

Obliczalność i złożoność

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. zimowy 2025/26

`inf.ug.edu.pl/~amb/oblicz`

Andrzej M. Borzyszkowski (Instytut Informatyki) Obliczalność i złożoność sem. zimowy 2025/26 1 / 14

Wykład 02 Gramatyki

Gramatyka, przykład

- Wyrażenie $\rightarrow$ Liczba | Wyrażenie Operator Wyrażenie
Operator $\rightarrow + | - | * | \div$
Liczba $\rightarrow 2 | 3 | 5$ (to tylko przykłady liczb)
- możemy wyprowadzić przykładowe wyrażenie
Wyrażenie $\rightarrow$ Wyrażenie Operator Wyrażenie $\Rightarrow$
Liczba Operator Wyrażenie $\Rightarrow$
2 Operator Wyrażenie $\Rightarrow$
2 + Wyrażenie $\Rightarrow$
2 + Wyrażenie Operator Wyrażenie $\Rightarrow$
2 + Liczba Operator Wyrażenie $\Rightarrow$
2 + 3 Operator Wyrażenie $\Rightarrow$
2 + 3 * Wyrażenie $\Rightarrow$
2 + 3 * Liczba $\Rightarrow$
2 + 3 * 5

Andrzej M. Borzyszkowski (Instytut Informatyki) Obliczalność i złożoność sem. zimowy 2025/26 3 / 14

Gramatyki. Języki bezkontekstowe. Automaty ze stosem.

Andrzej M. Borzyszkowski (Instytut Informatyki) Obliczalność i złożoność sem. zimowy 2025/26 2 / 14

Wykład 02 Gramatyki

Gramatyka, definicja

- Gramatyka (bezkontekstowa): $\langle V, T, P, S \rangle$, gdzie
 - $S \in V$ jest wyróżnionym symbolem początkowym
 - V jest zbiorem (skończonym) symboli nieterminalnych (zmienne)
 - T jest alfabetem, t.j. skończonym zbiorem symboli terminalnych, stałych, $V \cap T = \emptyset$
 - $P \subseteq V \times (V \cup T)^*$ jest skończonym zbiorem produkcji zapisywanych jako $v \rightarrow w$ gdy $\langle v, w \rangle \in P$
gdy $\langle v, w_1 \rangle \in P, \dots, \langle v, w_n \rangle \in P$ możemy pisać $v \rightarrow w_1 | \dots | w_n$
- np. $V = \{W, O, L\}$, $S = W$, $T = \{+, -, *, \div, 2, 3, 5\}$ i produkcje
 $W \rightarrow L | W O W$, $O \rightarrow \{+, -, *, \div\}$, $L \rightarrow \{2, 3, 5\}$
- elementów $\langle V, T, S \rangle$ można nie wypisywać, wiadomo, że symbole nieterminalne są po lewej stronie produkcji, pozostałe są terminalne, pierwszy jest początkowy.

Andrzej M. Borzyszkowski (Instytut Informatyki) Obliczalność i złożoność sem. zimowy 2025/26 4 / 14

Gramatyka, wywód słowa

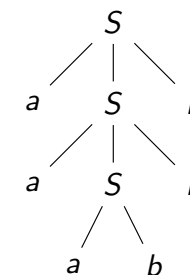
- jeśli $A \rightarrow v$ jest produkcją w gramatyce, to $uAw \Rightarrow uvw$ jest krokiem w wywodzie słowa uvw ze słowa uAw
- wywody zaczynają się od symbolu początkowego, $S \Rightarrow^* w$, $w \in (V \cup T)^*$ oznacza, że $S \Rightarrow w_1 \Rightarrow \dots \Rightarrow w_n = w$
- uwaga: zero kroków też jest dopuszczalne $S \Rightarrow^* S$
- docelowo chcemy wywieść słowa tylko z symboli terminalnych
- def.: język *definiowany przez gramatykę* $L(G) = \{w \in T^* \mid S \Rightarrow^* w\}$
- def.: język *bezkontekstowy* to język generowany przez jakąś gramatykę
- w powyższym przykładzie Wyrażenie $\Rightarrow^* 2 + 3 * 5$
- inny przykład $S \rightarrow ab|aSb$
- język generowany $\{a^n b^n \mid n > 0\}$, nie jest regularny

Gramatyki jednoznaczne

- def. gramatyka jest *jednoznaczna*, jeśli każde słowo generowane przez tę gramatykę ma tylko jeden wywód lewostronny
- gramatyka poprzednia nie jest jednoznaczna
- można jednak podać równoważną gramatykę jednoznaczną
- np. wymusić by działania były wykonywane od razu od lewej (tak jak na prostym kalkulatorze)
- albo by mnożenie miało wyższą siłę wiązania (tak jak na ogół przyjmujemy)
- istnieją języki bezkontekstowe nie posiadające jednoznacznej gramatyki

Drzewo wyvodu

- $S \rightarrow ab|aSb$
drzewo wyvodu słowa $aaabbb$
 $S \rightarrow aSb \Rightarrow aaSbb \Rightarrow aaabbb$



- na ogół mamy możliwość rozwijania w danym kroku różnych symboli terminalnych
- tw.: zawsze można stosować wywód do symbolu nieterminalnego najbardziej z lewej
- $W \rightarrow L|WOW$, $O \rightarrow \{+, -, *, \div\}$, $L \rightarrow \{2, 3, 5\}$
- mamy dwa wywody słowa $2 + 3 * 5$
 $W \rightarrow WOW \Rightarrow LOW \Rightarrow 2 + W \Rightarrow 2 + WOW \Rightarrow 2 + LOW \Rightarrow 2 + 3 * L \Rightarrow 2 + 3 * 5$
- oraz $W \rightarrow WOW \Rightarrow WOWOW \Rightarrow LOWOW \Rightarrow 2 + LOW \Rightarrow 2 + 3 * W \Rightarrow 2 + 3 * L \Rightarrow 2 + 3 * 5$

