

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Белгородский государственный технологический университет
им. В. Г. Шухова

Кафедра программного обеспечения вычислительной техники
и автоматизированных систем

Утверждено
научно-методическим советом
университета

Теория автоматов и формальных языков

*Методические указания к выполнению лабораторных работ
для студентов, обучающихся по направлению бакалавриата
09.03.04 «Программная инженерия»*

Белгород
2017

УДК 519.685.3(075)
ББК 32.973.26-018я7
Т33

Составитель доц. *Ю.Д. Рязанов*

Теория автоматов и формальных языков: методические указания к выполнению лабораторных работ для студентов, обучающихся по направлению бакалавриата 09.03.04 «Программная инженерия» / сост. Ю.Д. Рязанов. — Белгород: Изд-во БГТУ, 2017. — 24 с.

В методических указаниях представлены задания к лабораторным работам, охватывающим весь курс дисциплины «Теория автоматов и формальных языков».

Издание предназначено для студентов, обучающихся по направлению бакалавриата 09.03.04 «Программная инженерия»

Методические указания публикуются в авторской редакции.

УДК 519.685.3(075)
ББК 32.973.26-018я7

© Белгородский государственный
технологический университет
(БГТУ) им. В.Г.Шухова, 2017

Оглавление

Лабораторная работа № 1	
Формальные грамматики. Выводы	4
Лабораторная работа № 2	
Преобразования КС-грамматик.....	11
Лабораторная работа № 3	
Регулярные языки и конечные распознаватели	13
Лабораторная работа № 4	
Нисходящая обработка контекстно-свободных языков.....	18
Лабораторная работа № 5	
Восходящая обработка контекстно-свободных языков методом «перенос-опознание»	20
Лабораторная работа № 6	
Восходящая обработка контекстно-свободных языков методом «перенос-свертка»	22

Лабораторная работа № 1

Формальные грамматики. Выводы

Цель работы: изучить основные понятия теории формальных языков и грамматик.

З а д а н и е

1. Написать программу, выполняющую левый вывод в заданной КС-грамматике.

Исходные данные: КС-грамматика.

Выполнение: на каждом шаге вывода отображается промежуточная цепочка (на первом шаге она представляет собой начальный нетерминал, на последнем — терминальную цепочку) и правила, которые можно применить на данном шаге. Пользователь выбирает одно из предложенных правил и процесс повторяется.

Результат: терминальная цепочка, последовательность номеров правил, участвовавших в её выводе, дерево вывода в линейной скобочной форме (ЛСФ ДВ).

Пример выполнения программы

КС-грамматика:

1. $S \rightarrow aAbS$
2. $S \rightarrow b$
3. $A \rightarrow SAc$
4. $A \rightarrow \varepsilon$

Левый вывод:

Шаг 1.

Промежуточная цепочка: S

Можно применить:

1. $S \rightarrow aAbS$
2. $S \rightarrow b$

Применяем правило 1

Шаг 2.

Промежуточная цепочка: $aAbS$

Можно применить:

3. $A \rightarrow SAc$
4. $A \rightarrow \varepsilon$

Применяем правило 3

Шаг 3.

Промежуточная цепочка: $aSAcbS$

Можно применить:

1. $S \rightarrow aAbS$
2. $S \rightarrow b$

Применяем правило 2

Шаг 4.

Промежуточная цепочка: $abAcbS$

Можно применить:

3. $A \rightarrow SAc$
4. $A \rightarrow \epsilon$

Применяем правило 4

Шаг 5.

Промежуточная цепочка: $abcbS$

Можно применить:

1. $S \rightarrow aAbS$
2. $S \rightarrow b$

Применяем правило 2

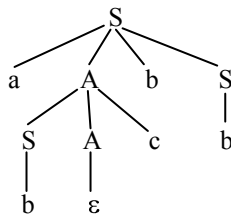
Шаг 6.

