

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Кибербезопасность информационных систем»

ФОРМАЛЬНЫЕ ЯЗЫКИ

Методические указания к проведению лабораторных работ

Ростов-на-Дону
ДГТУ
2018

УДК 004.3

Составители: Глушкова В.Н., Пиневиц Е.В.

Методические указания к проведению лабораторных работ.
– Ростов-на-Дону : Донской гос. техн. ун-т, 2018. – 20 с.

Методические указания к проведению лабораторных работ по дисциплине «Формальные языки» предназначены для студентов очной формы обучения по направлению подготовки 10.05.01 «Компьютерная безопасность» позволяют студентам оптимальным образом организовать процесс выполнения лабораторных работ и подготовки к защите лабораторных работ по контрольным вопросам в форме собеседования.

УДК 004.3

Печатается по решению редакционно-издательского совета
Донского государственного технического университета

Научный редактор: зав. кафедрой «Кибербезопасность информационных систем» канд. техн. наук, доцент А.И. Жуков

Ответственный за выпуск доцент кафедры «Кибербезопасность информационных систем» канд. техн. наук, доцент Е.В. Пиневиц

В печать _____.____.20__ г.
Формат 60x84/16. Объем _____ усл. п. л.
Тираж ____ экз. Заказ № ____.

Издательский центр ДГТУ
Адрес университета и полиграфического предприятия:
344000, г. Ростов-на-Дону, пл. Гагарина, 1

© Донской государственный
технический университет, 2018

Лабораторная работа № 1. Распознавание типов формальных языков и грамматик

Цель:

- закрепить понятия «алфавит», «цепочка», «формальная грамматика» и «формальный язык», «выводимость цепочек», «эквивалентная грамматика»;
- сформировать умения и навыки распознавания типов формальных языков и грамматик по классификации Хомского, построения эквивалентных грамматик.

Постановка задачи к лабораторной работе № 1

При выполнении лабораторной работы следует реализовать следующие действия:

- 1) составить грамматику, порождающую формальный язык, заданный в соответствии с вариантом;
- 2) определить тип формальной грамматики и языка по классификации Хомского;
- 3) разработать программное средство, распознающее тип введенной пользователем грамматики по классификации Хомского.

Варианты индивидуальных заданий представлены в таблице 1.1.

Вариант	Формальный язык
1	$L(G)=\{a^n b^m c^k \mid n, m, k > 0\}$
2	$L(G)=\{(ab)^n (cb)^m \mid n, m \geq 0\}$
3	$L(G)=\{0^n (10)^m \mid n, m \geq 0\}$
4	$L(G)=\{wcwcw \mid w \in \{a, b\}^+\}$
5	$L(G)=\{c^{2n} d^n \mid n > 0\}$
6	$L(G)=\{l+l-l \mid l \in \{a, b\}^+\}$
7	$L(G)=\{(10)^{n-1} (01)^{n+1} \mid n > 0\}$
8	$L(G)=\{(ac)^n \mid n > 0, a \in \{b, d\}, c \in \{+, -\}\}$
9	$L(G)=\{\perp (010)^n \perp \mid n > 0\}$
10	$L(G)=\{a_1 a_2 \dots a_n a_n \dots a_2 a_1 \mid a_i \in \{0, 1\}\}$
11	$L(G)=\{a_1 a_2 \dots a_n a_1 a_2 \dots a_n \mid a_i \in \{c, d\}\}$
12	$L(G)=\{ab.b \mid a_i \in \{+, -\}, b \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^+\}$

Таблица 1.1 – Варианты индивидуальных заданий к лабораторной работе № 1

Лабораторная работа № 2. Эквивалентные преобразования контекстно-свободных грамматик

Цель:

- закрепить понятия «эквивалентные грамматики», «приведенная КС-грамматика»;
- сформировать умения и навыки эквивалентных преобразований контекстно-свободных грамматик.

Постановка задачи к лабораторной работе № 2

Разработать программное средство, автоматизирующее процесс эквивалентного преобразования КС-грамматик. Программное средство должно выполнять следующие функции:

- 1) организация ввода грамматики и проверка ее на принадлежность к классу КС-грамматик;
- 2) проверка существования языка КС-грамматики;
- 3) реализация эквивалентных преобразований грамматики, направленных на удаление:
 - а) бесполезных символов;
 - б) недостижимых символов;
 - в) ϵ -правил;
 - г) цепных правил;
 - д) левой факторизации правил;
 - е) прямой левой рекурсии.

Варианты индивидуальных заданий представлены в таблице 2.1.

