Σ = alfabet ; L - limbaj ; w = cuvânt
Q - mulț. stări finale; F = mulț. st. finale
δ = funcție tranziție ; $\delta : Q \times \Sigma \to Q$
$L^0 = \{\lambda\} \neq \emptyset$; $\emptyset^0 = \{\lambda\} = \{\lambda\}^0$; $\emptyset L = L\emptyset = \emptyset$
$L^m = L^{m-1} \cdot L$ concatenare ; $L^* = L^0 \cup L^1 \ldots$
$L^+ = L^1 \cup L^2 \ldots$; $L^* = L^+ \cup \{\lambda\}$; $(L^*)^* = L^*$
$L(L_1 \cup L_2) = LL_1 \cup LL_2$; $(L_1 \cup L_2)L = L_1L \cup L_2L$
$w^* = \{\lambda, w, ww, \ldots\}$; $\bar{L} = \Sigma^* \setminus L$; $\emptyset^* = \{\lambda\}$; $\emptyset^+ = \emptyset$

DFA = $(Q, \Sigma, \delta, q_0, F)$;
$\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a)$
NFA = $(Q, \Sigma, \delta, q_0, F)$
$\delta : Q \times \Sigma \to 2^Q$
$(q_0, abba) \vdash (q_3, bba) \vdash (q_2, ba) \vdash$
$\vdash (q_2, a) \vdash (q_2, \lambda) \in F$
L(NFA) = L(DFA)
Orice limbaj finit e acc. de DFA
λ-NFA = $(Q, \Sigma, \delta, q_0, F)$
$\delta : Q \times (\Sigma \cup \{\lambda\}) \to 2^Q$
Complementul unui DFA e tot DFA

NFA → DFA transformare

δ	a	b
q_0	q_1	q_0, q_3
q_1	q_2	q_4
q_2	q_2	q_2
q_3	$\emptyset$	q_0
q_4	q_0, q_2, q_3	$\emptyset$

NFA

δ	a	b
q_0 F	q_1	q_{03}
q_1	q_2	q_4
q_{03} F	$\emptyset \cup q_1 = q_1$	$q_0 \cup \{q_0, q_3\} = q_{03}$
q_2 F	q_2	q_2
q_4	q_{023}	$\emptyset$
q_{023}	$q_1 \cup q_2 \cup \emptyset = q_{12}$	$q_{03} \cup q_2 \cup q_0 = q_{023}$
q_{12} F	$q_2 \cup q_2 = q_2$	$q_4 \cup q_2 = q_{24}$
q_{24} F	$q_2 \cup q_{023} = q_{023}$	$q_2 \cup \emptyset = q_2$

DFA

stările finale în DFA sunt cele care conțin cel puțin un indice din NFA (st. f.)

$\langle q \rangle$ închiderea unei stări = $\{q\} \cup$ mulț. stărilor în care ajungi doar cu λ pornind din q

$L = L_1 \cap L_2 = \{w \in \Sigma^* \mid w \in L_1$ și $w \in L_2\}$
$F = F_1 \times F_2$
$L = L_1 \cup L_2 = \{w \in \Sigma^* \mid w \in L_1$ sau $w \in L_2\}$
$F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$
$L = L_1 \setminus L_2 = \{w \in \Sigma^* \mid w \in L_1$ și $w \notin L_2\}$
$F = (Q_1 \setminus F_1) \times F_2$
$L = L_2 \setminus L_1 = \ldots$
$L = \{w \mid w \in a^*b^*\} \Rightarrow \bar{L} = \Sigma^* \setminus L = \{a,b\}^* \setminus \{a^*, b^*\}$
(q_0, r_0) stare init
$\delta((q_i, r_i), x) = (\delta_1(q_i, x), \delta_2(r_i, x))$

L = reg $\Rightarrow \exists p \in \mathbb{N}$ a.î. $\forall \alpha \in L$ cuvânt cu $|\alpha| \geq p$, $\exists\ \alpha = u \cdot v \cdot w$ cu propr.:
1) $|u \cdot v| \leq p$
2) $|v| \geq 1$
3) $u \cdot v^i \cdot w \in L, \forall i \geq 0$

λNFA → NFA/DFA transformare
Stare inițială DFA = închiderea stării inițiale din NFA-λ
Stări finale DFA = cele care conțin cel puțin o stare care în NFA-λ avea în λ-închidere cel puțin o stare finală.

δ NFA-λ	a	b	λ	λ^* = λ-închiderea
q_0 init	$q_0 q_1$	q_2	q_2, q_3	$q_0, q_2, q_3, q_4, q_5, q_6$
q_1	$\emptyset$	$\emptyset$	q_2	q_1, q_2, q_4, q_6
q_2 F	q_3	$\emptyset$	q_4	q_2, q_4, q_6
q_3	q_6	q_3, q_6	q_5	q_3, q_5, q_2, q_6, q_4
q_4	q_6	q_5	q_6	q_4, q_6
q_5	q_6	q_2	q_6	q_5, q_2, q_4, q_6
q_6 F	$\emptyset$	q_6	$\emptyset$	q_6

NFA-λ $\quad \lambda^* a \lambda^*$
q_0 i $\xrightarrow{\lambda^*} q_{023456} \xrightarrow{a} q_{0136} \xrightarrow{\lambda^*} q_{0123456}$
$q_1 \xrightarrow{\lambda^*} q_{1246} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$
q_2 F $\xrightarrow{\lambda^*} q_{246} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$
$q_3 \xrightarrow{\lambda^*} q_{23456} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$
$q_4 \xrightarrow{\lambda^*} q_{46} \xrightarrow{a} q_6 \xrightarrow{\lambda^*} q_6$
$q_5 \xrightarrow{\lambda^*} q_{2456} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$
q_6 F $\xrightarrow{\lambda^*} q_6 \xrightarrow{a} \emptyset \longrightarrow \emptyset$