Терминальная цепочка: $авсвв$

Последовательность правил: 1 3 2 4 2

ЛСФ ДВ: $S(aA(S(b)A())c)bS(b)$

Примечание. Дерево вывода имеет вид:



2. Выполнить левый (правый) вывод терминальной цепочки в заданной грамматике (см. варианты заданий п.1), построить дерево вывода. Определить, существует ли неэквивалентный вывод полученной цепочки и, если существует, представить его деревом вывода.

3. Написать программу, определяющую, можно ли применить заданную последовательность правил при левом выводе цепочки в заданной КС-грамматике.

Исходные данные:

- КС-грамматика;
- последовательность номеров правил.

Результат: “да” или “нет”.

Пример выполнения программы

КС-грамматика: 1. $S \rightarrow aAbS$
 2. $S \rightarrow b$
 3. $A \rightarrow SAc$
 4. $A \rightarrow \varepsilon$

Последовательность правил: 1 3 2 1 4

Результат: “нет”.

4. Для каждой последовательности правил (см. варианты заданий п.2) определить, можно ли её применить при левом (правом) выводе терминальной цепочки в заданной КС-грамматике, и, если можно, построить дерево вывода.

5. Написать программу, определяющую, можно ли применить заданную последовательность правил при выводе цепочки в заданной КС-грамматике.

Исходные данные:

- КС-грамматика;
- последовательность номеров правил.

Результат: “да” или “нет”.

Пример выполнения программы

КС-грамматика: 1. $S \rightarrow aAbS$
 2. $S \rightarrow b$
 3. $A \rightarrow SAc$
 4. $A \rightarrow \varepsilon$

Последовательность правил: 1 3 2 1 4

Результат: “да”.

6. Для каждой последовательности правил (см. варианты заданий п.2) определить, можно ли её применить при выводе терминальной цепочки в заданной КС-грамматике, и, если можно, построить дерево вывода и записать эквивалентные левый и правый вывод.

В а р и а н т ы з а д а н и й

Вариант 1

1. КС-грамматика

- | | |
|-------------------------|--------------------------------|
| 1. $S \rightarrow aSSa$ | 5. $A \rightarrow Ba$ |
| 2. $S \rightarrow bSb$ | 6. $A \rightarrow \varepsilon$ |
| 3. $S \rightarrow cAb$ | 7. $B \rightarrow bB$ |
| 4. $A \rightarrow Aa$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 1, 3, 4, 6, 3, 5, 7, 8, 6, 2, 3, 6
2. 1, 1, 2, 3, 3, 3, 4, 5, 6, 6, 7, 8, 6
3. 1, 3, 3, 5, 7, 4, 6, 8, 6, 2, 3, 6, 6
4. 1, 2, 3, 6, 1, 3, 5, 7, 8, 6, 3, 4, 6

Вариант 2

1. КС-грамматика

- | | |
|-------------------------|--------------------------------|
| 1. $S \rightarrow Aba$ | 5. $A \rightarrow BaB$ |
| 2. $S \rightarrow bS$ | 6. $A \rightarrow \varepsilon$ |
| 3. $S \rightarrow cAbA$ | 7. $B \rightarrow b$ |
| 4. $A \rightarrow AaS$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 4, 7, 5, 3, 7, 7, 6, 6, 4, 6, 2, 3, 6, 6
2. 1, 7, 4, 3, 6, 6, 5, 7, 4, 2, 3, 6, 6, 6, 7
3. 1, 4, 5, 7, 7, 4, 6, 2, 1, 6, 7, 3, 6, 6, 7
4. 1, 4, 3, 5, 7, 6, 6, 7, 7, 4, 2, 6, 6, 7, 6

