

Теория автоматов и формальных языков

Введение

Лектор: Екатерина Вербицкая

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»

6 сентября 2016г.

- Естественные
 - ▶ Русский, английский...

- Естественные
 - ▶ Русский, английский...
- Искусственные
 - ▶ Эсперанто, ложбан...
 - ▶ Клингонский, эльфийский...

- Естественные
 - ▶ Русский, английский...
- Искусственные
 - ▶ Эсперанто, ложбан...
 - ▶ Клингонский, эльфийский...
 - ▶ C++, Java, C#, Haskell, OCaml, Perl, Coq, Agda...

- Текстовые редакторы
- Компиляторы, интерпретаторы, трансляторы
- Среды разработки
- Все нуждаются в некотором формализованном представлении языка

- Синтаксис — правила построения программ из символов
- Семантика — правила истолкования программ, определяющие их смысл

Пример: язык арифметических выражений

- Алфавит символов: цифры, скобки, знаки арифметических операций ($+$, $-$, $*$, $/$)
- Синтаксис
 - ▶ **Терм**: последовательность цифр или любое **выражение** в скобках
 - ▶ **Слагаемое**: последовательность **термов**, соединенных знаками умножения и деления
 - ▶ **Выражение**: последовательность **слагаемых**, соединенных знаками сложения и вычитания (перед первым **слагаемым** может стоять минус)
- Семантика
 - ▶ Значение выражения

- Язык, на котором дано описание языка
 - ▶ Естественный язык

- Язык, на котором дано описание языка
 - ▶ Естественный язык
 - ▶ Язык металингвистических формул Бэкуса (БНФ)

- Язык, на котором дано описание языка
 - ▶ Естественный язык
 - ▶ Язык металингвистических формул Бэкуса (БНФ)
 - ▶ Синтаксические диаграммы

- Язык, на котором дано описание языка
 - ▶ Естественный язык
 - ▶ Язык металингвистических формул Бэкуса (БНФ)
 - ▶ Синтаксические диаграммы
 - ▶ Грамматики...

- **Алфавит** — конечное множество символов

- ▶ $\{a, b, c, \dots, z\}$
- ▶ $\{\alpha, \beta, \gamma, \dots, \omega\}$
- ▶ $\{0, 1\}$
- ▶ $\{\text{let}, \text{in}, \text{where}, \dots\}$

- **Цепочка (предложение, слово)** — любая конечная последовательность символов алфавита
 - ▶ cat
 - ▶ KAT
 - ▶ 011000110110000101110100
 - ▶ `main = putStrLn . show . inc 2 where inc = \x -> x + 1`
- **Пустая цепочка ε** — цепочка, не содержащая ни одного символа
 - ▶ ε не является символом алфавита

- **Конкатенация строк α и β ($\alpha \cdot \beta = \alpha\beta$)** — результат приписывания строки β в конец строки α
 - ▶ $\forall \alpha \beta \gamma. (\alpha \cdot \beta) \cdot \gamma = \alpha \cdot (\beta \cdot \gamma)$
 - ▶ $\forall \alpha. \alpha \cdot \varepsilon = \varepsilon \cdot \alpha = \alpha$

БНФ — Бэкуса-Наура форма

- **Символ** — элементарное понятие языка
 - ▶ $+$ означает сложение в языке арифметических выражений
- **Метапеременная** — сложное понятие языка
 - ▶ Переменной $\langle \text{выражение} \rangle$ можно обозначить выражение
- **Формула**
 - ▶ $\langle \text{определяемый символ} \rangle ::= \langle \text{посл.1} \rangle \mid \dots \mid \langle \text{посл.}n \rangle$
 - ▶ В правой части формулы — альтернатива конкатенаций строк, составленных из символов и метапеременных
- **Пример: число**
 - ▶ $\langle \text{число} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{цифра} \rangle \langle \text{число} \rangle$

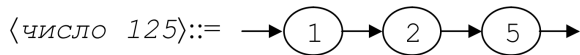
Расширенная форма Бэкуса Наура (EBNF)

