

Kako je definiran **končni avtomat**?

končni avtomat je definiran s petročkom:

$$M = \langle Q, \Sigma, \delta, q_0, F \rangle$$

Q - končna množica stanj

Σ - vhodna abeceda

δ - funkcija prehodov

q_0 - začetno stanje

F - množica končnih stanj

Kako slikajo **funkcije prehodov** δ
pri ε -NKA, NKA in DKA?

$$\varepsilon\text{-NKA} \mid \delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$$

$$\text{NKA} \mid \delta : Q \times \Sigma \rightarrow 2^Q$$

$$\text{DKA} \mid \delta : Q \times \Sigma \rightarrow Q$$

Kaj je **abeceda** Σ ?

abeceda je neprazna
končna množica simbolov

$$\Sigma = \{a_1, a_2, \dots, a_n\}$$

Kaj je **niz** oz. **beseda**?

niz je končno zaporedje
simbolov neke abecede

$$w = a_1 a_2 \dots a_k; a_i \in \Sigma$$

Kaj je **regularna gramatika**?

regularna gramatika je levo ali desno
linearna gramatika, definirana kot
četrček spremenljivk, končnih
znakov, produkcij in začetnega
simbola gramatike

$$G = \langle V, T, P, S \rangle$$

$$V \cup T = \Sigma \quad V \cap T = \emptyset$$

V - množica spremenljivk oz. vmesnih simbolov $\subseteq \Sigma$

T - množica znakov oz. končnih simbolov $\subset \Sigma$

P - množica produkcij

S - začetni simbol $\in N$

Kako predstavimo **vse**
nize dolžine k nad abecedo Σ

vse nize dolžine k nad neko abecedo
izrazimo kot vse stike k -tih simbolov

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^k = \{a_1 a_2 \dots a_k \mid \langle a_1, a_2, \dots, a_k \rangle \in \times_k \Sigma\}$$

Kako predstavimo **vse**
nize nad abecedo Σ

vse nize nad neko abecedo izrazimo
kot unijo vseh potenc abecede

$$\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$$

Kaj je **jezik** L ?

jezik je poljubna množica
nizov nad neko abecedo

$$L(\Sigma) \subseteq \Sigma^*$$

S katerimi pravili je definiran **regularni izraz**?

regularni izraz je definiran
z naslednjimi pravili:

regularni izrazi \emptyset , ε in a
opisujejo jezike \emptyset , $\{\varepsilon\}$ in $\{a\}$

$(r_1 + r_2)$ opisuje jezik $L(r_1) \cup L(r_2)$
 $(r_1 r_2)$ opisuje jezik $L(r_1)L(r_2)$
 $(r_1)^*$ opisuje jezik $(L(r_1))^*$

Kaj je **relacija izpeljave** pri prepisovalnem sistemu?

relacija izpeljave je relacija, ki
preslika nek niz v drug niz z uporabo
produkcije iz prepisovalnega sistema

$$(\implies) = \{\langle \alpha A \beta, \alpha B \beta \rangle \mid (A \rightarrow B) \in P\}$$

Kaj je **regularni jezik**?

regularni jezik je tak, ki ga lahko opišemo z regularnim izrazom, končnim avtomatom, ...

Kako opišemo **jezike avtomatov**
 ε -NKA, NKA in DKA?

jezike avtomatov opišemo s tistimi nizi, ki se začnejo s q_0 in se končajo v stanju avtomata, ki je končno

$$L(M_\varepsilon) = \{w \mid \hat{\delta}(q_0, w) \cap F \neq \emptyset\}$$

$$L(M_N) = \{w \mid \hat{\delta}(q_0, w) \cap F \neq \emptyset\}$$

$$L(M_D) = \{w \mid \delta(q_0, w) \in F\}$$

Kako **pretvorimo** ε -NKA v NKA

ε -NKA **pretvorimo** v NKA tako, da odstranimo vse ε povezave, ter preverimo katere povezave moramo dodati da dobimo spet isti jezik

Kaj je **prepisovalni sistem**?

prepisovalni sistem je par abecede in množice produkcij

$$G = \langle \Sigma, P \rangle$$

$$P \subset \Sigma^* \times \Sigma^*$$

$$\begin{array}{l} \text{produkcija} \\ \omega_1 \rightarrow \omega_2 \in P \end{array}$$

$$\begin{array}{l} \text{izpeljava} \\ \alpha\omega_1\beta \Rightarrow \alpha\omega_2\beta \end{array}$$

Kako poteka pretvorba KA v RI
po **metodi z eliminacijo**?

