част са само преходи в A_2
$w = \alpha b \beta \in L(A_1).L(A_2)$

(95) L^*
Нека $L = L(A)$ $A = (\Sigma, Q, q_{st}, \delta, F)$ строим $N = (\Sigma, Q, Q_s, \Delta, F)$

$$\Delta(q,a) \begin{cases} \{\delta(q,a)\} & q \in Q \setminus F \\ \{\delta(q,a)\}\ \underbrace{\delta(q_{st}, a)}_{\text{нов пр}}\ \} & q \in F \end{cases}$$

$L(N) = (L(A))^+$
$\Rightarrow$) $\alpha \in L(N)$ има успешно изпълнение $q_{st} \xrightarrow{\alpha} f \in F$
маркиране $\forall$ места в които се използва нов преход

$$q_{start} \xrightarrow{\alpha_0} f_1 \xrightarrow{a_1} q_1 \xrightarrow{\alpha_1} f_2 \xrightarrow{a_2} q_2 \rightarrow \dots \rightarrow q_l \xrightarrow{\alpha_l} f_{l+1} = f$$

$\delta^*(q_{st}, \alpha) = f_1$

$\delta^*(q_{st}, a_i \alpha_i) = \delta^*(q_i, \alpha_i) = f_{i+1} \in F$

$\alpha = \alpha_0 a_1 \alpha_1 \dots a_l \alpha_l \Rightarrow \alpha \in (L(A))^+$

продължение на 95

$\Leftarrow$) Нека $\alpha \in L(A)^+$ тогава $\alpha = \alpha_0 \alpha_1 \dots \alpha_l$

$(q_{st}, \alpha_0) \xrightarrow[A]{*} (f_0, \varepsilon)$ $f_0 \in F$

$(q_{st}, a_i \alpha_i') \underset{A}{\vdash} (q_i, \alpha_i') \xrightarrow[A]{*} (f_i, \varepsilon)$

$q_1 \in \Delta(f_0, a_1)$ N съборна A

$q_2 \in \Delta(f_1, a_2)$ $(q_{st}, \alpha) = (q_{st}, \alpha_0 a_1 \alpha_1' a_2 \alpha_2' \dots a_l \alpha_l)$

$\xrightarrow[N]{*} (f_0, a_1 \alpha_1' \dots a_l \alpha_l) \xrightarrow[N]{}$

$\xrightarrow[N]{*} (f_0\, a_2 \alpha_2' \dots a_l \alpha_l') \xrightarrow[N]{*}$

$\underset{N}{\vdash} (q_2, \alpha_i \dots a_l \alpha_l) \xrightarrow[N]{*} \dots \xrightarrow[N]{*}$

$\xrightarrow[N]{*} (f_l, \varepsilon)$

$L^* = L^+ \cup \{\varepsilon\} = (L(A))^+ \cup \{\varepsilon\} = L(N) \cup \{\varepsilon\}$

уизнит

(96) Нека $\Sigma = \{a, b\}$ докажете че $\exists$ нерег. езици L_1 и L_2 над Σ т.че $L_1^* \{ab\} L_2$ е рег.

1 начин Нека $L_1 = \{a^n b^n \mid n \in \mathbb{N}\} \cup \{a, b\}$ и $L_2 = \{a^n (ab)^n \mid n \in \mathbb{N}\}$
тъй като $a, b \in L_1$ то е ясно че $L_1^* = (a+b)^*$
при това положение $L^* (a,b) L_2$ се състои от $\forall$ думи които завършват на ab.
последният език е рег. тъй като се задава с $(a+b)^* ab$

2 начин Нека $L_1 = L_2 = \{(ab)^n \mid n$ е просто$\}$
с помощта на лемата за покачване се вижда че този език не е рег.
от една страна $L_1^* = \{(ab)^n \mid n = 0$ или $n \geq 2\}$
$\forall n \geq 2$ може да се представи като сума на прости числа
от тук лесно се съобразява че $L_1^* \{ab\} L_2 =$
$= \{(ab)^n \mid n \geq 3\}$ и този език е рег. тъй като се задава с израза $ababab(ab)^*$
уизнит 1

(97) Каква е връзката между езиците на състоянията на A и класовете на $\approx$ позовски?