- Более емкие операции
- **Итерация**
 - ▶ $\langle x \rangle ::= \{ \langle y \rangle \}$ эквивалентно: $\langle x \rangle ::= \varepsilon \mid \langle y \rangle \langle x \rangle$
- **Условное вхождение**
 - ▶ $\langle x \rangle ::= [\langle y \rangle]$ эквивалентно: $\langle x \rangle ::= \varepsilon \mid \langle y \rangle$
- Скобки для группировки
 - ▶ $(\langle x \rangle \mid \langle y \rangle) \langle z \rangle$ эквивалентно: $\langle x \rangle \langle z \rangle \mid \langle y \rangle \langle z \rangle$

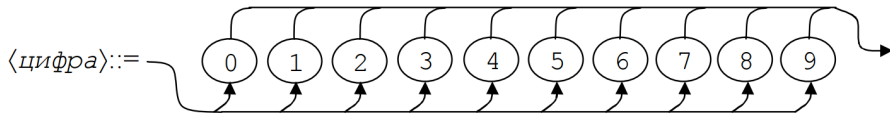
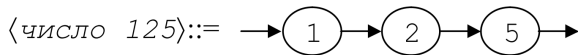
Пример: арифметические выражения

$$\begin{aligned} \langle \textit{expr} \rangle &::= [-] \langle \textit{factor} \rangle \{ \langle + - \rangle \langle \textit{factor} \rangle \} \\ \langle + - \rangle &::= + \mid - \\ \langle \textit{factor} \rangle &::= \langle \textit{term} \rangle \{ \langle * / \rangle \langle \textit{term} \rangle \} \\ \langle * / \rangle &::= * \mid / \\ \langle \textit{term} \rangle &::= \langle \textit{number} \rangle \mid (\langle \textit{expr} \rangle) \end{aligned}$$

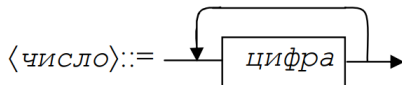
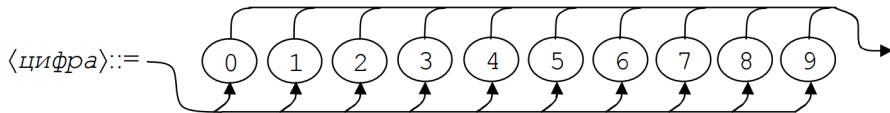
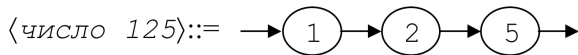
Синтаксические диаграммы Вирта



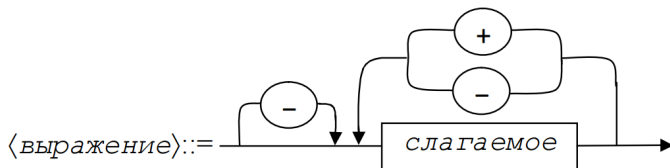
Синтаксические диаграммы Вирта



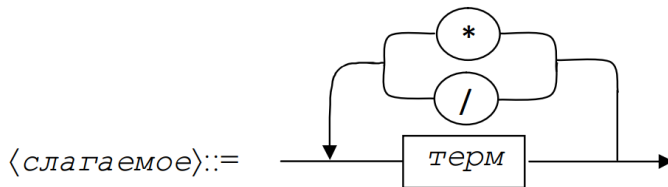
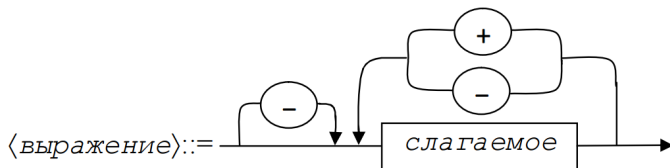
Синтаксические диаграммы Вирта



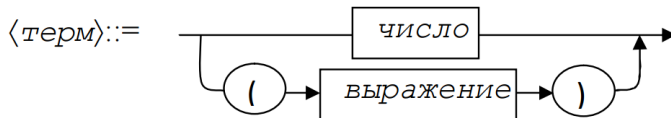
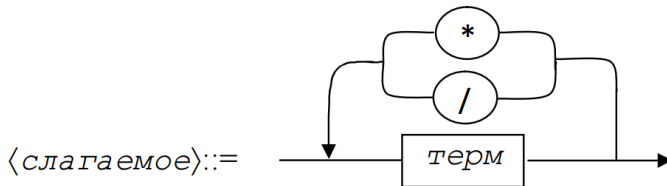
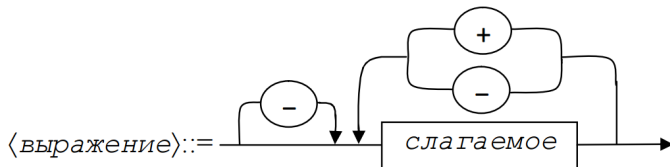
Синтаксические диаграммы Вирта