Вариант	Контекстно-свободная грамматика
1	$G = (\{S, A, B, D, E\}, \{a, b, c, e\}, P, S)$, где P : 1) $S \rightarrow AB \mid \varepsilon$; 2) $A \rightarrow Aa \mid S \mid a$; 3) $B \rightarrow bD \mid bS \mid b$; 4) $D \rightarrow ccD$; 5) $E \rightarrow eE \mid e$.
2	$G = (\{E, T, F, G, H\}, \{+, -, *, /, n, m, h\}, P, E)$, где P : 1) $E \rightarrow T \mid E+T \mid E-T \mid \varepsilon$; 2) $T \rightarrow F \mid F*T \mid F/T \mid \varepsilon$; 3) $F \rightarrow G \mid Fn \mid n$; 4) $G \rightarrow Gm$; 5) $H \rightarrow Hh \mid h$.
3	$G = (\{S, R, T, X, Y\}, \{a, b, p, g, y\}, P, S)$, где P : 1) $S \rightarrow R \mid T$; 2) $R \rightarrow pX \mid paR \mid paT \mid \varepsilon$; 3) $T \rightarrow Tg \mid g$; 4) $X \rightarrow aXb$; 5) $Y \rightarrow aYa \mid y$.
4	$G = (\{Q, A, B, C, D\}, \{a, b, c, d\}, P, Q)$, где P : 1) $Q \rightarrow acA \mid acB \mid \varepsilon$; 2) $B \rightarrow A \mid Cb \mid \varepsilon$; 3) $A \rightarrow Aa \mid Ab \mid a$; 4) $C \rightarrow dCc$; 5) $D \rightarrow dc$.
5	$G = (\{R, T, F, G, K\}, \{m, i, j, k, ^, \sim, \perp\}, P, R)$, где P : 1) $R \rightarrow R \sim T \perp \mid R^{\wedge} T \perp \mid \varepsilon$; 2) $T \rightarrow F \mid Fi \mid Fj \mid Gk \mid \varepsilon$; 3) $G \rightarrow GkG$; 4) $K \rightarrow K \mid Km \mid m$.
6	$G = (\{S, X, Y, Z, K\}, \{x, y, z, k, \#, \$\}, P, S)$, где P : 1) $S \rightarrow X \mid Y \mid Z$; 2) $X \rightarrow x \# X \mid x \# Y \mid \varepsilon$; 3) $Y \rightarrow Yy \$ \mid Yz \$ \mid \$ \mid \varepsilon$; 4) $Z \rightarrow Zz \$$; 5) $K \rightarrow Kk \$ \mid k \$$.
7	$G = (\{S, L, M, P, N\}, \{n, m, l, p, @, \perp\}, V, S)$, где V : 1) $S \rightarrow @nL \mid @mM \mid P$; 2) $L \rightarrow M \mid Ll \perp \mid Lm \perp \mid \varepsilon$; 3) $M \rightarrow L \mid Mm \mid mm$; 4) $N \rightarrow pN @ \mid @$; 5) $P \rightarrow nmP$.
8	$G = (\{X, Y, Z, K, L\}, \{a, b, l, =, <, >, \wedge, \vee, \neg\}, V, X)$, где V : 1) $X \rightarrow Y \mid Y=Y \mid Y < Y \mid Y > Y \mid K$; 2) $Y \rightarrow Y \wedge Z \mid Y \vee Z \mid \varepsilon$; 3) $Z \rightarrow \neg a \mid \neg b \mid \varepsilon$; 4) $K \rightarrow \neg K$; 5) $L \rightarrow l \mid a \mid b$.
9	$G = (\{Q, A, B, C, D\}, \{0, 1, -\}, P, Q)$, где P : 1) $Q \rightarrow 01A \mid 01B \mid A$; 2) $A \rightarrow 0B1 \mid B \mid 1 \mid \varepsilon$; 3) $B \rightarrow BA0 \mid B1 \mid C \mid \varepsilon$;

	4) $C \rightarrow 0C11$; 5) $D \rightarrow -D1 \mid -0 \mid -1$.
10	$G = (\{R, T, U, W, V\}, \{0, 1, +, -, *, /\}, P, R)$, где P : 1) $R \rightarrow T1T \mid T1U \mid W \mid \varepsilon$; 2) $T \rightarrow U \mid T01 \mid T10 \mid \varepsilon$; 3) $U \rightarrow +U \mid +0 \mid +1$; 4) $W \rightarrow W-W \mid W+W$; 5) $V \rightarrow *0 \mid /1$.

Таблица 2.1 – Варианты индивидуальных заданий к лабораторной работе № 2 и 5

Вариант	Контекстно-свободная грамматика
11	$G = (\{S, R, T, F, E\}, \{a, b, k, \{, [, \},], \perp\}, P, S)$, где P : 1) $S \rightarrow \{R \mid [R$; 2) $R \rightarrow Ra \mid Ra \mid a \mid T \mid F \mid \varepsilon$; 3) $F \rightarrow \{F \mid bb$; 4) $T \rightarrow [T$; 5) $E \rightarrow k \perp$.
12	$G = (\{Y, K, M, L, S\}, \{a, b, *, /, ^\wedge\}, P, Y)$, где P : 1) $Y \rightarrow KS \mid KM$; 2) $K \rightarrow K^* \mid K' \mid S$; 3) $S \rightarrow Sa / \mid Sb / \mid \varepsilon$; 4) $M \rightarrow *M^*$; 5) $L \rightarrow L^{\wedge} \mid ^{\wedge}a$.

Продолжение таблицы 2.1 – Варианты индивидуальных заданий к лабораторной работе № 2 и 5

Лабораторная работа № 3.

«Построение конечного автомата по регулярной грамматике»

Цель:

- закрепить понятия «регулярная грамматика», «недетерминированный и детерминированный конечный автомат»;
- сформировать умения и навыки построения конечного автомата по регулярной грамматике и преобразования недетерминированного конечного автомата к детерминированному конечному автомату.

Основы теории

Алгоритм 3.1. Построение КА по регулярной грамматике

Вход: регулярная грамматика $G = (V_T, V_N, P, S)$.

Выход: КА $M = (Q, T, F, H, Z)$.

Шаг 1. Пополнить грамматику правилом $A \rightarrow aN$, где $A \in V_N$, $a \in V_T$ и N - новый нетерминал, для каждого правила вида $A \rightarrow a$, если в грамматике нет соответствующего ему правила $A \rightarrow aB$, где $B \in V_N$.

Шаг 2. Начальный символ грамматики S принять за начальное состояние КА H . Из нетерминалов образовать множество состояний автомата $Q = V_N \cup \{N\}$, а из терминалов – множество символов входного алфавита $T = V_T$.

Шаг 3. Каждое правило $A \rightarrow aB$ преобразовать в функцию переходов $F(A, a) = B$, где $A, B \in V_N, a \in V_T$.