Вариант 3

1. КС-грамматика

- | | |
|------------------------|------------------------|
| 1. $S \rightarrow Ssa$ | 5. $A \rightarrow BaB$ |
| 2. $S \rightarrow b$ | 6. $A \rightarrow S$ |
| 3. $S \rightarrow Ab$ | 7. $B \rightarrow b$ |
| 4. $A \rightarrow AaA$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 2, 3, 4, 5, 6, 7, 8, 6, 2, 2
2. 1, 2, 3, 4, 5, 7, 8, 6, 2, 6, 2
3. 1, 3, 4, 6, 2, 5, 8, 6, 2, 7, 2
4. 1, 2, 3, 4, 6, 6, 2, 2, 7, 8, 6

Вариант 4

1. КС-грамматика

- | | |
|--------------------------|--------------------------------|
| 1. $S \rightarrow AaBaA$ | 5. $A \rightarrow a$ |
| 2. $S \rightarrow bSb$ | 6. $A \rightarrow \varepsilon$ |
| 3. $A \rightarrow B$ | 7. $B \rightarrow bS$ |
| 4. $A \rightarrow Sa$ | 8. $B \rightarrow A$ |

2. Последовательности правил вывода

1. 2, 1, 6, 8, 3, 8, 6, 4, 1, 5, 8, 6, 5
2. 2, 1, 8, 3, 8, 6, 4, 1, 8, 6, 5, 5, 6
3. 2, 1, 4, 1, 5, 8, 6, 5, 8, 3, 8, 6, 6
4. 2, 1, 8, 6, 4, 3, 8, 6, 8, 6, 3, 4, 6

Вариант 5

1. КС-грамматика

- | | |
|------------------------|--------------------------------|
| 1. $S \rightarrow SS$ | 5. $A \rightarrow Ba$ |
| 2. $S \rightarrow A$ | 6. $A \rightarrow \varepsilon$ |
| 3. $A \rightarrow AbA$ | 7. $B \rightarrow bB$ |
| 4. $A \rightarrow B$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 1, 2, 6, 2, 3, 6, 6, 2, 4, 8, 6
2. 1, 2, 4, 8, 6, 1, 2, 6, 2, 3, 6, 6
3. 1, 1, 2, 3, 6, 6, 2, 6, 2, 4, 8, 6
4. 1, 2, 1, 4, 2, 2, 8, 6, 3, 6, 6, 6

Вариант 6

1. КС-грамматика

- | | |
|------------------------|-------------------------|
| 1. $S \rightarrow SaS$ | 5. $A \rightarrow b$ |
| 2. $S \rightarrow A$ | 6. $B \rightarrow bB$ |
| 3. $A \rightarrow aA$ | 7. $B \rightarrow bBaB$ |
| 4. $A \rightarrow AB$ | 8. $B \rightarrow a$ |

2. Последовательности правил вывода

1. 1, 2, 4, 5, 8, 2, 4, 4, 5, 8, 6, 8
2. 1, 2, 4, 4, 5, 8, 6, 8, 2, 4, 5, 8
3. 1, 2, 4, 6, 8, 4, 8, 5, 2, 4, 8, 5
4. 1, 2, 2, 4, 4, 4, 6, 5, 8, 5, 8, 8

Вариант 7

1. КС-грамматика

- | | |
|-------------------------|-----------------------|
| 1. $S \rightarrow aSbS$ | 5. $A \rightarrow Ba$ |
| 2. $S \rightarrow aS$ | 6. $A \rightarrow b$ |
| 3. $S \rightarrow A$ | 7. $B \rightarrow b$ |
| 4. $A \rightarrow Sa$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 2, 3, 5, 8, 4, 3, 6, 2, 3, 5, 8, 6
2. 2, 1, 3, 5, 8, 4, 3, 6, 2, 3, 5, 8, 6
3. 1, 2, 2, 3, 3, 5, 5, 8, 8, 4, 6, 3, 6
4. 2, 1, 2, 3, 5, 8, 6, 3, 5, 8, 4, 3, 6