$\lambda^* b \lambda^*$
$q_0 \xrightarrow{\lambda^*} q_{023456} \xrightarrow{b} q_{2356} \xrightarrow{\lambda^*} q_{23456}$
$q_1 \xrightarrow{\lambda^*} q_{1246} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$
$q_2 \xrightarrow{\lambda^*} q_{246} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$
$q_3 \xrightarrow{\lambda^*} q_{23456} \xrightarrow{b} q_{2356} \xrightarrow{\lambda^*} q_{23456}$
$q_4 \xrightarrow{\lambda^*} q_{46} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$
$q_5 \xrightarrow{\lambda^*} q_{2456} \xrightarrow{b} q_{256} \xrightarrow{\lambda^*} q_{2456}$
$q_6 \xrightarrow{\lambda^*} q_6 \xrightarrow{b} q_6 \xrightarrow{\lambda^*} q_6$

NFA-δ	a (rezultat a)	b (rezultat b)
q_0 init, F	$q_{0123456}$	q_{23456}
q_1, F	q_{23456}	q_{2456}
q_2, F	q_{23456}	q_{2456}
q_3, F	q_{23456}	q_{23456}
q_4, F	q_6	q_{2456}
q_5, F	q_{23456}	q_{2456}
q_6, F	$\emptyset$	q_6

DFA-δ	a	b
q_{023456} init	$q_{0123456}$	q_{23456}
$q_{0123456}$ F	$q_{0123456}$	q_{23456}
q_{23456} F	q_{23456}	q_{23456}

$L = \{a^m b^m \mid m \geq 0\} \notin REG$ Exemplu Lema pompare
Pp. L reg $\Rightarrow \exists_p \in \mathbb{N}$ ..... Lema pompare
Fie $\alpha = a^p b^p \in L$ cu $|\alpha| = 2p \geq p$
$\alpha = u \cdot v \cdot w$
Caz I: $v = a^k$ din (1) și (2) $\Rightarrow 1 \leq |v| \leq p$, $|u| = 0$
Deci $1 \leq |a^k| \leq p \Rightarrow 1 \leq k \leq p$ (*)
Din (3) $\Rightarrow u \cdot v^i w \in L_0, \forall i \geq 0$
Alegem $i = 2 \Rightarrow \beta = u \cdot v^2 \cdot w = a^{p+k} b^p \notin L_0$
pt. că din (*) $k \geq 1$

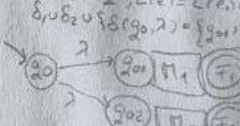
Σ fixat exp. reg = RegEx(Σ), mulț. de cuv. peste alfabetul $\Sigma \cup \{(,), +, \cdot, *, \emptyset, \lambda\}$
1) $\emptyset, \lambda \in$ RegEx și $a \in$ RegEx, $\forall a \in \Sigma$
2) $e_1, e_2 \in$ RegEx $\Rightarrow (e_1 + e_2) \in$ RegEx reuniune
3) — " — $\Rightarrow (e_1 \cdot e_2) \in$ — " — concat
4) $e \in$ RegEx $\Rightarrow e^* \in$ RegEx stelare
Precedența operatorilor: () > * > · > +

RegEx → λNFA

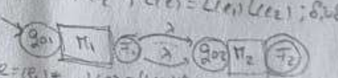
RegEx	$e = \emptyset$	$e = \lambda$	$e = a, a \in \Sigma$
Limbaj	$L = \emptyset$	$L = \{\lambda\}$	$L = \{a\}$
AF	→ q_0	→ (q_0)	→ $q_0 \xrightarrow{a}$ (q_1)

Caz i

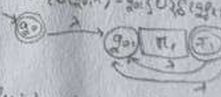
Caz ii, iii, iv) pp. că expr. reg e_k și $L(e_k)$, $k \in \{1,2\}$ avem deja AF$(L(e_k)) = (Q_k, \Sigma_k, q_{0k}, F_k, \delta_k)$ cu $Q_1 \cap Q_2 = \emptyset$ disjuncte
→ q_{0k} $\boxed{M_k}$ F_k unde M_k = toate celelalte stări și tranziții ale automatului
Reuniune: $e = e_1 + e_2$; $L(e) = L(e_1) \cup L(e_2)$
$\delta_1 \cup \delta_2 \cup \{\delta(q_0, \lambda) = \{q_{01}, q_{02}\}\}$



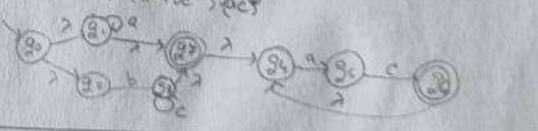
Concatenare: $e = e_1 \cdot e_2$; $L(e) = L(e_1)L(e_2)$; $\delta_1 \cup \delta_2 \cup \{\delta(q_f, \lambda) \ni q_{02}\ \forall q_f \in F_1\}$



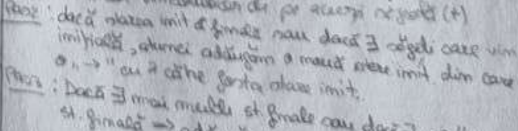
Stelare: $e = (e_1)^*$, $L(e) = (L(e_1))^*$,
$\delta_1 \cup \{\delta(q_0, \lambda) = q_{01}\} \cup \{\delta(q_{f_1}, \lambda) \ni q_{01} \mid \forall q_f \in F_1\}$