Шаг 4. Во множество заключительных состояний включить все вершины, помеченные символами $B \in V_N$ из правил вида $A \rightarrow aB$, для которых имеются соответствующие правила $A \rightarrow a$, где $A, B \in V_N, a \in V_T$.

Шаг 5. Если в грамматике имеется правило $S \rightarrow \varepsilon$, где S - начальный символ грамматики, то поместить S во множество заключительных состояний.

Шаг 6. Если получен НКА, то преобразовать его в ДКА.

Алгоритм 3.2. Преобразование НКА в ДКА

Вход: НКА $M = (Q, T, F, H, Z)$.

Выход: ДКА $M' = (Q', T, F', H, Z')$.

Шаг 1. Пометить первый столбец таблицы переходов M' ДКА начальным состоянием (множеством начальных состояний) НКА M .

Шаг 2. Заполняем очередной столбец таблицы переходов M' , помеченный символами D , для этого определяем те состояния M , которые могут быть достигнуты из каждого символа строки D при каждом входном символе x . Поместить каждое найденное множество R (в том числе \emptyset) в соответствующие позиции столбца D таблицы M' , т.е.:
 $F'(D, x) = \{s \mid s \in F(t, x) \text{ для некоторого } t \in D\}$.

Шаг 3. Для каждого нового множества R (кроме \emptyset), полученного в столбце D таблицы переходов M' , добавить новый столбец в таблицу, помеченный R .

Шаг 4. Если в таблице переходов КА M' есть столбец с незаполненными позициями, то перейти к шагу 2.

Шаг 5. Во множество Z' ДКА M' включить каждое множество, помечающее столбец таблицы переходов M' и содержащее $q \in Z$ НКА M .

Шаг 6. Составить таблицу новых обозначений множеств состояний и определить ДКА M' в этих обозначениях.

Пример 3.1. Дана регулярная грамматика $G = (\{a, b\}, \{S, A, B\}, P, S)$ с правилами P : 1) $S \rightarrow aB \mid aA$; 2) $B \rightarrow bB \mid a$; 3) $A \rightarrow aA \mid b$. Построить по регулярной грамматике КА и преобразовать полученный автомат к детерминированному виду.

Решение задачи включает следующую последовательность действий.

1 Построим по регулярной грамматике КА.

1.1 Пополним грамматику правилами $A \rightarrow bN$ и $B \rightarrow aN$, где N – новый нетерминал.

1.2 Начальное состояние конечного автомата $H=S$. Множество состояний автомата $Q = V_N = \{S, A, B, N\}$, множество символов входного алфавита $T = V_T = \{a, b\}$.

1.3 Значения сформированной функции переходов даны в таблице 2.1.

F	S	A	B	N
a	A, B	A	N	\emptyset
b	\emptyset	N	B	\emptyset

Таблица 2.1 – Функция переходов автомата М

1.4 Множество заключительных состояний $Z = \{N\}$.

1.5 Для начального символа грамматики ε -правила отсутствуют. Конечный автомат M - недетерминированный, граф НКА представлен на рисунке 2.1 слева.

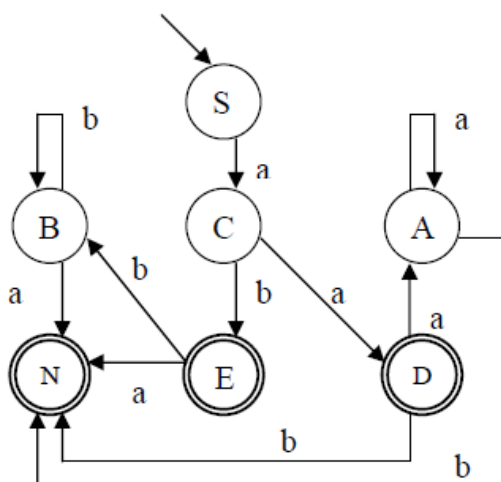
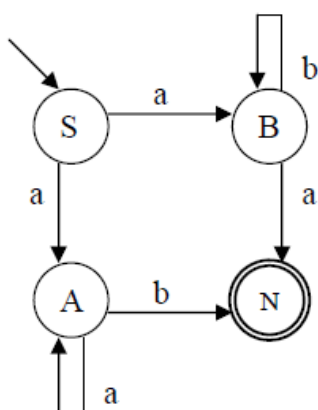


Рисунок 2.1 - Граф НКА (слева) и ДКА (справа) для Р- грамматики 2 Построим по НКА M ДКА M' .

2.1 Строим таблицу переходов для ДКА M' (таблица 2.2).

Шаг	1	2	3	4	5	6	7
F	S	A, B	A, N	B, N	A	N	B
a	A, B	A, N	A	N	A	\emptyset	N
b	\emptyset	B, N	N	B	N	\emptyset	B

Таблица 2.2 – Построение функции переходов для ДКА M'

2.2 Во множество заключительных состояний автомата M' включим элементы $Z' = \{(A, N), (B, N), N\}$.

2.3 Введем следующие новые обозначения состояний автомата M' : $(A, B) = C$, $(A, N) = D$, $(B, N) = E$.

2.4 Искомый ДКА определяется следующей пятеркой объектов:

$Q' = \{S, A, B, C, D, E, N\}$, $T = \{a, b\}$, функция переходов задана таблицей 2.3, $H = \{S\}$, $Z' = \{N, D, E\}$.

Граф полученного ДКА представлен на рисунке 2.1 справа.

F'	S	A	B	C	D	E	N
a	C	A	N	D	A	N	\emptyset
b	\emptyset	N	B	E	N	B	\emptyset

Таблица 3.3 – Функция переходов для ДКА M'

Постановка задачи к лабораторной работе № 3

Разработать программное средство, реализующее следующие функции:

- 1) ввод произвольной формальной грамматики с клавиатуры и проверка ее на принадлежность к классу регулярных грамматик;
- 2) построение по заданной регулярной грамматике конечного автомата;
- 3) преобразование недетерминированного конечного автомата к детерминированному конечному автомату;
- 4) вывод графа результирующего конечного автомата на экран.

