

DFA $= (Q, \Sigma, \delta, q_0, F)$
Q skończony zbiór stanów
Σ skończony alfabet wejściowy
δ funkcja przejścia postaci $Q \times \Sigma \rightarrow Q$
q_0 stan początkowy
$F \subseteq Q$ zbiór stanów akceptujących
Minimalizacja DFA
<ol style="list-style-type: none"> forall p końcowy, q niekończowy, oznacz (p,q) forall $(p, q) \in (F \times F) \cup (Q \setminus F \times Q \setminus F), p \neq q$ if $\exists_{a \in \Sigma} (\delta(p, a), \delta(q, a))$ jest oznaczona, oznacz (p,q) (rekurencyjnie). nieoznaczone scalamy.
PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$
Q skończony zbiór stanów
Σ alfabet wejściowy
Γ alfabet stosowy
$q_0 \in Q$ stan początkowy
$Z_0 \in \Gamma$ symbol początkowy na stosie
$F \subset Q$ zbiór stanów akceptujących (jeśli $F = \emptyset$ to akceptujemy przez pusty stos)
δ funkcja przejścia postaci $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$
LOP Zał., że L regularny. Wtedy istnieje stała n , że jeśli $z \in L$ oraz $ z \geq n$, to można podzielić z na $z = uvw$ takie, że: <ol style="list-style-type: none"> $v \geq 1$ $uv \leq n$ $\forall i \in \mathbb{N} z^i = uv^i w \in L$
Podział $\alpha = uvw$, $ uv \leq n$ oraz $ v \geq 1$. Wybieramy i dla którego $ uv^i w \notin L$ a powinien.
LOP bezk. Zał., że L bezkontekstowy. Wtedy istnieje stała n , że jeśli $z \in L$ oraz $ z \geq n$, to można podzielić z na $z = uvwxy$, takie, że: <ol style="list-style-type: none"> $vx \geq 1$ $vwx \leq n$ $\forall i \in \mathbb{N} z^i = uv^i w x^i y \in L$
Lemat Ogdena Niech L język bezkontekstowy. Wtedy istnieje stała n taka, że jeśli $z \in L$ oraz $ z \geq n$ i oznaczymy w z n lub więcej pozycji jako wyróżnione, to można podzielić z na $z = uvwxy$ takie, że: <ol style="list-style-type: none"> v i x zawierają łącznie co najmniej jedną wyróżnioną pozycję vwx zawiera co najwyżej n wyróżnionych pozycji $\forall i \in \mathbb{N} z^i = uv^i w x^i y \in L$
Klasa języków regularnych jest domknięta na operację sumy, dopełnienia, przecięcia, złożenia i domknięcia Kleene’ego. Gramatyka bezkontekstowa $G=(N,T,P,S)$
N - skończony zbiór zmiennych (nieterminale)

T - skończony zbiór zmiennych końcowych(terminale, alfabet)
P - skończony zbiór produkcji postaci $A \rightarrow \alpha$ gdzie $A \in N$ i $\alpha \in (N \cup T)^*$
$S \in N$ - symbol początkowy

Postać normalna Chomsky’ego postaci:
 $A \rightarrow BC$ albo $A \rightarrow a$
Konstrukcje:

- If po prawej terminal a to zastępujemy go C_a i dopisujemy $C_a \rightarrow a$
- If prawa strona dłuższa niz 1 to zastępujemy $A \rightarrow B_1 \dots B_n$ przez $A \rightarrow B_1 D_1, D_1 \rightarrow B_2 D_2, \dots, D_{n-2} \rightarrow B_{n-1} B_n$

FIRST(X) - dla symboli

- X-terminal, to FIRST(X)=X
- $X \rightarrow \varepsilon$ to do FIRST(X) dodajemy ε
- X - nieterminal i $X \rightarrow Y_1 Y_2 \dots Y_k$ to dodajemy a do $FIRST(X)$ jeśli istnieje i takie, że $a \in FIRST(Y_i)$ oraz $\varepsilon \in FIRST(Y_j)$ dla każdego $j < i$.
 $\varepsilon \in FIRST(X)$ jeśli należy do wszystkich $FIRST(Y_i)$.
- $FIRST(X\alpha) = FIRST(X)$ gdy $\varepsilon \notin FIRST(X)$
- $FIRST(X\alpha) = FIRST(X) \cup FIRST(\alpha)$ gdy $\varepsilon \in FIRST(X)$

FOLLOW(A) - dla nieterminali

- Dla początkowego S do $FOLLOW(S)$ dodajemy $\$$
- Jeśli mamy produkcję $A \rightarrow \alpha B \beta$ to do $FOLLOW(B)$ dodajemy wszystkie symbole z $FIRST(\beta)$ poza ε
- Jeśli $A \rightarrow \alpha B$ lub $A \rightarrow \alpha B \beta$, gdzie $\varepsilon \in FIRST(\beta)$ to do $FOLLOW(B)$ dodajemy wszystkie symbole z $FOLLOW(A)$

LL(1) - $A \rightarrow \alpha$
Tabela: nazwy kolumn terminale i \$!!!⌈
nazwy wierszy nieterminale ⇔

- \forall produkcji $A \rightarrow \alpha$ z gramatyki wykonaj 2 i 3
- foreach $a \in T$ if $a \in FIRST(\alpha)$ to wpisz $A \rightarrow \alpha$ do $M[A, a]$
- if $\varepsilon \in FIRST(\alpha)$ to dla każdego $b \in FOLLOW(A)$ wpisz $A \rightarrow \alpha$ do $M[A, b]$ Jeżeli $\varepsilon \in FIRST(\alpha)$ oraz $\varepsilon \in FOLLOW(A)$, dodaj $A \rightarrow \alpha$ do $M[A, \$]$
- PROTIP: nie ma w tabeli ε !