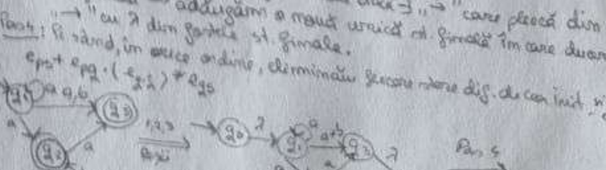


Exemplu: $L_1 = a^*$, $L_2 = bc^*$, $L_3 = ac$, $L_4 = ((L_1 + L_2) \cdot (L_3))^*$
$L_4 = (a^* + bc^*)(ac)^*$



NFAλ → RegEx
AFE = automat finit extins, $M = (Q, \Sigma, et, q_0, F)$, funcție etichetare $et : Q \times Q \to$ RegEx(Σ)
$et(p, q) \Rightarrow e_{pq}$ expr. reg. asoc. săgeții de la $p \to q$
Pas 1: reuninea simbolurilor de pe aceeași săgeată (+)
Pas 2: dacă starea init e finală sau dacă $\exists$ săgeți care vin către starea inițială, atunci adăugăm o nouă stare init. din care ducem o „→" cu λ către fosta stare init.
Pas 3: Dacă $\exists$ mai multe st. finale sau dacă $\exists$ „→" care pleacă din st. finală → adăugăm o nouă unică st. finală în care ducem „→" cu λ din fostele st. finale.
Pas 4: Pe rând, în orice ordine, eliminăm fiecare stare dif. de cea init. și f.
$e_{ps} + e_{pq} \cdot (e_{qq})^* e_{qs}$





Minimizare DFA

stări echiv.: $\forall p,q \in Q, p \equiv q \iff \forall w \in \Sigma^*, \hat{\delta}(p,w) \in F \iff \hat{\delta}(q,w) \in F$

stări separabile: —"— $p \not\equiv q \iff \exists w \in \Sigma^*, \hat{\delta}$ —"— $\iff \hat{\delta}(q,w) \notin F$

cuvinte echivalente: $x \not\equiv y$ conform lui L dacă $\exists z$ cuvânt a.î. exact unul din cuvintele $xz, yz \in L$ și celălalt nu

ex: $L = \{a^m b^m \mid m \geq 3\} \Rightarrow a^4 b^4 \not\equiv \lambda$ $(a^4 b^4 b^5 \notin L, ab^5 \notin L)$

$a^2 b^4 \equiv ab^6 \Rightarrow z = b^2 \Rightarrow a^7 b^2 = a^7 b^8$

$z / b^2 \Rightarrow$ nu găs

DFA trb să fie complet definit, dacă nu, atunci se adaugă o stare nefinală faux, toate tranzițiile lipsă se adaugă spre acea stare + o buclă cu toate simbolurile din alfabet

Partiționăm stările care distinctiv, dacă putem de acea partiție, se alipesc q_{13} de exemplu, practic

δ	a	b
q_1	q_3	q_4
q_2	q_3	q_4

$q_1 \equiv q_2 \Rightarrow q_{12}$

Partițiile A,B,C ...

Σ^* conține $\{\lambda\}$ și dacă $x \in \Sigma^*$, atunci $ax \in \Sigma^* \; \forall a \in \Sigma$

Gramatici: $G = (N, T, S, P)$; $N = \{A, B, C\}$ simboluri neterminale $T = \{a, b, c\}$ simb. term; $S \in N$ simb. start; P = producții mult.

Gramatici reg: $A \to aB \mid a \mid \lambda$

Exemplu: $L = \{a^{2m+1} \mid m \geq 1\}$ și λ care $\notin T$ fără A,B ... ⇒ stop

$S \to aA$; $A \to aS \mid aB$; $B \to a$

GIC/CFG/context-free grammar: $A \to \alpha$; $A \in N$ și $\alpha \in (N \cup T)^*$

Exemple: $L_1 = \{a^m b^m / m \geq 0\}$: $S \to aSb \mid \lambda$

$L_2 = \{a^{2m} b^k c^{3m} \mid m \geq 1, k \geq 1\} \Rightarrow S \to aaScc / aaAccc$

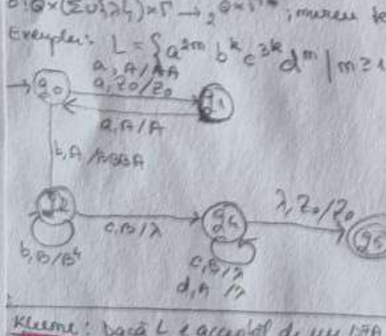
$A \to bA \mid b$

APD = automate push-down = $(Q, \Sigma, \Gamma, \delta, q_0, F, z_0)$; restul la fel

$\Gamma = \{A, B, C ...\}$ alfabet stivă; $\Sigma = \{a, b, c ...\}$ alfabet intrare; $z_0 \in \Gamma$ simb inițial al stivei

$\delta: Q \times (\Sigma \cup \{\lambda\}) \times \Gamma \to 2^{Q \times \Gamma^*}$; mereu terminăm cu $\lambda, z_0/\lambda$

Exemplu: $L = \{a^{2m} b^k c^{3k} d^m \mid m \geq 1; k \geq 1\}$



Kleene: Dacă L e acceptat de un DFA A, atunci $\exists$ E expr reg. a.î. $L(E) = L$. Dem:

Pt $k=0$, R^0_{ij} este o mulț. finită formată din toate literele pt care $\exists$ tranziții de la i la j și eventual λ dacă $i=j$, r^0_{ij}; poate fi scris ca o inducție printre acele litere

Pp că pt $\forall R^k_{ij}$ putem construi o expr reg r^k_{ij} cu $L(r^k_{ij}) = R^k_{ij}$. Definim $r^{k+1}_{ij} = r^k_{ij} \cup (r^k_{i,k+1} (r^k_{k+1,k+1})^* r^k_{k+1,j})$. Arătăm că corespunde: $L(r^{k+1}_{ij}) = L(r^k_{ij} \cup (r^k_{i,k+1} (r^k_{k+1,k+1})^* r^k_{k+1,j})) = L(r^k_{ij}) \cup (L(r^k_{i,k+1}) L(r^k_{k+1,k+1})^* L(r^k_{k+1,j})) = R^k_{ij} \cup (R^k_{i,k+1} (R^k_{k+1,k+1})^* R^k_{k+1,j}) = R^{k+1}_{ij}$

Lema pompare CFL: Fie L un CFL $\Rightarrow \exists p \in \mathbb{N}$ a.î. $\forall \alpha \in L$ cuvânt cu $|\alpha| \geq p$, $\exists \alpha = u \cdot v \cdot w \cdot x \cdot y$ cu propr: 1) $|v \cdot w \cdot x| \leq p$; 2) $|v \cdot x| \geq 1$ 3) $u \cdot v^i \cdot w \cdot x^i \cdot y \in L \; \forall i \in \mathbb{N}$

FNC = forma normală Chomsky: G e în FNC dacă $A \to BC$; $A \to a$

$S \to \lambda$ dacă S nu mai apare în niciun membru drept

Alg. transf. CFG → FNC

Pas I: 1) eliminăm simb. neterminale ⇒ păstrăm A care are măcar o producție fără litere mari ($A \to bc \mid d \mid \lambda$) sau care cele au doar litere mari ($D \to BAd \mid B$), practic eliminăm recursivitatea fără stop

2) eliminăm simb. unreachable ⇒ dacă nu se ajunge niciodată în A

Pas II: 1) dacă neterminalul are λ producții și nu alte producții de ex: $B \to \lambda$ când dispare B din toate celelalte producții, apare în locul lor una fără B

exemplu: $A \to aABa$ / $D \to B \mid BAd$ $\Rightarrow$ $A \to aAa$ / $D \to Ad \mid \lambda$

sau eliminăm doar λ-producțiile, unde dacă $A \to bc \mid d \mid \lambda$ și $B \to Ad$; $C \to A$ ⇒ după ce producția cu λ [illegible] A ⇒ adică: $A \to bc \mid d$, $B \to Ad \mid d$; $C \to A \mid \lambda$

repetăm pasul ăsta până nu mai avem ce să eliminăm

dacă avem $S \to \lambda$ și $C \to S$, atunci se adaugă $S' \to \lambda \mid S$, deci am și aut $S \to \lambda$ și nu are voie ca S să ducă în lambda [illegible] că e înlocuit de $S' \to S \mid \lambda$ și S devine o stare normală, care nu mai apare nicăieri în dr. membrului

Pas III: eliminăm „unit products" = producțiile formate doar din A sau B (o literă mare) și e înlocuită cu toate producțiile ei

ex: $C \to A$; $A \to bc \mid d \Rightarrow C \to bc \mid d$ și $A \to bc \mid d$

Pas IV: se repetă pas I, II, III

Pas V: adăugăm neterminale noi pt terminalele din cuv de lungime > 1

$A \to aa \mid aAa \mid bc \mid d \mid Ad$ ⇒ $A \to X_1X_1 \mid X_1AX_1 \mid X_2X_3 \mid X_4 \mid AX_4$

$X_1 \to a$, $X_2 \to b$, $X_3 \to c$; $X_4 \to d$ ⇒ $X_1 \to a$; $X_2 \to b$; $X_3 \to c$; $X_4 \to d$

Pas VI: se adaugă neterminale noi pt. a sparge cuv. de lungime > 2

$A \to X_1AX_1 \mid X_4X_3X_4X_2X_2$ ⇒ $A \to X_1Y_1 \mid X_4Y_6 \mid$

$X_1 \to a$; $X_2 \to b$ ... ⇒ $Y_1 \to AX_1$; $Y_6 \to X_3Y_7$; $Y_7 \to X_4Y_8$; $Y_8 \to X_2X_2$

Σ = alfabet, L = limbaj, w = cuv
Q = mulț. stări finită ; F = mulț. st. finale
δ = fct. tranziție : $\delta: Q\times\Sigma \to Q$
$L^0 = \{\lambda\} \neq \emptyset$; $\emptyset^0 = \{\lambda\} = \{\emptyset\}$.
$\emptyset L = \emptyset$; $L^m = L^{m-1} \circ L$ concatenare
$L^* = L^0 \cup L^1 \cup L^2 \ldots$; $L^+ = L^1 \cup L^2 \cup \ldots$
$L^* = L^+ \cup \{\lambda\}$; $(L^*)^* = L^*$
$L(L_1 \cup L_2) = LL_1 \cup LL_2$
$(L_1 \cup L_2)L = L_1L \cup L_2L$
$w^* = \{\lambda, w, ww, www, \ldots\}$
$\overline{L} = \Sigma^* \setminus L$; $\emptyset^* = \{\lambda\}$; $\emptyset^n = \emptyset$
$A \setminus B = A \cap \overline{B}$