Варианты индивидуального задания представлены в таблице 3.4.

Таблица 3.4 – Варианты индивидуального задания к лабораторной работе № 3

Вариант	Регулярная грамматика
1	$G = (\{S, C, D\}, \{0, 1\}, P, S)$, где P : 1) $S \rightarrow 1C \mid 0D$; 2) $C \rightarrow 0D \mid 0S \mid 1$; 3) $D \rightarrow 1C \mid 1S \mid 0$.
2	$G = (\{S, A, B, C\}, \{a, b, c\}, P, S)$, где P : 1) $S \rightarrow aA \mid bB \mid aC$; 2) $A \rightarrow bA \mid bB \mid c$; 3) $B \rightarrow aA \mid cC \mid b$; 4) $C \rightarrow bB \mid bC \mid a$.

Продолжение таблицы 3.4 – Варианты индивидуального задания к лабораторной работе № 3

Вариант	Регулярная грамматика
3	$G = (\{K, L, M, N\}, \{a, b, +, -, \perp\}, P, K)$, где P : 1) $K \rightarrow aL \mid bM$; 2) $L \rightarrow -N \mid -M$; 3) $M \rightarrow +N$; 4) $N \rightarrow aL \mid bM \mid \perp$.
4	$G = (\{X, Y, Z, W, V\}, \{0, 1, \sim, \#, \&\}, P, X)$, где P : 1) $X \rightarrow 0Y \mid 1Z \mid \varepsilon$; 2) $Y \rightarrow 0Z \mid \sim W \mid \#$; 3) $Z \rightarrow 1Y \mid 1W \mid 0V$; 4) $W \rightarrow 0W \mid 1W \mid \#$; 5) $V \rightarrow \&Z$.
5	$G = (\{K, L, M, N, Q, P, R, S\}, \{0, 1, *, \$, /\}, V, K)$, где V : 1) $K \rightarrow 1L \mid 0N$; 2) $L \rightarrow 0M \mid 0P \mid /Q$; 3) $N \rightarrow 1R \mid 1M \mid *S$; 4) $Q \rightarrow 1P$; 5) $P \rightarrow *L \mid \$$; 6) $M \rightarrow \$$; 7) $S \rightarrow 0R$; 8) $R \rightarrow /N \mid \$$.
6	$G = (\{E, A, B, C, D\}, \{0, 1, a, b, c\}, P, E)$, где P : 1) $E \rightarrow 0A \mid \varepsilon$; 2) $A \rightarrow aB \mid aD$; 3) $B \rightarrow bB \mid 1C \mid c$; 4) $D \rightarrow aD \mid 0C \mid c$.
7	$G = (\{X, Y, Z, V, W\}, \{0, 1, x, y, z\}, P, X)$, где P : 1) $X \rightarrow yY \mid zZ$; 2) $Y \rightarrow 1V$; 3) $Z \rightarrow 0W \mid 0Y$; 4) $V \rightarrow xZ \mid xW \mid 1$; 5) $W \rightarrow 1Y \mid 0$.
8	$G = (\{S, A, B, C, D\}, \{a, b, c, d, \perp\}, P, S)$, где P : 1) $S \rightarrow aA \mid bB$; 2) $A \rightarrow cC \mid \perp$; 3) $C \rightarrow cC \mid cA$; 4) $B \rightarrow dD \mid \perp$; 5) $D \rightarrow dD \mid dB$.
9	$G = (\{K, L, M, N, P\}, \{0, 1, \&, \%, a, b\}, C, K)$, где C : 1) $K \rightarrow 1M \mid \varepsilon$; 2) $M \rightarrow 0L \mid \&N \mid \&P$; 3) $L \rightarrow 1L \mid 0L \mid \%P$; 4) $N \rightarrow aN \mid bN \mid \%P$; 5) $P \rightarrow 1P \mid aP \mid 0$.
10	$G = (\{I, J, K, M, N\}, \{0, 1, \sim, !\}, P, I)$, где P : 1) $I \rightarrow 0J \mid 1K \mid 0M$; 2) $J \rightarrow \sim K \mid 0M$; 3) $K \rightarrow \sim M \mid 0J \mid 0N$; 4) $M \rightarrow 1K \mid !$; 5) $N \rightarrow 0I \mid 1I \mid !$.
11	$G = (\{S, A, B, C, D, E\}, \{a, b, c, d, e, \$, \perp\}, P, S)$, где P : 1) $S \rightarrow aA \mid bB \mid cC$; 2) $A \rightarrow dD$; 3) $B \rightarrow \#D \mid \$E$; 4) $D \rightarrow dD \mid dB \mid \perp$; 5) $C \rightarrow eE$; 6) $E \rightarrow eE \mid eB \mid \perp$.
12	$G = (\{X, Y, Z, V\}, \{(\,), y, z, v\}, P, X)$, где P : 1) $X \rightarrow (Y \mid \varepsilon$; 2) $Y \rightarrow yY \mid zY \mid zZ$; 3) $Z \rightarrow zZ \mid vZ \mid vV$; 4) $V \rightarrow vV \mid)$.

Лабораторная работа № 4. Минимизация конечных автоматов

Цель:

- закрепить понятия «недостижимые состояния автомата», «эквивалентные состояния автомата», «минимальный конечный автомат»;
- сформировать умения и навыки минимизации детерминированного конечного автомата.

Основы теории

Конечный автомат может содержать лишние состояния двух типов: недостижимые и эквивалентные состояния.

