Занятие 6

Порождающие грамматики Пусть заданы два непересекающихся (конечных) алфавита: Σ (основной, терминалные символы) и Ω (служебный, нетерминальные символы или грамматические категории). *Порождающая грамматика* задается четверкой $G = \langle \Sigma, \Omega, \Pi, S \rangle$, где $S \in \Omega$ — начальный символ, а Π — конечный набор *замен* (редукций) вида $u \to v$ для некоторых $u, v \in (\Sigma \cup \Omega)^*$. Применение редукции $u \to v$ к слову w сводится к замене одного из вхождений слова u в w на v (если слово u не входит в w, то редукция неприменима). Соответствующее правило вывода задается схемой:

$$\frac{xuy}{xvy} \qquad x, y \in (\Sigma \cup \Omega)^*.$$

.

Выводом слова w в грамматике G называется такая конечная последовательность слов $w_0, w_1, \ldots, w_n$, $\; w_i \in (\Sigma \cup \Omega)^*$, для которой $w_0 = S$, $w_n = w$, а каждое слово w_{i+1} получается из w_i применением одной из редукций. Вывод слова w называется *порождением слова* w, если $w \in \Sigma^*$, т.е. w не содержит служебных символов.

Произвольное подмножество множества Σ^* всех слов в алфавите Σ называется *формальным языком.* Множество всех слов, порожденных грамматикой G, образует формальный язык $L(G) \subseteq \Sigma^*$, порожденный грамматикой G.

1. Отождествим натуральные числа с их унарными записями: $0 = \Lambda,\ 1 = |,\ 2 = ||,\ \ldots$. Построить грамматику, порождающую а) натуральный ряд; б) все слова в алфавите $\{a, b, c\}$; с) все двоичные записи натуральных чисел.

   a) $\begin{array}{lll} S & \to & \\ S & \to & |S \end{array}$ b) $\begin{array}{lll} S & \to & \\ S & \to & aS \\ S & \to & bS \\ S & \to & cS \end{array}$ c) $\begin{array}{lll} S & \to & 0 \\ S & \to & 1A \\ A & \to & 0A \\ A & \to & 1A \\ A & \to & \end{array}$

2. Какой язык порождается грамматикой $\{S \to aSb\,; S \to \}$? Построить грамматику, порождающую все слова в алфавите $\{a, b\}$, содержащие одинаковое количество букв a и b.

Общее понятие исчисления обобщает понятие грамматики в следующих направлениях: 1) разрешаются правила с несколькими посылками и они не обязательно сводятся к заменам; 2) проверка завершенности вывода — общего вида, т.е. не обязательно сводится к проверке отсутствия служебных символов (их вообще может не быть).

Исчисление в алфавите Σ задается множествами R и F, допускающими алгоритмическую проверку принадлежности (далее просто проверки):

- R состоит из пар вида $\langle \Gamma, v\rangle$, где Γ — конечное множество слов в алфавите Σ, а v — слово в алфавите Σ. Пара соответствует правилу вывода $\Gamma \vdash v$, а R перечисляет все правила, допустимые для данного исчисления. Случай $\Gamma = \emptyset$ соответствует аксиомам. Заметим, что в логике вместо строчной записи правила $\langle \{v_1, \dots, v_n\}, v\rangle$ часто используют фигуру
$$\frac{v_1, \dots, v_n}{v}\,.$$
- $F \subseteq \Sigma^*$ состоит из тех слов, на которых вывод можно закончить результативно.

При этом существенно, что есть алгоритмы, которые правильно отвечают на вопросы «$\langle \Gamma, v\rangle \in R$?» и «$v \in R$?» для произвольных Γ и v.

Выводом в исчислении называется любая конечная последовательность слов $v_0, \dots, v_n$, в которой каждый ее член v_i получен из предыдущих по одному из правил $\langle \Gamma, v\rangle \in R$, т.е. $\Gamma \subseteq \{v_j \mid j < i\}$ и $v = v_i$. Вывод $v_0, \dots, v_n$ *порождает* слово $v = v_n$, если $v_n \in F$. (Если $v_n \notin F$, то вывод выводит слово v_n, но ничего не порождает.)

Исчисление порождает множество (язык), состоящее из всех слов, порожденных его выводами. Такие множества названы *«породимыми»*.

3. Представить грамматику в виде исчисления общего вида, т.е. определить алфавит исчисления Σ и множества R, F.

4. а) Представить в формате исчисления определение замкнутых термов сигнатуры $\{a, f^2, g^1\}$. б) Построить грамматику, порождающую множество всех таких термов.

   а) Алфавит Σ состоит из сигнатурных символов, а также скобок и запятой. К множеству R отнесем пару $\langle \emptyset, a\rangle$, а также все пары видов

$\langle \{x;y\}, f(x,y) \rangle$ и $\langle \{x\}, g(x) \rangle$, где $x, y \in \Sigma^*$. Условие завершения тривиально: $F = \Sigma^*$. Более привычная запись правил:

$$\frac{}{a} \qquad \frac{x \quad y}{f(x,y)} \qquad \frac{x}{g(x)} \qquad (x, y \in \Sigma^*)$$

б) $S \rightarrow a$
$S \rightarrow g(S)$
$S \rightarrow f(S,S)$

5. Условимся записывать переменные $x_0, x_1, \ldots$ словами в алфавите $\{x, |\}$ так: $x_0 = x$, $x_1 = x|$, $x_2 = x||$, $\ldots$ Доказать породимость множества всех термов, составленных из этих переменных и функциональных символов $f(x,y)$ и $g(x)$. Построить грамматику, порождающую это множество. Можно ли для этого множества предложить тривиальное исчисление, все правила которого — аксиомы (т.е. не имеют посылок)?

6. Доказать что объединение и пересечение двух породимых множеств — породимо.

Домашнее задание

7. Построить грамматику, порождающую все двоичные дроби (конечные, со знаком, с дробной частью и без).

8. Построить грамматику, порождающую всевозможные пары вида $\langle$натуральное число$\rangle \# \langle$его остаток от деления на 2$\rangle$. Использовать унарную запись.

9. Построить грамматику, порождающую все арифметические выражения, правильно построенные из натуральных чисел с помощью операций $+$, $*$ и скобок. Правила опускания скобок не учитывать. Числа представлять в двоичной записи.

10. Доказать породимость множества всех формул логики высказываний. Переменные записывать словами в алфавите $\{x, |\}$ как в задаче 5.

11. Доказать породимость множества всех модальных формул, выводимых в логике **K**. Тот же вопрос для логики отношений. Можно ли здесь обойтись тривиальным исчислением? (Обсудить.)

12. Грамматика называется бесконтекстной, если все ее замены имеют вид $A \to v$, где A — служебная буква. Доказать, что язык, порожденный бесконтекстной грамматикой, можно задать тривиальным исчислением общего вида, у которого все правила — аксиомы.