DFA = $(Q, \Sigma, \delta, q_0, F)$
NFA => $\delta: Q \times \Sigma \to 2^Q$
λ-NFA => $\delta: Q \times (\Sigma \cup \{\lambda\}) \to 2^Q$
$L_{NFA} = L_{DFA}$

Orice limbaj finit e acceptat de un DFA.
Complementul unui DFA este tot un DFA

$L = L_1 \cap L_2 = \{w \in \Sigma^* \mid w \in L_1$ și $w \in L_2\}$
$F = F_1 \times F_2$
$L = L_1 \cup L_2 = \{w \in \Sigma^* \mid w \in L_1$ sau $L_2\}$
$F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$
$L = L_1 \setminus L_2 = \{w \in \Sigma^* \mid w \in L_1$ și $w \notin L_2\}$
$F = (Q_1 \setminus F_1) \times F_2$
$L = \{w \mid w \in a^*b^*\} \Rightarrow \overline{L} = \Sigma^* \setminus L = \{a,b\}^* \setminus \{a^*, b^*\}$
(q_0, r_0) stare init

λNFA → NFA/DFA transformare

$\langle q \rangle$ = închiderea unei stări = $\{q\} \cup$ mulțimea stărilor în care ajungi doar cu λ pornind din q
stare init DFA = închiderea stării inițiale în NFA-λ
stări finale DFA = cele care conțin cel puțin o stare care în λNFA avea λ închiderea cel puțin o stare finală.

δNFAλ	a	b	λ	λ* (λ-închiderea)
q_0 init	q_0, q_1	q_2	q_2, q_3	$\langle q_0 \rangle = \{q_0, q_2, q_3, q_4, q_5, q_6\}$
q_1	∅	∅	q_2	q_1, q_2, q_4, q_6
$q_2 \in F$	q_3	∅	q_4	q_2, q_4, q_6
q_3	q_6	q_3, q_6	q_5	q_2, q_3, q_4, q_5, q_6
q_4	q_6	q_5	q_6	q_4, q_6
q_5	q_6	q_2	q_2, q_6	q_2, q_4, q_5, q_6
$q_6 \in F$	∅	q_6	∅	q_6

δDFA	a	b
$\langle q_0 \rangle = q_{023456}$ init, ∈F	$q_{0123456}$	q_{23456}
$q_{0123456}$ F	$q_{0123456}$	q_{23456}
q_{23456} F	q_{23456}	q_{23456}

în NFAλ	λ*aλ*	λ*bλ*	δNFA a	δNFA b
q_0 init	$q_0 \xrightarrow{\lambda^*} q_{023456} \xrightarrow{a} q_{0136} \xrightarrow{\lambda^*} q_{0123456}$	$q_0 \xrightarrow{\lambda^*} q_{023456} \xrightarrow{b} q_{2356} \xrightarrow{\lambda^*} q_{23456}$	$q_0, q_1, q_2, q_3, q_4, q_5, q_6$	q_2, q_3, q_4, q_5, q_6
q_1	$q_1 \xrightarrow{\lambda^*} q_{1246} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$	$q_1 \xrightarrow{\lambda^*} q_{1246} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$	q_2, q_3, q_4, q_5, q_6	q_2, q_4, q_5, q_6
$q_2 \in F$	$q_2 \xrightarrow{\lambda^*} q_{246} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$	$q_2 \xrightarrow{\lambda^*} q_{246} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$	q_2, q_3, q_4, q_5, q_6	q_2, q_4, q_5, q_6
q_3	$q_3 \xrightarrow{\lambda^*} q_{23456} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$	$q_3 \xrightarrow{\lambda^*} q_{23456} \xrightarrow{b} q_{2356} \xrightarrow{\lambda^*} q_{23456}$	q_2, q_3, q_4, q_5, q_6	q_2, q_3, q_4, q_5, q_6
q_4	$q_4 \xrightarrow{\lambda^*} q_{46} \xrightarrow{a} q_6 \xrightarrow{\lambda^*} q_6$	$q_4 \xrightarrow{\lambda^*} q_{46} \xrightarrow{b} q_{56} \xrightarrow{\lambda^*} q_{2456}$	q_6	q_2, q_4, q_5, q_6
q_5	$q_5 \xrightarrow{\lambda^*} q_{2456} \xrightarrow{a} q_{36} \xrightarrow{\lambda^*} q_{23456}$	$q_5 \xrightarrow{\lambda^*} q_{2456} \xrightarrow{b} q_{256} \xrightarrow{\lambda^*} q_{2456}$	q_2, q_3, q_4, q_5, q_6	q_2, q_4, q_5, q_6
$q_6 \in F$	$q_6 \xrightarrow{\lambda^*} q_6 \xrightarrow{a} \emptyset \xrightarrow{\lambda^*} \emptyset$	$q_6 \xrightarrow{\lambda^*} q_6 \xrightarrow{b} q_6 \xrightarrow{\lambda^*} q_6$	∅	q_6

În NFA, stările finale devin: $q_0, q_1, q_2, q_3, q_4, q_5, q_6$

NFA → DFA transformare

δNFA	a	b
$q_0 \in F$	q_1	$\{q_0, q_3\}$
q_1	q_2	q_4
$q_2 \in F$	q_2	q_2
q_3	∅	q_0
q_4	q_0, q_2, q_3	∅

δDFA	a	b
$q_0 \in F$	q_1	q_{03}
q_1	q_2	q_4
$q_{03} \in F$	$\emptyset \cup q_1 = q_1$	$q_0 \cup \{q_0, q_3\} = q_{03}$
$q_2 \in F$	q_2	q_2
q_4	q_{023}	∅
$q_{023} \in F$	$q_1 \cup q_2 \cup \emptyset = q_{12}$	$q_{03} \cup q_2 \cup q_0 = q_{023}$
$q_{12} \in F$	$q_2 \cup q_2 = q_2$	$q_4 \cup q_2 = q_{24}$
$q_{24} \in F$	$q_2 \cup q_{023} = q_{023}$	$q_2 \cup \emptyset = q_2$

stări finale în DFA sunt cele care conțin cel puțin un indice stare finală din NFA.