В има най малкия възможен брой състояния измежду автомата A които разпознават L.

→ При даден език L и ДКА $A = \langle \Sigma, Q, q_s, \delta, F \rangle$ който го разпознава, има ДКА A' който разпознава L с точно толкова състояния, колкото са класовете на еквивалентност относно рел. $\approx_L$

доказателство

$$w \in L_A(\delta(p, x)) \iff \delta^*(\delta(p,x), w) \in F$$
$$\iff \delta^*(p, xw) \in F$$
$$\iff xw \in L_A(p)$$

продължение на 97.

$w \in L_A(\delta(q,x)) \iff xw \in L_A(q)$

$p \equiv_A q \Rightarrow L_A(p) = L_A(q) \Rightarrow L_A(\delta(p,x)) = L_A(\delta(q,x))$
$\Rightarrow \delta(p,x) \equiv_A \delta(q,x)$

A' (тук може би да се замести с B)

$A' = (\Sigma, Q', q'_{st}, \delta', F')$

$Q' = \{ [q]_{\equiv_A} \mid q \in Q \}$

$q'_{st} = [q_{st}]_{\equiv_A}$ $\qquad \delta'([q]_{\equiv_A}, x) = [\delta(q,x)]_{\equiv_A}$

изпит 1

98 формулирайте и докажете критерия за минималност на КДА без недостижими състояния.

Нека A е свързан КДА който разпознава L
A е минимален $\iff \forall p \in Q\ \forall q \in Q\ p \neq q \Rightarrow$
$\Rightarrow L_A(p) \neq L_A(q)$

доказателство
$\Rightarrow$ A е минимален. тогава f е биекция $f(q) = L_A(q)$
в частност f е инекция

$\Leftarrow$ f е инекция $f: Q^A \to Q^B$ тогава $|Q^A| \leq |Q^B|$
и B е минимален $\Rightarrow$ A е минимален.

(99) Нека $\Sigma = \{a, b\}$ G е зададена с правилата

$S \to aS \mid aSb \mid \varepsilon$. Докажете че езикът

$U = \{\alpha \in L(G) \mid \exists!$ дърво на извод P т.че $yield(P) = \alpha\}$ е КС.

- Тривиално се проверява че $L(G) = \{a^n b^m \mid n \geq m\}$
- Ще докажем че $U = \{a^n \mid n \in \mathbb{N}\} \cup \{a^n b^n \mid n \in \mathbb{N}\}$
- $\forall$ дума от вида a^n има единствено дърво на извод при което се използва n пъти първото правило и един път 3° правило.

$S \Rightarrow aS \Rightarrow aaS \Rightarrow \ldots \Rightarrow aa\ldots aS \Rightarrow a^n$.

- другата поредност на правилата е невъзможна тъй като $S \to \varepsilon$ трябва да бъде последна.

- Също очевидно 2° правило не може да се използва защото ще има b. в извода.

- $\forall$ дума от вида $a^n b^n$ има ! дърво на извод при което се използва n пъти 2° правило и един път 3° правило

$S \to aSb \to \ldots \to a\ldots aSb\ldots b \xrightarrow{\varepsilon} a\ldots ab\ldots b = a^n b^n$

- отново друга поредност е невъзможна. Да допуснем че има извод на думата която използва 1° правило този извод би имал вида

$S \Rightarrow^* a^k S b^k \Rightarrow a^{k+1} S b^k \Rightarrow \ldots$ и той $\neq a^n b^n$

U е КС като обединение на 2 КС езика

- останалите думи в $L(G)$ имат вида $a^n b^m$ $n > m \geq 1$
$\Leftarrow$ Никоя от тях не $\notin U$ защото за нея има по $\neq$ 2 дървета на извод. Може едно започващо с 1° правило $(n>m)$ и може едно с 2° правило.