Определение 1. Два различных состояния q и q' в конечном автомате $M = (Q, T, F, H, Z)$ называются n -эквивалентными, $n \in N \cup \{0\}$, если, находясь в одном из этих состояний и получив на вход любую цепочку символов $\omega : \omega \in V^*, |\omega| \leq n$, автомат может перейти в одно и то же множество конечных состояний.

Определение 2. Состояние q КА называется недостижимым, если к нему нет пути из начального состояния автомата.

Определение 3. КА, не содержащий недостижимых и эквивалентных состояний, называется приведенным или минимальным КА.

Алгоритм 1. Устранение недостижимых состояний КА

Вход: КА $M = (Q, T, F, H, Z)$.

Выход: КА $M' = (Q', T, F', H, Z')$.

Шаг 1. Поместить начальное состояние КА в список достижимых состояний Q_0 , т.е. $Q_0^0 = H$.

Шаг 2. Для новых элементов списка достижимых состояний пополнить список группой их состояний-приемников, отсутствующих в нем, т.е.

$$Q_0^i = Q_0^{i-1} \cup \{p \mid \forall q \in Q_0^{i-1} \exists F(q, t) = p\}$$

Шаг 3. Повторить шаг 2, пока список достижимых состояний не перестанет меняться. То есть, если $Q_0^i \neq Q_0^{i-1}$ то $i := i+1$, иначе $Q_0 = Q_0^i$.

Шаг 4. Исключить из множества Q состояний КА все состояния, отсутствующие в списке Q_0 достижимых состояний, т.е. $Q' = Q \cap Q_0$.

Шаг 5. Исключить недостижимые заключительные состояния и пары функции переходов, содержащие недостижимые состояния, т.е.

$$Z' = Z \cap Q_0,$$

$$F' = F - \{F(q, t) = p \mid q \in (Q - Q_0)\}.$$

Пример 1. Устранить недостижимые состояния КА $M = (Q, T, F, H, Z)$, где $Q = \{A, B, C, D, E, F, G\}$, $T = \{a, b\}$, $H = \{A\}$, $Z = \{D, E\}$ и функция переходов задана таблицей 3.1. Граф исходного КА M представлен на рисунке 4.1.

F	A	B	C	D	E	F	G
a	B			C	B	D	F
b	C	D	E	E	D	G	E

Таблица 4.1 – Функция переходов конечного автомата M

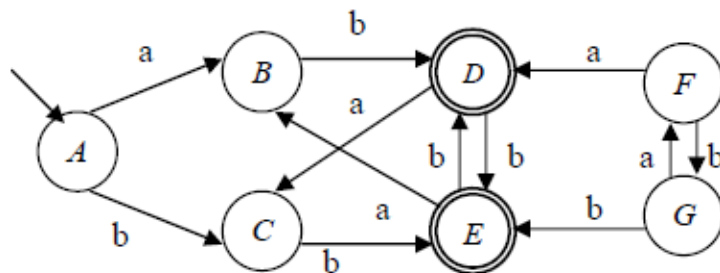


Рисунок 4.1 – Граф исходного конечного автомата M

Последовательность устранения недостижимых состояний КА имеет вид:

$$Q_0 = \{A\};$$

$$Q_1 = \{A, B, C\};$$

$$Q_2 = \{A, B, C, D, E\};$$

$$Q_3 = \{A, B, C, D, E\}; \text{ т.к. } Q_2 = Q_3, \text{ то } Q_d = \{A, B, C, D, E\}.$$

$$Q_n = \{F, G\}; Q' = \{A, B, C, D, E\}; Z' = \{D, E\}.$$

Функция переходов автомата M' представлена в таблице 4.2.

F	A	B	C	D	E
a	B			C	B
b	C	D	E	E	D

Таблица 4.2 - Функция переходов автомата M'

Граф КА M' после устранения недостижимых состояний представлен на рисунке 4.2.

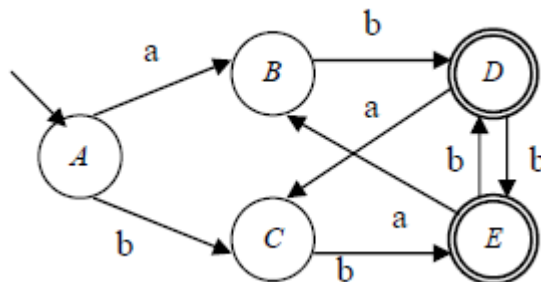


Рисунок 4.2 - Граф КА M' после устранения недостижимых состояний

Алгоритм 2. Объединение эквивалентных состояний КА

Вход: КА $M' = (Q', T, F', H, Z')$ без недостижимых состояний.

Выход: минимальный КА $M'' = (Q'', T, F'', H, Z'')$.

Шаг 1. На первом шаге строим нулевое разбиение $R(0)$, состоящее из двух классов эквивалентности: заключительные состояния КА - Z и не заключительные - $Q-Z$.

Шаг 2. На очередном шаге построения разбиения $R(n)$ в классы эквивалентности включить те состояния, которые по одинаковым входным символам переходят в $n-1$ эквивалентные состояния, т.е.

$$R(n) = \{r_i(n) : \{q_{ij} \in Q : \forall t \in T F(q_{ij}, t) \subseteq r_j(n-1)\} \forall i, j \in N\}.$$

Шаг 3. До тех пор, пока $R(n) \neq R(n-1)$ полагаем $n := n+1$ и идем к шагу 2.

Шаг 4. Переобозначить оставшиеся неразбитые группы состояний и включить их в таблицу новых обозначений состояний автомата.

Шаг 5. Определить эквивалентный КА M'' в новых обозначениях.

Пример 2. Минимизировать конечный автомат из примера 4.1.