Postać normalna Chomsky'ego

- gramatyka $\langle V, T, P, S \rangle$ jest w postaci normalnej Chomsky'ego jeśli wszystkie produkcje są jednej z postaci
 - $A \rightarrow a$, gdzie $A \in V$, $a \in T$
 - $A \rightarrow BC$, gdzie $B, C \in V \setminus S$
 - $S \rightarrow \varepsilon$
- w szczególności symbol końcowy nie występuje po prawej stronie
- tw.: dla każdej gramatyki bezkontekstowej istnieje równoważna w postaci normalnej Chomsky'ego

Postać normalna Greibach

- gramatyka $\langle V, T, P, S \rangle$ jest w postaci normalnej Greibach jeśli wszystkie produkcje są jednej z postaci
 - $A \rightarrow aA_1 \dots A_n$, gdzie $a \in T$, $A_1, \dots, A_n \in V \setminus S$, $n \geq 0$
 - $S \rightarrow \varepsilon$
 w szczególności symbol końcowy nie występuje po prawej stronie
- tw.: dla każdej gramatyki bezkontekstowej istnieje równoważna w postaci normalnej Greibach

Przykład $\{a^n b^n \mid n \in \mathbb{N}\}$

- w stanie początkowym czytamy serię liter a , każda litera odkładana jest na stos
pierwsza odczytana litera b zmienia stan, w tym stanie można odczytać jedynie litery b , każdy odczyt zdejmuje literę a ze stosu
- trójki stan, słowo, stos zmieniają się następująco:
 - $\langle q_0, abb, Z \rangle \Rightarrow \langle q_0, bb, a \rangle \Rightarrow \langle q_1, b, \varepsilon \rangle$
odczyt kolejnej litery jest niemożliwy, bo stos jest pusty
 - $\langle q_0, aab, Z \rangle \Rightarrow \langle q_0, ab, a \rangle \Rightarrow \langle q_0, b, aa \rangle \Rightarrow \langle q_1, \varepsilon, a \rangle$
słowo zostało odczytane do końca ale stos pozostał niepusty
 - $\langle q_0, aabb, Z \rangle \Rightarrow \langle q_0, abb, a \rangle \Rightarrow \langle q_0, bb, aa \rangle \Rightarrow \langle q_1, b, a \rangle \Rightarrow \langle q_1, \varepsilon, \varepsilon \rangle$
słowo zostało odczytane do końca i stos jest pusty – zostało zaakceptowane

Automaty ze stosem

- automat ze stosem jest to automat (być może niedeterministyczny, z cichymi przejściami) zaopatrzony w dodatkową pamięć w postaci stosu, na który kładzione są pewne elementy
- w każdym ruchu automat odczytuje symbol z taśmy wejściowej i pobiera element z wierzchu stosu, następnie zmienia stan i kładzie na stos ciąg elementów
- w danym stanie może być możliwy ruch zależny tylko od wierzchu stosu, bez czytania taśmy wejściowej
- automat zaczyna działanie w stanie początkowym z pustym stosem (jego pustość jest widoczna specjalnym symbolem Z)
- definicje akceptowania:
 - stany akceptujące, jak dla zwykłych automatów
 - akceptowane są słowa w momencie gdy stos jest pusty
- tw.: dla automatu akceptującego w/g jednej z definicji istnieje inny akceptujący w/g drugiej

Gramatyki bezkontekstowe a automaty ze stosem

- Tw.: języki bezkontekstowe są to dokładnie języki akceptowane przez automaty ze stosem
- dowód złożony, poza treścią tego wykładu
- konsekwencje dla bezpieczeństwa aplikacji

Lemat o pompowaniu

- Tw.: jeśli L jest językiem bezkontekstowym, to istnieje stała n taka, że dla każde słowo $z \in L$ dłuższe niż n posiada rozkład $z = uvwxy$ taki, że
 - $|vwx| \leq n$
 - $|v| + |x| \geq 1$
 - $uv^\ell wx^\ell y \in L$ dla wszystkich $\ell \in \mathbb{N}$
- przykład zastosowania: język $\{a^n b^n c^n \mid n \in \mathbb{N}\}$ nie jest językiem bezkontekstowym
 albo fragment pompowany jest na granicy liter i po napompowaniu będzie zaburzona kolejność,
 albo pompowane będą jednorodne fragmenty, ale tylko dwa,
 a w słowach są trzy

Operacje na językach bezkontekstowych

- tw.: klasa języków bezkontekstowych jest zamknięta na sumę, konkatenację i iterację Kleene'ego
- iteracja: symbol S generuje język L , dodajemy nowy symbol S' i nową produkcję $S' \rightarrow \varepsilon \mid SS'$
- konkatenacja i suma podobnie
- klasa języków bezkontekstowych nie jest zamknięta na przekrój
 - świadczył o tym przykład
 $\{a^n b^n c^n \mid n \in \mathbb{N}\} = \{a^n b^n c^k \mid n, k \in \mathbb{N}\} \cap \{a^k b^n c^n \mid n, k \in \mathbb{N}\}$
 ani na uzupełnienie
- przekrój i różnica jest językiem bezkontekstowym jeśli jeden z języków jest regularny
 - dw.: rozpatrujemy iloczyn kartezjański automatów, jeden z nich ma dodatkowy stos, ale drugi nie potrzebuje stosu