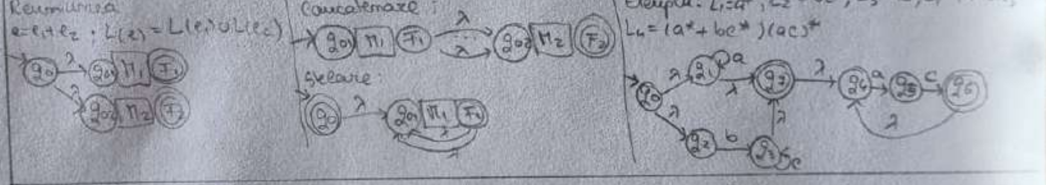
RegEx → λNFA

Caz i

RegEx	$e=\emptyset$	$e=\lambda$	$e=a, a\in\Sigma$
Limbaj	$L=\emptyset$	$L=\{\lambda\}$	$L=\{a\}$
AF			

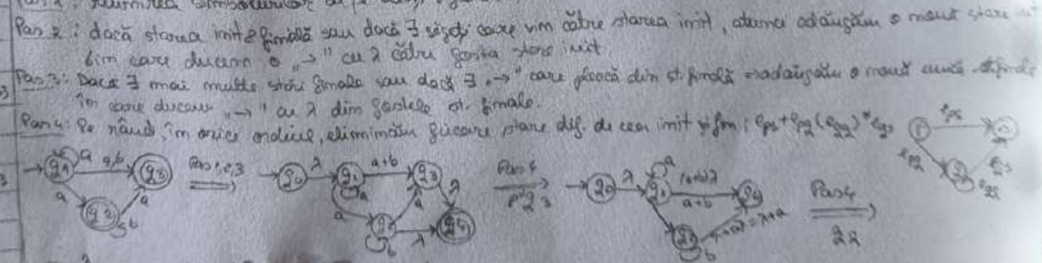
Caz ii, iii, iv
Pp. că e expr. reg e_k și $L(e_k)$, $k\in\{1,2\}$ avem deja AF $(L(e_k)) = (Q_k, F_k, \delta_k, q_{0k}, F_k, \Sigma_k)$
cu $Q_1 \cap Q_2 = \emptyset$
$(q_{0k})\ M_k\ (F_k)$ unde M_k = toate celelalte stări și tranziții ale automatului

Reuniunea
$e = e_1 + e_2$: $L(e) = L(e_1) \cup L(e_2)$
Concatenare:
Stelare:
Exemplu: $L_1 = a^*$, $L_2 = bc^*$, $L_3 = ac$; $L_4 = (L_1+L_2)L_3^*$
$L_4 = (a^* + bc^*)(ac)^*$



λNFA → RegEx

AFE = automat finit extins, $M = (Q, \Sigma, et, q_0, F)$, funcții etichetare $et: Q\times Q \to RegEx(\Sigma)$, $et(p,q) \Rightarrow e_{pq}$ expr. reg asoc. săgeții de la $p \to q$
Pas 1: reuniunea simbolurilor de pe aceeași săgeată (+)
Pas 2: dacă starea init e finală sau dacă ∃ săgeți care vin către starea init, atunci adăugăm o nouă stare init din care ducem o „→" cu λ către fosta stare init
Pas 3: Dacă ∃ mai multe stări finale sau dacă ∃ „→" care pleacă din st. finală ⇒ adăugăm o nouă unică st. finală în care ducem „→" cu λ din fostele st. finale.
Pas 4: Pe rând, în orice ordine, eliminăm fiecare stare dif. de cea init și fin: $e_{ps} + e_{pq}(e_{qq})^* e_{qs}$



$\lambda a^*(b + ab^* + ab^*a) = a^*b + a^*ab^* + a^*ab^*a$

RegEx(Σ) = fam. exp. reg. mulț. cuv peste alfabetul $\Sigma \cup \{(,), +, \cdot, *, \emptyset, \lambda\}$
+ ⇒ reuniune ; · ⇒ concat ; * ⇒ stelare
1) $\emptyset, \lambda \in RegEx$ și $a \in RegEx$ $\forall a \in \Sigma$
2) $e_1, e_2 \in RegEx \Rightarrow (e_1 + e_2) \in RegEx$
3) — " — $\Rightarrow (e_1 \cdot e_2) \in RegEx$
4) $e^* \in RegEx$
Precedența operații: () > * > · > +

L reg ⇒ ∃p∈ℕ a.î. ∀α∈L cuv cu |α| ≥ p ∃ α = u·v·w cu propr.:
1) $|u \cdot v| \le p$
2) $|v| \ge 1$
3) $u \cdot v^i \cdot w \in L, \forall i \ge 0$