Последовательность построения разбиений будет иметь вид:

$$R(0) = \{\{A, B, C\}, \{D, E\}\}, n = 0;$$

$$R(1) = \{\{A\}, \{B, C\}, \{D, E\}\}, n = 1;$$

$$R(2) = \{\{A\}, \{B, C\}, \{D, E\}\}, n=2.$$

Т.к. $R(1) = R(2)$, то искомое разбиение построено.

Переобозначим оставшиеся неразбитые группы состояний:

$$X = \{B, C\}, Y = \{D, E\}.$$

Получим минимальный автомат M'' , где $Q'' = \{A, X, Y\}$, $Z'' = \{Y\}$.

Функция переходов автомата M'' представлена в таблице 4.3.

F''	A	X	Y
a	X		X
b	X	Y	Y

Таблица 3.3 - Функция переходов автомата M''

Граф переходов конечного автомата после его минимизации показан на рисунке 4.3.

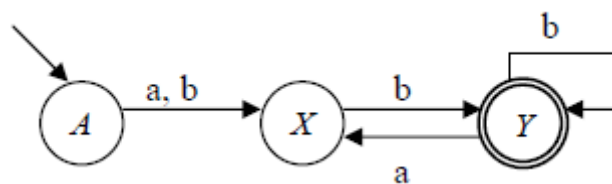


Рисунок 4.3 – Граф минимального КА M''

Постановка задачи к лабораторной работе № 4

Разработать программное средство, реализующее следующие функции:

- 1) ввод исходного конечного автомата и вывод на экран его графа;
- 2) устранение недостижимых состояний конечного автомата;
- 3) исключение эквивалентных состояний конечного автомата;
- 4) вывод на экран графа минимального конечного автомата.

Разработать серию контрольных примеров для тестирования реализованных алгоритмов.

Варианты индивидуальных заданий к лабораторной работе № 4 представлены на рисунке 4.4.

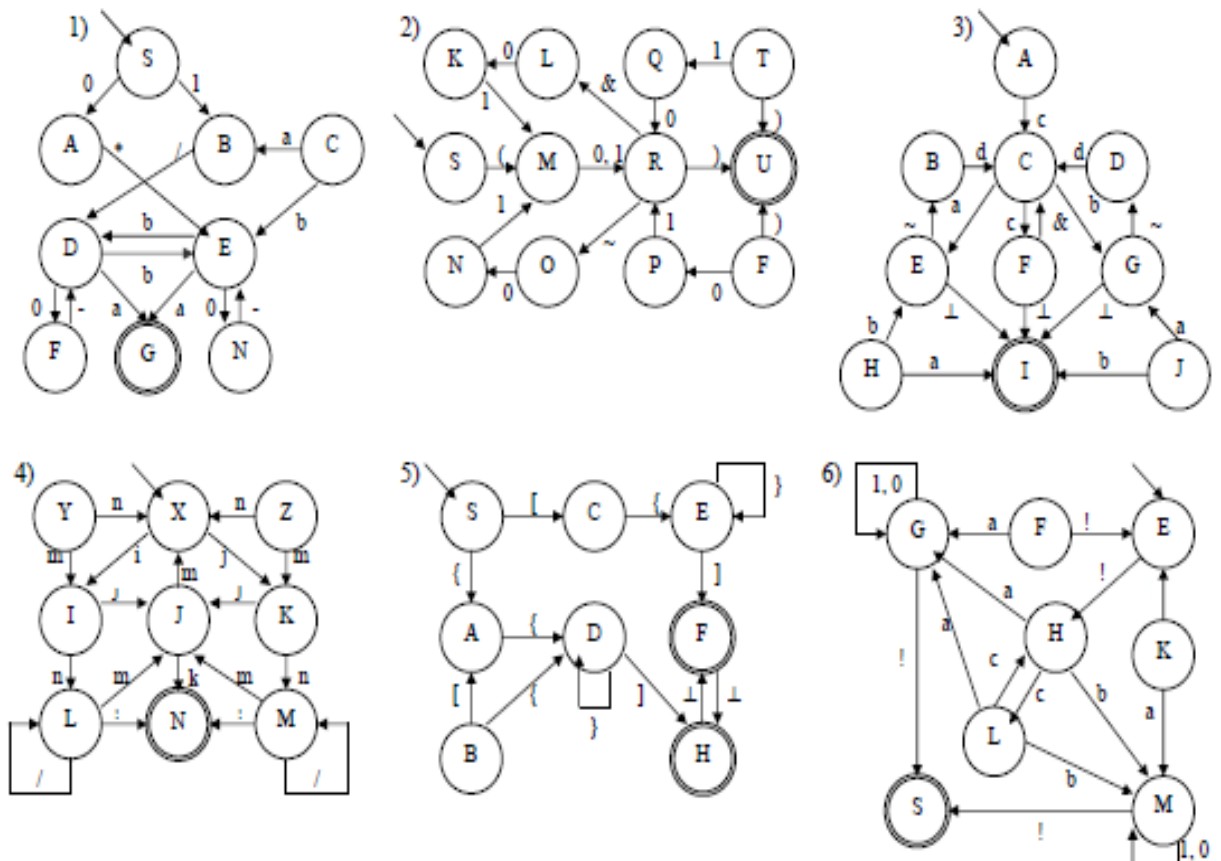


Рисунок - 4.4 Варианты индивидуальных заданий к лабораторной работе № 4

Лабораторная работа № 5. Построение автомата с магазинной памятью по контекстно-свободной грамматике

Цель: - закрепить понятия «автомат с магазинной памятью (МП-автомат)», «расширенный МП-автомат», «конфигурация МП-автомата»; «строка и язык, допускаемые МП-автоматом»;

- сформировать умения и навыки построения МП-автомата и расширенного МП-автомата по КС-грамматике, разбора входной строки с помощью МП-автомата.