(100) Нека $\Sigma = \{a, b\}$ и $\# \notin \Sigma$. За ф-я $f : \Sigma^* \to \Sigma^*$ означаваме
$Gr(f) = \{\alpha \# f(\alpha) \mid \alpha \in \Sigma^*\}$

докажете че f е изчислима ф тогава и само тогава когато $Gr(f)$ е разрешим език.

$\Rightarrow$) Нека f е изчислима ф-я.
- Строим м. Тюринг M която разрешава $Gr(f)$.
- M има 2 ленти.
- При дадена входна дума β, M започва да копира β от 1° в/у 2° лентата до когато стигне символа $\#$.
- Ако достигне ⊔ преди $\#$, M отхвърля β.
- В противен случай, M симулира машината за f в/у 2° лента. Когато симулацията приключи M сравнява съдържанието от 2° лента с остатъка от β (след $\#$).
- В случай че съвпадат M приеме β, в случай че не съвпадат M отхвърля β.

$\Leftarrow$) Нека $Gr(f)$ е разрешим език.
- Строим маш. Тюринг N която изчислява ф-ята f.
- N има 3 ленти.
- При входна дума α, N организира следния цикъл:
  - копира α от 1° лента върху 3° и добавя символа $\#$ след което копира 2° лента в/у 3° лента.
  - симулира маш. разрешител за $Gr(f)$ в/у 3° лента.
- Ако симулацията приключи успешно, изходът на N (думата $f(\alpha)$) се намира на 2° лента.
- Ако е неуспешна, N генерира в/у 2° лента следващата дума в каноничната наредба, изтрива 3° лента, и преминава към нова итерация на цикъла.
- N винаги ще завърши работа тъй като f е тотална т.е $f(\alpha)$ е дефинирано за $\forall \alpha \in \Sigma^*$

CamScanner

(101) Нека $\Sigma = \{a, b\}$ и $\# \notin \Sigma$, докажете че $L \subseteq \Sigma^*$ е полуразрешим $\Leftrightarrow \exists$ разрешим език $R \subseteq (\Sigma \cup \{\#\})^*$ със свойството:

$\alpha \in L \Leftrightarrow \exists \beta \in \Sigma^*$ т.че $\alpha \# \beta \in R$

за $\forall \alpha \in \Sigma^*$

appropriate

$\Leftarrow$) трябва да се организира търсене на подходяща дума β по същия начин както $\Leftarrow$) на зад 190 (използва се разрешител R в место $Gr(f)$).
- Ако $\exists \beta$ то тя ще бъде намерена и α ще се разпознае.
- Ако $\nexists \beta$ то търсенето ще зацикли ∞ и α няма да се разпознае.

$\Rightarrow$) - Нека L е полуразрешим език и нека M разпоз. L.

- деф. N с 2 ленти която работи по следния начин:
1. Ако входът няма вида $\alpha \# \beta$ то N зав. неуспешно.
2. Нека входът е $\alpha \# \beta$.

N копира α върху 2ра лента и след това симулира M в/у тази лента за толкова на брой стъпки, колкото е $|\beta|$ т.е не $\forall$ стъпка премества главата на първата лента на дясно.

Ако симулацията на M е успешно $\rightarrow N$ приключ. усп
- В случай че в/у 1ва лента се достига $\sqcup$ и M не е усп $\rightarrow N$ не е усп
- да съобразим че R има исканите свойства.
- Най напред N е разрешител, тъй като завършва при $\forall$ вход $\Rightarrow R$ е разрешим.
- Ако $\alpha \in L$ то α се разпознава от M и това става по n брой стъпки $n \in \mathbb{N}$.
- $\alpha \# a^n \in R$ тъй като N ще приключи успешно.
- Така може да вземем ... $\beta = a^n$.
- Обратно нека $\exists \beta \in \Sigma^*$ т.че $\alpha \# \beta \in R \Rightarrow N$ успешно но това е възможно $\Leftrightarrow M$ прикл. успешно т.е когато $\alpha \in L$.