SLR Tabela:
nazwy kolumn AKCJE (terminale i \$!!!)
i PRZEJŚCIA (nieterminale)
nazwy wierszy stany

- zbiory sytuacji $C = I_0, \dots, I_n$
Zaczynamy od $I_0 = domkniecie([S' \rightarrow \bullet S])$
- tabelka + redukcje (zaznaczyć ew. konflikty)
konstrukcja tabelki: w części akcji s_x (shift) i r_x (reduce), a w części przejść (nieterminale) x (liczba)
ACC dla $S' \rightarrow \bullet S$

- redukcja do FOLLOW(A) (if redukcja była z $A \rightarrow \beta$)

LR(1)

- zbiory sytuacji z PODGLĄDEM
- podgląd początkowy \$
- podgląd przy domknięciu: mamy $[A \rightarrow \alpha \bullet B \beta, a] \in I$ dla każdej produkcji z $B \rightarrow \gamma$ dodaj $[B \rightarrow \bullet \gamma, FIRST(Ba)]$
- tabelka + redukcje (zaznaczyć ew. konflikty)

LALR

- zbiory sytuacji z PODGLĄDEM (SLR, ale z podglądem z LR(1))
- sklejamy jądra

LEADING(A)-pierwsze term. z A

- $a \in LEADING(A)$ jeśli mamy produkcję $A \rightarrow Ba\beta$ lub $A \rightarrow a\beta$
- if exists prod. $A \rightarrow B\alpha$ i $a \in LEADING(B)$ to $a \in LEADING(A)$
- foreach nieterminali liczymy 1 i powtarzamy 2 aż nie się nie zmienia

TRAILING(A)-ostatnie term. z A

- $a \in TRAILING(A)$ jeśli mamy produkcję $A \rightarrow \beta aB$ lub $A \rightarrow \beta a$
- if exists prod. $A \rightarrow \alpha B$ i $a \in TRAILING(B)$ to $a \in TRAILING(A)$
- foreach nieterminali liczymy 1 i powtarzamy 2 aż nie się nie zmienia

Tab. priorytetów $\doteq \langle \rangle$

$TT \ T \doteq T$

$TNT \ T \doteq T$

TN foreach $a \in LEADING(N)$ do $T \lessdot a$ (wiersze) ⇔
 NT foreach $a \in TRAILING(N)$ do $a \gtrdot T$ (kolumny) ⌈
\$ zawsze gorszy

Zbiory sytuacji

- Wzbogacenie $S' \rightarrow S$
- Ponumerować produkcje (do redukcji!!!).

$E \rightarrow \varepsilon \ E \rightarrow \cdot$

- dla kropek, na końcu w tabeli numer z produkcji

Rekurencja

- $A \rightarrow A\alpha|B$
- $A \rightarrow \beta A'$
- $A' \rightarrow \alpha A'|\varepsilon$

Faktoryzacja

- $A \rightarrow \alpha \beta_1 | \dots | \alpha \beta_k$
- $A \rightarrow \alpha A'$
- $A' \rightarrow \beta_1 | \dots | \beta_k$

język	lem	slowo	notes
$\omega = xxy \wedge x \neq \varepsilon$	LOP	ab^na^n	$i = 0$
$\omega = xy yz \wedge y \neq \varepsilon$	reg	$len \geq 4$	dobrac krótsze
$\omega \omega^R \wedge \omega _a \equiv \omega _b \equiv 0(mod13)$	LOP	$a^{13n}b^{13n}b^{13n}a^{13n}$	ozn.
$\omega : \omega _a \equiv \omega _b(mod3)$	reg	mini	
$\omega = xy y^R \wedge y \neq \varepsilon$	reg	2 obok	
$\omega : palindrom \wedge \omega _a = \omega _c$	LOP	$a^nc^nc^na^n$	
$\omega = xcycz \wedge xy \text{ i } yz \in \{a,b\}^*$ palindromy	Ogd	$a^mbca^mcb a^m$	śr. ozn.
$ \omega _a = \omega _b$	bezk.		
$ \omega _a = \omega _b = \omega _c$	LOP	$a^nb^nc^n$	
$\omega : \omega _a \neq \omega _b \neq \omega _c$	Ogd	$a^{m+m!}b^ma^{m+m!}$	ozn b.
$\omega : \omega _a = \omega _b = \omega _c$	LOP	$a^nb^nc^n$	i=0
$\omega : \omega _a = \omega _c > \omega _b$	LOP	$a^{n+1}b^nc^{n+1}$	
$\omega \omega \omega$	LOP	$0^n1^n0^n1^n0^n1^n$	i=0
$\omega \omega^R \omega$	LOP	$0^n1^n1^n0^n0^n1^n$	i=0
$a^nc^kb^n : n \neq k$	Ogd	$a^{n!+n}c^nb^{n!+n}$	