Exemplu lema pompare
$L = \{a^m b^m \mid m \ge 0\} \notin REG$
Pp. L reg. ⇒ ∃p∈ℕ ... lema p
Fie $\alpha = a^p b^p \in L$ cu $|\alpha| = 2p \ge p$
$\alpha = u \cdot v \cdot w$
Caz I: $v = a^k$, din (1)(2) ⇒ $1 \le |v| \le p$, $|u| = 0$
Deci $1 \le |a^k| \le p \Rightarrow 1 \le k \le p$ (*)
Din (3) ⇒ $u \cdot v^i w \in L$, $\forall i \ge 0$
Alegem $i = 2 \Rightarrow \beta = u \cdot v^2 w = a^{p+k} b^p \notin L$
pt. că în (*) $k \ge 1$ ⇒ abs

Lema pompare CFL: Fie L un CFL ⇒ ∃p∈ℕ a.î. ∀α∈L cuvânt cu |α| ≥ p, ∃ α = u·v·w·x·y cu propr.:
1) $|v \cdot w \cdot x| \le p$ 2) $|vx| \ge 1$ 3) $u v^i w x^i y \in L$ $\forall i \in ℕ$

Σ^* conține $\{\lambda\}$ și dacă $x \in \Sigma^*$, atunci $ax \in \Sigma^*$ $\forall a \in \Sigma$

$L = \{a^m b^m c^n \mid m > n > n \ge 0\} \notin CFL$ valori i = 0 sau i = 2
Pp. p.a. L ∈ CFL ⇒ ∃p∈ℕ ... lema
alegem $\alpha = a^{p+2} b^{p+1} c^p \in L \Rightarrow |\alpha| = 3p+3 \ge p$ $\forall p \in ℕ$
Avem $\alpha = uvwxy$ a.î. $|vwx| \le p$ și $|vx| \ge 1$ ⇒
⇒ $1 \le |vx| \le p$ (*)
Fie $vx = a^k b^t$, $k \ge 1, t \ge 1 \overset{(*)}{\Rightarrow} 1 \le k + t \le p$
Alegem $i = 0 \Rightarrow \beta = a^{p+2-k} b^{p+1-t} c^p \in L \Rightarrow |\beta|_b > |\beta|_c \Leftrightarrow$
$\Leftrightarrow p+2-k < p+1-t \Rightarrow 1 < t$, dar $1 \le t$ ⇒ abs

Gramatici: $G = (N, T, S, P)$; $N = \{A, B, \}$ simb neterm.; $T = \{a, b, c\}$ simb term.; $S \in N$ simb start; P = mulț. producții; λ sau $d \in T$ fără A, B => stop

Gramatică reg: $A \to aB \mid a \mid \lambda$

Exemplu: $L = \{a^{2m+1} \mid m \geq 1\}$

$S \to aA$; $A \to aS \mid aB$; $B \to a$

GIC/CFG: $A \to \alpha$; $A \in N$ și $\alpha \in (N \cup T)^*$

Exemplu: $L_1 = \{a^m b^m \mid m \geq 0\}$: $S \to aSb \mid \lambda$

$L_2 = \{a^{2m} b^k c^{3m} \mid m \geq 1, k \geq 1\}$

$S \to aaSccc \mid aaAccc$

$A \to bA \mid b$

Minimizare DFA

Stări echiv: $\forall p, q \in Q$, $p \equiv q \Leftrightarrow \forall w \in \Sigma^*$, $\hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F$

altfel sunt separabile

Cuvinte echiv $= (x \not\equiv y) =$ conform. cu L dacă $\exists z$ cuv a.î. unul din cuv xz, $yz \in L$ și celălalt nu

Ex: $L = \{a^m b^m \mid m \geq 5\}$

$a^4 b^4 \neq \lambda$ ($a^4 b^4$, $a^5 b^5 \notin L$, $a^5 b^5 \in L$)

$a^4 b^5 = a^3 b^6$ $z = b^2$ grădina

DFA trb să fie complet definită, dacă nu, atunci se adaugă o stare nefinală (aux). Toate tranzițiile lipsă se adaugă spre acea stare + o buclă cu toate simb. din alfabet

Partiționare stările care nu sunt echiv, dacă partiția de aceeași partiție, se dispar

q_i	a	b
q_1	q_3	q_4
q_2	q_3	q_4

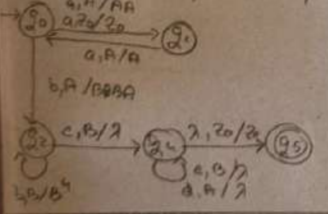
$q_1 = q_2 \Rightarrow q_{12}$

Partiții $A_i, B_i, C_i \ldots$

APD $= (Q, \Sigma, \Gamma, \delta, q_0, F, z_0)$; $\Gamma = \{A, B, C \ldots\}$ alfabet stivă, $\Sigma = \{a, b, c \ldots\}$ alfabet intrare; $z_0 \in \Gamma$ simb. inițial al stivei

$\delta: Q \times (\Sigma \cup \{\lambda\}) \times \Gamma \to 2^{Q \times \Gamma^*}$; mereu terminăm cu $\lambda, z_0 / \lambda$

Ex: $L = \{a^{2m} b^k c^{3k} d^m \mid m \geq 1, k \geq 1\}$



GIC ex: $L = \{0^{3k} 1^{\ell} 0^k \mid k \geq 0, \ell \geq 1\}_{S_1} \cdot \{0^i 1^{j+3} \mid i \neq j\}_{S_2}$

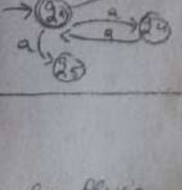
$S \to S_1 S_2 1^3$

$S_1 \to 0^3 S 0 \mid A$, $A \to 1A \mid 1$

$S_2 \to U \mid V$; $U \to 0U \mid 0T$; $V \to 1V \mid 1T$

$T \to 0T1 \mid \lambda$

Un NFA care nu e DFA și nici λNFA



Tauslav Alexio

FNC = forma normală Chomsky => G e în FNC dacă $A \to BC$; $A \to a$; $S \to \lambda$ dacă S nu mai apare în niciun membru drept