Основы теории

КС-языки можно распознавать с помощью автомата с магазинной памятью (МП-автомата).

Определение 5.1. МП-автомат можно представить в виде семерки:

$$M = (Q, T, N, F, q_0, N_0, Z), \quad (5.1)$$

где Q – конечное множество состояний автомата;

T – конечный входной алфавит;

N – конечный магазинный алфавит;

F – магазинная функция, отображающая множество $(Q \times (T \cup \{ \varepsilon \}) \times N)$ во множество всех подмножеств множества $Q \times N^*$, т.е.

$$F: (Q \times (T \cup \{ \varepsilon \}) \times N) \rightarrow P(Q \times N^*);$$

q_0 – начальное состояние автомата, $q_0 \in Q$;

N_0 – начальный символ магазина, $N_0 \in T$;

Z – множество заключительных состояний автомата, $Z \subseteq Q$.

Определение 5.2. Конфигурацией МП-автомата называется тройка вида:

$$(q, \omega, \alpha) \in (Q \times T^* \times N^*), \quad (5.2)$$

где q – текущее состояние автомата, $q \in Q$;

ω – часть входной строки, первый символ которой находится под входной головкой, $\omega \in T^*$;

α – содержимое магазина, $\alpha \in N^*$.

Общая схема МП-автомата представлена на рисунке 5.1.

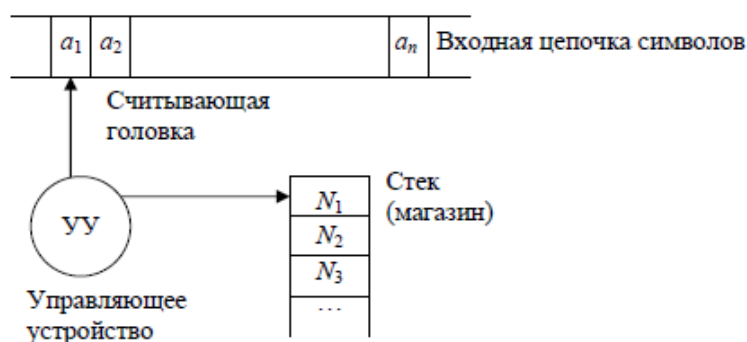


Рисунок 5.1 – Схема МП-автомата

Алгоритм 5.1. Функционирование МП-автомата

Начальной конфигурацией МП-автомата является конфигурация (q_0, ω, N_0) . Шаг работы МП-автомата будем представлять в виде отношения непосредственного следования конфигураций (обозначается « $\mid=\rangle$ ») и отношения достижимости конфигураций (обозначается « $\mid=^*$ »). Если одним из значений магазинной функции $F(q \in Q, t \in (T \cup \{\varepsilon\}), S \in N)$ является $(q' \in Q, \gamma \in N^*)$, то записывается $(q, t\omega, S\alpha) \mid= (q', \omega, \gamma\alpha)$. При этом возможны следующие варианты.

1) Случай $t \in T$. Автомат находится в текущем состоянии q , читает входной символ t , имеет в вершине стека символ S . Он переходит в очередное состояние q' , сдвигает входную головку на ячейку вправо и заменяет верхний символ S строкой γ магазинных символов. Вариант $\gamma = \varepsilon$ означает, что S удаляется из стека.

2) Случай $t = \varepsilon$. Отличается от первого случая тем, что входной символ t просто не принимается во внимание, и входная головка не сдвигается. Такой шаг работы МП-автомата называется ε -шагом, который может выполняться даже после завершения чтения входной строки.

Заключительной конфигурацией МП-автомата является конфигурация (q, ε, α) , где $q \in Z$.

Определение 5.3. МП-автомат допускает входную строку ω , если существует путь по конфигурациям $(q_0, \omega, N_0) \mid=^* (q, \varepsilon, \alpha)$ для некоторых $q \in Z$ и $\alpha \in N^*$.

Определение 5.4. Язык L , распознаваемый (принимаемый) МП-автоматом M определяется как множество вида:

$$L(M) = \{ \omega \mid \omega \in T^* \text{ и } (q_0, \omega, N_0) \mid=^* (q, \varepsilon, \alpha) \text{ для некоторых } q \in Z \text{ и } \alpha \in N^* \}.$$

Определение 5.5. МП-автомат с магазинной функцией

$F: (Q \times (T \cup \{ \varepsilon \}) \times N^*) \rightarrow P(Q \times N^*)$ называется расширенным МП-автоматом, т.е. автоматом, который может заменять цепочку символов конечной длины в верхушке стека на другую цепочку символов конечной длины.

Существуют КС-языки, МП-автоматы и расширенные МП-автоматы, определяющие один и тот же язык.

Алгоритм 5.2. Построение МП-автомата по КС-грамматике

Построим МП-автомат, выполняющий левосторонний разбор. Данный автомат обладает только одним состоянием и принимает входную строку опустошением магазина. Стек используется для размещения текущей сентенции, первоначально это начальный символ грамматики. Очередная сентенция получается заменой верхнего нетерминала стека.

Вход: КС-грамматика $G = (V_T, V_N, P, S)$.

Выход: МП-автомат $M = (Q, T, N, F, q_0, n_0, Z)$ такой, что $L(M) = L(G)$.

Шаг 1. Положить $Q = \{q\}$, $q_0 = q$, $Z = \varnothing$, $N = V_T \cup V_N$, $T = V_T$, $N_0 = S$.

Шаг 2. Для каждого правила вида $(A \rightarrow \beta) \in P$, где $\beta \in V^*$, сформировать магазинную функцию вида $F(q, \varepsilon, A) = (q, \beta)$. Эти функции предписывают замещать нетерминал в вершине стека по правилу грамматики.

