Синтаксический анализ. Построение МП-автомата

1. Напоминание.

Формальное определение МП-автомата

$M = \langle Q, V, Z, \delta, q_0, z_0, F \rangle$
Q – множество состояний;
V – алфавит входных символов;
Z – специальный алфавит магазинных символов;
δ – функция переходов автомата:

$$Q \times (V \cup \{\lambda\}) \times Z \to P(Q \times Z^*),$$

где $P(Q \times Z^*)$ – множество подмножеств $Q \times Z^*$;
$q_0 \in Q$ – начальное состояние автомата;
$z_0 \in Z$ – начальное состояние магазина (маркер дна);
$F \subseteq Q$ – множество конечных состояний.

2. Напоминание.

Работа автомата $M = \langle Q, V, Z, \delta, q_0, z_0, F \rangle$

1) состояние автомата $(q, a\alpha, z\beta)$
2) читает символ a, находящийся под головкой (сдвигает ленту);
3) не читает ничего (читает λ, не сдвигает ленту);
4) из функции δ определяет новое состояние q', если $(q', \gamma) \in \delta(q, a, z)$ или $(q', \gamma) \in \delta(q, \lambda, z)$.
5) читает верхний (в стеке) символ z и записывает цепочку γ т.к. $(q', \gamma) \in \delta(q, a, z)$, при этом, если $\gamma = \lambda$, то верхний символ магазина просто удаляется.
6) работа автомата заканчивается (q, λ, λ)

3. Напоминание:

на каждом шаге автомата возможны три случая:

1) функция $\delta(q, a, z)$ определена – осуществляется переход в новое состояние;
2) функция $\delta(q, a, z)$ не определена, но определена $\delta(q, \lambda, z)$ – осуществляется переход в новое состояние (лента не продвигается);
3) функции $\delta(q, a, z)$ и $\delta(q, \lambda, z)$ не определены – дальнейшая работа автомата не возможна (цепочка не разобрана).

4. Напоминание:

язык $L(M) = \{\alpha \mid (q_0, \alpha, z_0) \succ^* (q', \lambda, \lambda), q' \in F\}$ допускаемый автоматом M – это множество всех цепочек символов, допускаемых данным автоматом.

МП-автомат называется детерминированным, если из каждой его конфигурации возможно не более одного перехода в следующую конфигурацию. Иначе МП-автомат называется недетерминированным.

5. Для построения МП-автомата необходимо привести контекстно-свободную грамматику к одной из нормальных форм:

нормальной форме Хомского

нормальной форме Грейбах.

6. Нормальная форма Хомского:

КС-грамматика $G = \langle T, N, P, S \rangle$ имеет нормальную форму Хомского, если правила P имеют вид:

1) $A \to BC$, где $A, B, C \in N$;
2) $A \to a$, где $A \in N, a \in T$;
3) $S \to \lambda$, где $S \in N$ – начальный символ, и если такое правило существует, то нетерминал S не должен встречаться в правой части правил.

7. Грамматика в нормальной форме Хомского называется **бинарной**, т.к. один нетерминальный символ может быть заменен на два нетерминальных символа.

В дереве вывода грамматики в нормальной форме Хомского каждая вершина распадается:

- на **две другие вершины** (в соответствии с первым видом правил $A \to BC$),
- либо содержит **один последующий лист** с терминальным символом (в соответствии со вторым видом правилом вывода $A \to a$).

Третий вид правил введен для того, чтобы к нормальной форме Хомского можно было преобразовывать грамматики КС-языков, содержащих пустые цепочки символов.

8. Алгоритм преобразования контекстно-свободной грамматики $G = \langle T, N, P, S \rangle$ к грамматике $G' = \langle T, N', P', S \rangle$ в нормальной форме Хомского:

I. преобразовать исходную грамматику к приведенному виду (исключить бесплодные и недостижимые символы, цепные и λ-правила);

II. установить $N' = N$

III. построение P'.

Правила вида:

1) $A \to a$, где $A \in N, a \in T$ переносятся P' без изменений;
2) $A \to BC$, где $A, B, C \in N$ переносятся в P' без изменений;
3) $S \to \lambda$, где $S \in N$ переносятся в P' без изменений;
4) $A \to aB$, где $A, B \in N$, $a \in T$
   преобразуются в правила вида $A \to DB$ и $D \to a$ добавляются во множество правил P', нетерминальный символ D добавляется во множество нетерминалов N' грамматики G';
5) правила вида $A \to Ba$, где $A, B \in N$, $a \in T$
   преобразуются в правила вида $A \to BD$ и $D \to a$ грамматики P', нетерминальный символ D добавляется в N';
6) правила вида $A \to ab$, где $A \in N$, $a, b \in T$
   преобразуются в правила вида $A \to BD, B \to a$, $D \to b$ грамматики P', нетерминальные символы B, D добавляется в N';
7) правила вида $A \to X_1 X_2 ... X_k$, где $k > 2$ и $X_i \in N \cup T$
   преобразуются в правила вида $A \to Y_1 Y_2$, $Y_2 \to Y_3 Y_4$, $Y_4 \to Y_5 Y_6$, …, $Y_1 \to X_1$, $Y_3 \to X_2$, $Y_5 \to X_3$… Правила вида $Y_i \to X_j$ могут потребовать дальнейшего преобразования. Если достигнут вид правил, который указан в определении, то они добавляются в P', а новые нетерминальные символы в N'.