pri **metodi z eliminacijo** izbiramo vozlišča, ki jih bomo eliminirali, ter avtomat povežemo nazaj, tako,

da na nove povezave zapišemo regularne izraze,
 ki opisujejo prehode, ki smo jih eliminirali,
 dokler nam ne ostane le začetno in
 končno stanje, nato pa za zapis
 uporabimo naslednji
 regularni izraz

$$(R + SQ^*T)^*SQ^*$$

R - povezava $q_0 \rightarrow q_0$

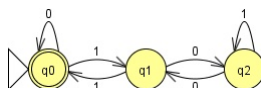
S - povezava $q_0 \rightarrow q_e$

Q - povezava $q_e \rightarrow q_e$

T - povezava $q_e \rightarrow q_0$

(*todo slika*)

Kakšen je DKA za **preverjanje**
deljivosti s 3 v binarnem sistemu?



$$(0 + 1(01^*0)^*1)^*$$

Katere operacije ohranjajo **regularnost jezika**?

regularnost jezika ohranjajo:

unija, stik, iteracija (po def.)
 presek, komplement, razlika, obrat, XOR

Kako dobimo **presek** dveh regularnih jezikov?

presek dveh avtomatov dobimo
 kot t.i. produktni avtomat

$$M_1 = \langle Q_1, \Sigma, \delta_1, q_{10}, F_1 \rangle$$

$$M_2 = \langle Q_2, \Sigma, \delta_2, q_{20}, F_2 \rangle$$

$$M_1 \cap M_2 = \langle Q_1 \times Q_2, \Sigma, \delta', (q_{10}, q_{20}), F_1 \times F_2 \rangle$$

$$\delta'((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$$

končna so tista stanja, ki
 so končna v obeh avtomatih

Kako dobimo **komplement** regularnega jezika?

komplement regularnega jezika dobimo
 tako, da nekončna stanja postanejo

končna ter obratno

$$M = \langle Q, \Sigma, \delta, q_0, F \rangle$$

$$M = \langle Q, \Sigma, \delta, q_0, Q \setminus F \rangle$$

Kako dobimo **razliko** dveh regularnih jezikov?

razliko dveh regularnih jezikov dobimo tako, da razliko spremenimo v presek s komplementom drugega jezika in naredimo ustrezen produktni avtomat

$$L_1 \setminus L_2 = L_1 \cap \overline{L_2}$$

Kako dobimo **obrat** regularnega jezika?

obrat regularnega jezika dobimo tako, da končna stanja ε -povežemo z novim začetnim, začetno postane končno, poleg tega pa obrnemo še vse smeri povezav

$$M = \langle Q, \Sigma, \delta, q_0, F \rangle$$

$$M^R = \langle Q \cup \{q'_0\}, \Sigma \cup \{\varepsilon\}, \delta', q'_0, \{q_0\} \rangle$$

$$\begin{aligned} \delta'(q'_0, \varepsilon) &= F \\ \delta(q_1, a) = q_2 &\iff \delta'(q_2, a) = q_1 \end{aligned}$$

Kako pokažemo **enakost** dveh regularnih jezikov?

enakost dveh regularnih jezikov pokažemo

z minimizacijo | minimizacija DKA je enolična

$$\mathbf{z množicami} \mid L_1 \setminus L_2 \cup L_2 \setminus L_1 = \emptyset$$

Kdaj pravimo, da je kontekstno neodvisna gramatika **dvoumna**?

kontekstno neodvisna gramatika je **dvoumna**, kadar kako izmed besed iz jezika lahko dobimo po več različnih izpeljavah

Kdaj pravimo, da ima kontekstno neodvisna gramatika **determinističen jezik**?

kontekstno neodvisna gramatika ima
determinističen jezik, kadar obstaja
nedvoumna kontekstno neodvisna
gramatika z enakim jezikom

G' je nedvoumna
 $L(G) = L(G')$

Kako **poenostavimo**
kontekstno neodvisno gramatiko?

kontekstno neodvisno gramatiko
poenostavimo tako, da odpravimo:

ε **produkcije** |

enotske produkcije |

nekoristne simbole | nedosegljivi in taki brez končnih
simbolov v izpeljavi

Kako **minimiziramo** končne avtomate?

končne avtomate minimiziramo
po naslednjem postopku:

pare stanj (končno, nekončno) označimo kot neekvivalenčna

...

Kako je definiran **jezik skladovnega avtomata**?

jezik skladovnega avtomata je definiran kot
zaporedje korakov do končnega stanja, ali pa
kot zaporedje korakov, ki izprazni sklad

$$L_F(M) = \{w \mid \langle q_0, w, Z_0 \rangle \vdash^* \langle q_F, \varepsilon, \gamma \rangle \wedge q_F \in F\}$$

$$L_S(M) = \{w \mid \langle q_0, w, Z_0 \rangle \vdash^* \langle q, \alpha, \varepsilon \rangle\}$$