Шаг 3. Для каждого $t \in V_T$ сформировать магазинную функцию вида $F(q, t, t) = (q, \varepsilon)$, которая выталкивает из стека символ, совпадающий с входным, и перемещает читающую головку. Эти функции обеспечивают опустошение стека.

Пример 5.1. Дана КС-грамматика:

$G(\{+, (,), a\}, \{S, A\}, \{S \rightarrow S+A / A, A \rightarrow (S) / a\}, \{S\})$. Последовательность построения МП-автомата будет иметь вид.

1) $Q = \{q\}$, $q_0 = q$, $T = \{+, (,), a\}$, $N = \{+, (,), a, S, A\}$, $N_0 = S$, $Z = \varnothing$.

2) $F(q, \varepsilon, S) = (q, S+A)$, $F(q, \varepsilon, S) = (q, A)$, $F(q, \varepsilon, A) = (q, (S))$;
 $F(q, \varepsilon, A) = (q, a)$.

3) $F(q, t, t) = (q, \varepsilon)$ для каждого $t \in \{+, (,), a\}$.

Распознавание строки (a) построенным МП-автоматом представлено в таблице 5.1. Полученный МП-автомат является недетерминированным.

Номер конфигурации	Текущее состояние	Входная строка	Содержимое магазина
1	q	(a)	S
2	q	(a)	A
3	q	(a)	(S)
4	q	$a)$	$S)$
5	q	$a)$	$A)$
6	q	$a)$	$a)$
7	q	$)$	$)$
8	q	ε	ε

Таблица 5.1 – Распознавание МП-автоматом строки (a)

Алгоритм 5.3. Построение расширенного МП-автомата по КС-грамматике

Построим МП-автомат, выполняющий правосторонний разбор. Данный автомат имеет единственное текущее состояние и одно заключительное состояние, в котором стек пуст. Стек содержит левую часть текущей сентенции. Первоначально в стек помещается специальный магазинный символ, маркер пустого стека $\#$. На каждом шаге автомат по правилу грамматики замещает нетерминалом строку верхних символов стека или дописывает в вершину входной символ.

Вход: КС-грамматика $G = (V_T, V_N, P, S)$.

Выход: расширенный МП-автомат $M = (Q, T, N, F, q_0, N_0, Z)$ такой, что $L(M) = L(G)$.

Шаг 1. Положить $Q = \{q, r\}$, $q_0 = q$, $Z = \{r\}$, $N = V_T \cup V_N \cup \{\#\}$, $T = V_T$, $N_0 = \#$.

Шаг 2. Для каждого правила вида $(A \rightarrow \beta) \in P$, где $\beta \in V^*$, сформировать магазинную функцию вида $F(q, \varepsilon, \beta) = (q, A)$, предписывающую заменять правую часть правила в вершине стека нетерминалом из левой части, независимо от текущего символа входной строки.

Шаг 3. Для каждого терминала $t \in T$ сформировать магазинную функцию вида $F(q, t, \varepsilon) = (q, t)$, которая помещает символ входной строки в вершину стека, если там нет правой части правила, и перемещает читающую головку.

Шаг 4. Предусмотреть магазинную функцию для перевода автомата в заключительное состояние $F(q, \varepsilon, \#S) = (r, \varepsilon)$.

Пример 5.2. Для грамматики из примера 5.1 построить расширенный МП-автомат. Последовательность построения МП-автомата будет иметь вид.

- 1) $Q = \{q, r\}, q_0 = q, T = \{+, (,), a\}, N = \{+, (,), a, S, A\}, N_0 = \#, Z = r.$
- 2) $F(q, \varepsilon, S+A) = (q, S), F(q, \varepsilon, A) = (q, S), F(q, \varepsilon, (S)) = (q, A),$
 $F(q, \varepsilon, a) = (q, A).$
- 3) $F(q, t, \varepsilon) = (q, t)$ для каждого $t \in \{+, (,), a\}.$
- 4) $F(q, \varepsilon, \#S) = (r, \varepsilon).$

Распознавание строки (a) расширенным МП-автоматом представлено в таблице 5.2. Полученный МП-автомат является детерминированным.

Номер конфигурации	Текущее состояние	Входная строка	Содержимое магазина
1	q	(a)	$\#$
2	q	$a)$	$\#($
3	q	$)$	$\#(a$
4	q	$)$	$\#(A$
5	q	$)$	$\#(S$
6	q	ε	$\#(S)$
7	q	ε	$\#A$
8	q	ε	$\#S$
9	r	ε	ε

Таблица 5.2 – Распознавание расширенным МП-автоматом строки (a)

Постановка задачи к лабораторной работе № 5

Разработать программное средство, реализующее следующие функции:

- а) ввод произвольной формальной грамматики и проверка ее на принадлежность к классу КС-грамматик;
- б) построение МП-автомата по КС-грамматике;
- в) построение расширенного МП-автомата по КС-грамматике.

Продемонстрировать разбор некоторой входной строки с помощью построенных автоматов для случая:

- а) входная строка принадлежит языку исходной КС-грамматики и допускается МП-автоматом;
- б) входная строка не принадлежит языку исходной КС-грамматики и не принимается МП-автоматом.

Индивидуальные варианты заданий представлены в таблице 2.1.

Содержание

1. Лабораторная работа № 1. Распознавание типов формальных языков и грамматик	3
2. Лабораторная работа № 2. Эквивалентные преобразования контекстно-свободных грамматик	
3. Лабораторная работа № 3. Построение конечного автомата по регулярной грамматике.....	6
4. Лабораторная работа № 4. Минимизация конечных автоматов.....	11
5. Лабораторная работа № 5. Построение автомата с магазинной памятью по контекстно-свободной грамматике	15