Alg transformare CFG → FNC

Pas I: 1) eliminăm simb + producțiile nonterminating => păstrăm A care are măcar o prod. fără litere mari ($A \to bc \mid d \mid \lambda$) sau care are doar litere mari ($D \to BAd \mid B$); practic eliminăm recursivitatea fără stop

2) eliminăm simb unreachable → dacă nu se ajunge niciodată în A

Pas II: 1) dacă neterminalul are λ-producții și nu alte producții de ex: $B \to \lambda$, când dispare B din toate celelalte producții, apare în locul lor una fără B, sau eliminăm doar

Exemplu: $A \to aABd$, $B \to B \mid BAd$ | $A \to aAd$, $B \to Ad \mid \lambda$

λ-producțiile, unde dacă $A \to bc \mid d \mid \lambda$ și $B \to Ad$; $C \to A$ → atunci apar producțiile cu λ în loc de A => adică: $A \to bc \mid d$; $B \to Ad \mid d$, $C \to A \mid \lambda$

Repetăm pasul ăsta până nu mai avem ce să eliminăm. Dacă eliminăm $S \to \lambda$ și $C \to S$, atunci se adaugă $S' \to \lambda \mid S$, pt. că ar fi p. $C \to \lambda$ și nu are voie ca S să ducă în λ una din producții, așa că e înlocuit de $S' \to S \mid \lambda$ și S devine o st. nouă S' care nu mai apare niciodată în dr. membrului

Pas III: eliminăm „unit products" = producțiile formate doar din A sau B (o literă mare) și e înlocuită cu toate producțiile ei

ex: $C \to A$; $A \to bc \mid d$ => $C \to bc \mid d$ și $A \to bc \mid d$

Pas IV: se repetă pașii anteriori

Pas V: adăugăm neterminale noi pt. terminalele din cuv. de lungime > 1

$A \to aa \mid aAa \mid bc \mid d \mid Ad$; $X_1 \to a$, $X_2 \to b$, $X_3 \to c$, $X_4 \to d$ | => $A \to X_1 X_1 \mid X_1 A X_1 \mid X_2 X_3 \mid X_4 \mid AX_4$; $X_1 \to a \ldots$

Pas VI: se adaugă neterminale noi pt. a sparge cuv. de lungime > 2

$A \to X_1 A X_1 \mid X_4 X_3 X_1 X_2 X_2$; $X_1 \to a$, $X_2 \to b$, $X_3 \to c$, $X_4 \to d$ | => $A \to X_1 Y_1 \mid X_4 Y_6 \mid$; $Y_1 \to AX_1$; $Y_6 \to X_3 Y_7$, $Y_7 \to X_1 Y_8$; $Y_8 \to X_2 X_2$

Teorie:

1) ∃ limbaje reg care nu sunt CFL => fals => $L_{reg} \subset CLF$ ierarhia Chomsky

2) $L_1 \subseteq L_2$, $L_2 \in REG$, deci $L_1 \in REG$? => fals, $L_1 = \{a^m b^m \mid m \in \mathbb{N}\}$, $L_2 = \{a, b\}^*$

3) este decidabil dacă limbajele acc. de 2 DFA sunt egale sau nu => aplicăm minimizare DFA pt. echivalare

4) gramaticile reg definesc un limbaj reg, pt ∀ limbaj reg se poate scrie un NFA

4) $L_2 \subseteq L_1$ și $L_2 \in CF$, atunci $L_1 \in CF$? => fals, $L_2 = \{a^m b^m \mid m \in \mathbb{N}\} \in CF$ $L_1 = L_2 \cup \{c^p \mid p \text{ prim}\} \notin CF$

5) $L_1 \subseteq L_2 \in REG$, $L_1 \in REG$, at $L_2 \in REG$? => fals, $L_1 = \{a^m \mid m\}$ și $L_2 = L_1 \cup \{b^p \mid p \text{ prim}\}$

6) $L_1 \cup L_2 = L_3$, $L_1, L_3 \in REG$, at $L_2 \in REG$? => $L_2 = L_3 = \{a, b\}^*$ $L_1 = \{a^m b^m \mid m \in \mathbb{N}\}$

7) orice gramatică reg. produce un limbaj reg.

8) $L_1 \cap \overline{L_2} = L_3$ și $L_2, L_3 \in REG$, at. $L_1 \in REG$? => fals $L_1 = \{a^m b^m \mid m \in \mathbb{N}\}$, $L_2 = \{a, b\}^*$, $L_3 = \emptyset$

$L_1 \cap \overline{L_2} = L_1 \cap \overline{\{a, b\}^*} = L_1 \cap \emptyset = \emptyset = L_3$

9) limbajele acceptate de pda sunt aceleași cu limbajele generate de GIC

10) e decidabil dacă un DFA acceptă toate cuv. de lungime 5 => adevărat, $L = \{w \mid |w| = 5\}$

11) $L_1 \cdot L_2 = L_3 L_4$, $L_1, L_2, L_3 \in REG$, at $L_4 \in REG$? => adevărat $L_1 = L_2 = \emptyset = L_4$ și $L_3 = \{a^m b^m c^m \mid m \geq 0\}$

(nu CF) (nu CF) $\{a^m b^m \mid m \geq 0\}$