Синтаксические диаграммы Вирта



Синтаксические диаграммы Вирта



Операции над строками

- **Обращение (реверс) цепочки** a^R — цепочка, символы которой записаны в обратном порядке
 - ▶ Если $x = abc$, $x^R = cba$
 - ▶ $\varepsilon^R = \varepsilon$
- **n -я степень цепочки** a^n — конкатенация n повторений цепочки
 - ▶ $a^0 = \varepsilon$
 - ▶ $a^n = a \cdot a^{n-1} = a^{n-1} \cdot a$
- **Длина цепочки** $|a|$ — количество составляющих ее символов
 - ▶ $|babb| = 4$, $|babb|_a = 1$, $|babb|_b = 3$, $|babb|_c = 0$
 - ▶ $|\varepsilon| = 0$

- V — алфавит
 - ▶ $V = \{0, 1\}$
- V^* — множество, содержащее все цепочки в алфавите V , включая пустую цепочку
 - ▶ $V^* = \{\varepsilon, 0, 1, 00, 11, 01, 10, 000, 001, 011, \dots\}$
- $V^+ = V^* \setminus \{\varepsilon\}$
 - ▶ $V^+ = \{0, 1, 00, 11, 01, 10, 000, 001, 011, \dots\}$
- V — подмножество множества всех цепочек в этом алфавите.
 - ▶ Для любого языка L справедливо $L \subseteq V^*$

- Порождающая грамматика G — это четверка $\langle V_T, V_N, P, S \rangle$
 - ▶ V_T — алфавит терминальных символов (терминалов)
 - ▶ V_N — алфавит нетерминальных символов (нетерминалов)
 - ★ $V_T \cap V_N = \emptyset$
 - ★ $V ::= V_T \cup V_N$
 - ▶ P — конечное множество правил вида $\alpha \rightarrow \beta$
 - ★ $\alpha \in V^* V_N V^*$
 - ★ $\beta \in V^*$
 - ▶ S — начальный нетерминал грамматики, $S \in V_N$

Пример: язык чисел в двоичной системе счисления

$$V_T = \{0, 1\}; V_N = \{S, N, A\}$$

$$S \rightarrow 0$$

$$S \rightarrow N$$

$$S \rightarrow -N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A$$

$$A \rightarrow 1A$$

$$A \rightarrow \varepsilon$$

Пример: язык чисел в двоичной системе счисления

$$V_T = \{0, 1\}; V_N = \{S, N, A\}$$

$$S \rightarrow 0$$

$$S \rightarrow N$$

$$S \rightarrow -N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A$$

$$A \rightarrow 1A$$

$$A \rightarrow \varepsilon$$

$$S \rightarrow 0|N| - N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A|1A|\varepsilon$$

Пример: язык чисел в двоичной системе счисления

$$V_T = \{0, 1\}; V_N = \{S, N, A\}$$

$$S \rightarrow 0$$

$$S \rightarrow N$$

$$S \rightarrow -N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A$$

$$A \rightarrow 1A$$

$$A \rightarrow \varepsilon$$

$$S \rightarrow 0|N| - N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A|1A|\varepsilon$$

$$S \rightarrow 0|[-]N$$

$$N \rightarrow 1A$$

$$A \rightarrow (0|1)A|\varepsilon$$

Отношение непосредственной выводимости

- $\alpha \rightarrow \beta \in P$
- $\gamma, \delta \in V^*$
- $\gamma\alpha\delta \Rightarrow \gamma\beta\delta$: $\gamma\beta\delta$ непосредственно выводится из $\gamma\alpha\delta$ при помощи правила $\alpha \rightarrow \beta$

Отношение выводимости

- $a_0, a_1, a_2, \dots, a_n \in V^*$
- $a_0 \Rightarrow a_1 \Rightarrow a_2 \Rightarrow \dots \Rightarrow a_n$
- $a_0 \xRightarrow{*} a_n$: a_n **выводится** из a_0
- $S \Rightarrow -N \Rightarrow -1A \Rightarrow -11A \xRightarrow{*} -1101A \Rightarrow -1101$