Стартовым символом результирующей грамматики G' является стартовый символ исходной грамматики G.

9. Пример: $G = \langle \{a,b\}, \{S,A,B\}, P, S \rangle$,

где $P = \{S \to bA \mid aB, \quad A \to a \mid aS \mid bAA, \quad B \to b \mid bS \mid aBB\}$

$P' = \{A \to a, B \to b\}$	правила типа 1 (переносятся в P')
$S \to bA : S \to CA, C \to b$	правила типа 4 (преобразуются, добавляется новый нетерминал и переносятся в P')
$S \to aB : S \to DB, D \to a$	правила типа 4
$A \to aS : A \to ES, E \to a$	правила типа 4
$B \to bS : B \to FS, F \to b$	правила типа 4
$A \to bAA : A \to TU, T \to b, U \to AA$	преобразуются к правилам 2 типа и 4 типа
$B \to aBB : B \to XY, X \to a, Y \to BB$	

Множество правил грамматики, приведенной К нормальной форме Хомского G':

$P' = \{A \to a, B \to b, S \to CA, C \to b, S \to DB, D \to a,$
$A \to ES, E \to a, B \to FS, F \to b, A \to TU, T \to b, U \to AA,$
$B \to XY, X \to a, Y \to BB\}$

Множество нетерминальных символов приведенной грамматики G':

$\{S, A, B, C, D, E, F, T, U, X, Y\}$

10. ***Определение***. Праворекурсивное правило:

правило вида $A \to \alpha A$, где $\alpha \in (T \cup N)^*$, $A \in N$

11. ***Определение.*** Леворекурсивное правило:

правило вида $A \to A\alpha$, где $\alpha \in (T \cup N)^*$, $A \in N$

12. Для каждой грамматики $G = \langle T, N, P, S \rangle$, содержащей леворекурсивные правила можно построить грамматику $G' = \langle T, N', P', S \rangle$, не содержащую леворекурсивных правил.

13. Для каждой грамматики $G = \langle T, N, P, S \rangle$, содержащей правокурсивные правила можно построить грамматику $G' = \langle T, N', P', S \rangle$, не содержащую правокурсивных правил.

14. Нормальная форма Грейбах:

контекстно-свободная грамматика $G = \langle T, N, P, S \rangle$ имеет нормальную форму Грейбах, если она не леворекурсивная (не содержит леворекурсивных правил), а правила P имеют вид:

1) $A \to a\alpha$, где $a \in T, \alpha \in (N \cup T)^*$;
2) $S \to \lambda$, где $S \in N$ — начальный символ, и если такое правило существует, то нетерминал S не должен встречаться в правой части правил.

Эта нормальная форма называется по имени Шейлы Грейбах (Sheila Greibach), которая первой описала способ построения таких грамматик.

15. Алгоритм устранения левой рекурсии.

Пусть правила грамматики $G = \langle T, N, P, S \rangle$ имеют вид:

$$A \to A\alpha_1 \mid A\alpha_2 \mid \ldots \mid A\alpha_m \mid \beta_1 \mid \beta_2 \mid \ldots \mid \beta_n,$$

где $\alpha_i, \beta_i \in (T \cup N)^*$ и цепочки β_i не начинаются с нетерминала A.

Введем нетерминал B.

Тогда эквивалентные правила без левой рекурсии:

$$A \to \mid \beta_1 \mid \beta_2 \mid \ldots \mid \beta_n \mid \beta_1 B \mid \beta_2 B \mid \ldots \mid \beta_n B \mid$$

$$B \to \mid \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_m \mid \alpha_1 B \mid \alpha_2 B \mid \ldots \mid \alpha_m B \mid$$

16. ***Пример:***

пусть правила P грамматики G имеют вид:

$A \to A + A \mid x$

Грамматика G порождает цепочки вида:

$x, x+x, x+x+x, x+x+x+x, \ldots$

Преобразование. Введем нетерминал A'

$A \to x \mid xA'$

$A' \to +A \mid +AA'$

Приведенная грамматика порождает цепочки вида:

$x, x+x, x+x+x, x+x+x+x, \ldots$

17. ***Пример:***

пусть правила P грамматики имеют вид G :

$\underbrace{E}_{A} \to \underbrace{E}_{A}\underbrace{+T}_{\alpha_1} \mid \underbrace{T}_{\beta_1},\quad T \to T \times F \mid F,\quad F \to (E) \mid a$

Преобразование грамматики:

$\underbrace{E}_{A} \to \underbrace{E}_{A}\underbrace{+T}_{\alpha_1} \mid \underbrace{T}_{\beta_1} \Rightarrow E \to T \mid TE',\ \underbrace{E'}_{A'} \to \underbrace{+T}_{\alpha_1} \mid \underbrace{+T}_{\alpha_1}\underbrace{E'}_{A'}$

$T \to T \times F \mid F \Rightarrow T \to F \mid FT', T' \to F \mid \times FT'$

Правила P' грамматики G':

$E \to T \mid TE'$

$E' \to +T \mid +TE'$

$T \to F \mid FT'$

$T' \to \times F \mid \times FT'$

$F \to (E) \mid a$

18. Построение МП-автомата:

пусть дана контекстно-свободная грамматика $G = \langle T, N, P, S \rangle$

Автомат $M = \langle Q, V, Z, \delta, q_0, z_0, F \rangle$, где

$$Q = \{q_0\} \ ; \quad V = T \ ; \quad Z = N \cup T \cup z_0; \quad F = \{q_0\}$$

Для всех $A \in N$ в левой части правил (нетерминалов)

$\delta^0(q_0, \lambda, A) = (q_0, \alpha^R)$ (где α^R - реверс цепочки α) (1)

Для всех $a \in T$ (терминалов)

$\delta(q_0, a, a) = (q_0, \lambda)$ (2)

Для перехода в конечное состояние

z$\delta(q_0, \lambda, z_0) = (q_0, \lambda)$ (3)

19. Пример.

Пусть G — грамматика, порождающая слова над алфавитом $\{0,1\}$, в которых одинаковое количество нулей и единиц:

Правила грамматики:

$S \to 0S1$

$S \to 1S0$

$S \to \lambda$

Построить автомат, допускающий язык, порожденный грамматикой G.

Множество состояний: $\{q\}$

Множество терминалов: $\{0,1\}$

Множество нетерминалов — $\{S\}$

Магазинный алфавит: $\{0,1,S,z\}$

Маркер дна МП-автомата: z

Множество конечных состояний: $\{q\}$

Начальное состояние МП-автомата: q

Функция переходов δ определена следующим образом:

$\delta(q, \lambda, S) = \{(q, 0S1), (q, 1S0), (q, \lambda)\}$ (пункт 1 построения δ);

$\delta(q, 0{,}0) = \{(q, \lambda)\}$, $\delta(q, 1{,}1) = \{(q, \lambda)\}$ (пункт 2 построения δ)

$\delta(q, \lambda, z) = (q, \lambda)$ (пункт 3 построения δ)

Пример.

Дана цепочка 0011

Начальная конфигурация МП-автомата:

$(q, 0011, zS)$ в магазин – реверс цепочки правила $S \to 0S1$ (пункт 1)

Последовательность тактов работы построенного автомата:

$(q, 0011, z1S0)$	(пункт 2)
$(q, 011, z1S)$	(пункт 1)
$(q, 011, z11S0)$	(пункт 2)
$(q, 11, z11S)$	(пункт 1)
$(q, 11, z11)$	(пункт 2)
$(q, 1, z1)$	(пункт 2)
(q, λ, z)	(пункт 3)
(q, λ, λ)	цепочка разобрана

20. Пример.

Построить автомат, допускающий язык, порожденный грамматикой G с правилами:

$S \to aSbb \mid a.$

Преобразуем грамматику к нормальной форме Грейбах:

$S \to aSA \mid a,$
$A \to bB,$
$B \to b.$

Построить автомат, допускающий язык, порожденный грамматикой G. Автомат будет иметь два состояния:

$\{ q_1, q_2\}$, где q_1 — заключительное состояниеи и q_2 — заключительное состояние.

Множество терминалов: $\{a, b\}$

Множество нетерминалов — $\{S, A, B\}$.

z – маркер дна магазина

Магазинный алфавит: $\{a, b, S, A, B, z\}$.

$\delta(q_1, \lambda, S) = \{(q_1, ASa), (q_1, a)\}$ (пункт 1 построения δ);

$\delta(q_1, \lambda, A) = \{(q_1, Bb)\}$ (пункт 1 построения δ);

$\delta(q_1, \lambda, B) = \{(q_1, b)\}$ (пункт 1 построения δ);

$\delta(q_1, a, a) = \{(q_1, \lambda)\}$ (пункт 2 построения δ);

$\delta(q_1, b, b) = \{(q_1, \lambda)\}$ (пункт 2 построения δ)

$\delta(q_1, \lambda, z) = (q_2, \lambda)$ (пункт 3 построения δ)

Дана цепочка $aabb$

Начальная конфигурация:

$(q_1, aabb, zS)$

Последовательность тактов работы построенного автомата:

$(q_1, aabb, zASa)$

(q_1, abb, zAS)

(q_1, abb, zAa)

(q_1, bb, zA)

(q_1, bb, zBb)

(q_1, b, zB)

(q_1, b, zb)

(q_1, λ, z)

(q_2, λ, z) цепочка разобрана