Отношение выводимости

- $a_0, a_1, a_2, \dots, a_n \in V^*$
- $a_0 \Rightarrow a_1 \Rightarrow a_2 \Rightarrow \dots \Rightarrow a_n$
- $a_0 \xRightarrow{*} a_n$: a_n **выводится** из a_0
- $S \Rightarrow -N \Rightarrow -1A \Rightarrow -11A \xRightarrow{*} -1101A \Rightarrow -1101$
- $\forall a \in V^*. a \xRightarrow{*} a$
- $a_0 \xRightarrow{+} a_n$: вывод использует хотя бы одно правило грамматики
- $a_0 \xRightarrow{k} a_n$: вывод происходит за k шагов

Язык, порождаемый грамматикой $G = \langle V_T, V_N, P, S \rangle$

- $L(G) = \{\omega \in V_T^* \mid S \xRightarrow{*} \omega\}$

- Грамматики G_1 и G_2 эквивалентны, если $L(G_1) = L(G_2)$

- Грамматика G_1 и G_2 эквивалентны, если $L(G_1) = L(G_2)$

$$V_T = \{0, 1\}$$

$$V_N = \{S, N, A\}$$

$$S \rightarrow 0|N| - N$$

$$N \rightarrow 1A$$

$$A \rightarrow 0A|1A|\varepsilon$$

- Грамматика G_1 и G_2 эквивалентны, если $L(G_1) = L(G_2)$

$$\begin{aligned}V_T &= \{0, 1\} \\ V_N &= \{S, N, A\}\end{aligned}$$

$$\begin{aligned}S &\rightarrow 0|N| - N \\ N &\rightarrow 1A \\ A &\rightarrow 0A|1A|\varepsilon\end{aligned}$$

$$\begin{aligned}V_T &= \{0, 1\} \\ V_N &= \{S, A\}\end{aligned}$$

$$\begin{aligned}S &\rightarrow 0|1A| - 1A \\ A &\rightarrow 0A|1A|\varepsilon\end{aligned}$$

- **Контекстно-свободная грамматика** — грамматика, все правила которой имеют вид $A \rightarrow \alpha, A \in V_N, \alpha \in V^*$

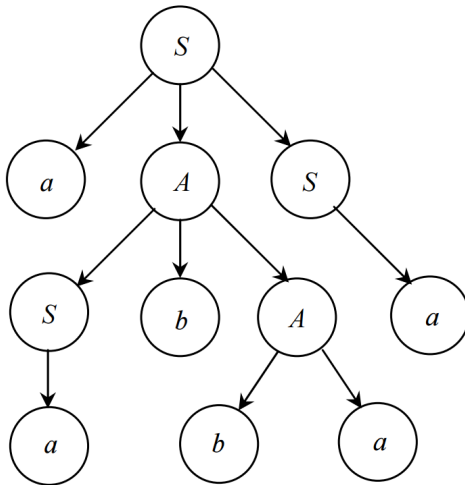
Дерево является **деревом вывода** для $G = \langle V_N, V_T, P, S \rangle$, если:

- Каждый узел помечен символом из алфавита V
- Метка корня — S
- Листья помечены терминалами, остальные узлы — нетерминалами
- Если узлы n_0, \dots, n_k — прямые потомки узла n , перечисленные слева направо, с метками A_0, \dots, A_k ; метка n — A , то $A \rightarrow A_0 \dots A_k \in P$

Пример дерева вывода

$G = \langle \{S, A\}, \{a, b\}, \{S \rightarrow aAS \mid a, A \rightarrow SbA \mid ba \mid SS\}, S \rangle$

$S \Rightarrow aAS \Rightarrow aSbAS \Rightarrow aabAS \Rightarrow aabbaS \Rightarrow aabbaa$



Теорема

Пусть $G = \langle V_N, V_T, P, S \rangle$ — КС-грамматика

Вывод $S \xRightarrow{*} \alpha$, где $\alpha \in V^*$, $\alpha \neq \varepsilon$ существует \Leftrightarrow существует дерево вывода в грамматике G с результатом α