Вариант 8

1. КС-грамматика

- | | |
|-------------------------|-----------------------|
| 1. $S \rightarrow aSbA$ | 5. $A \rightarrow aB$ |
| 2. $S \rightarrow aB$ | 6. $A \rightarrow S$ |
| 3. $S \rightarrow A$ | 7. $B \rightarrow b$ |
| 4. $A \rightarrow aAbS$ | 8. $B \rightarrow aA$ |

2. Последовательности правил вывода

1. 1, 3, 4, 5, 7, 2, 7, 5, 7
2. 1, 1, 2, 7, 5, 7, 3, 2, 7
3. 1, 5, 7, 3, 4, 2, 7, 5, 7
4. 1, 3, 1, 5, 2, 2, 7, 7, 7

Вариант 9

1. КС-грамматика

- | | |
|-------------------------|-----------------------|
| 1. $S \rightarrow SbSa$ | 5. $A \rightarrow aB$ |
| 2. $S \rightarrow Sa$ | 6. $A \rightarrow b$ |
| 3. $S \rightarrow A$ | 7. $B \rightarrow b$ |
| 4. $A \rightarrow aS$ | 8. $B \rightarrow Aa$ |

2. Последовательности правил вывода

1. 1, 2, 3, 5, 8, 6, 2, 3, 5, 8, 4, 3, 6
2. 2, 1, 3, 5, 8, 4, 3, 6, 2, 3, 5, 8, 6
3. 1, 2, 2, 3, 3, 5, 5, 8, 8, 4, 6, 3, 6
4. 2, 1, 2, 3, 5, 8, 6, 3, 5, 8, 4, 3, 6

Вариант 10

1. КС-грамматика

- | | |
|------------------------|-------------------------|
| 1. $S \rightarrow SaS$ | 5. $A \rightarrow b$ |
| 2. $S \rightarrow A$ | 6. $B \rightarrow Bb$ |
| 3. $A \rightarrow Aa$ | 7. $B \rightarrow BaBb$ |
| 4. $A \rightarrow BA$ | 8. $B \rightarrow a$ |

2. Последовательности правил вывода

- 1, 2, 4, 5, 8, 2, 4, 4, 5, 8, 6, 8
- 1, 2, 4, 4, 5, 8, 6, 8, 2, 4, 5, 8
- 1, 2, 4, 6, 8, 4, 8, 5, 2, 4, 8, 5
- 1, 2, 2, 4, 4, 4, 6, 5, 8, 5, 8, 8

Контрольные вопросы

1. Что такое формальный язык?
 2. Назовите способы задания формальных языков.
 3. Что такое формальная грамматика?
 4. Какие классы грамматик выделяют по классификации Хомского?
 5. Приведите примеры грамматик различных классов.
 6. Какие классы языков выделяют по классификации Хомского?
 7. Приведите примеры языков различных классов.
 8. Что такое терминальная цепочка?
 9. В чём заключается вывод терминальной цепочки в грамматике.
 10. Что такое промежуточная цепочка вывода?
 11. Приведите примеры различных способов представления вывода.
- Что такое дерево вывода?
12. Какие выводы называются эквивалентными?
 13. Чем различаются неэквивалентные выводы?
 14. Приведите примеры эквивалентных выводов?
 15. Приведите примеры неэквивалентных выводов?
 16. Чем различаются левые и правые выводы?
 17. Определите класс КС-грамматик, в которых левый и правый выводы одной и той же цепочки совпадают.
 18. Как определить, существуют ли два неэквивалентных вывода заданной выводимой цепочки?
 19. Какие грамматики называются неоднозначными?
 20. Приведите примеры неоднозначных грамматик.

Лабораторная работа № 2

Преобразования КС-грамматик

Цель работы: изучить основные эквивалентные преобразования КС-грамматик и научиться применять их для получения КС-грамматик, обладающих заданными свойствами.

З а д а н и е

1. Преобразовать исходную грамматику G (см. варианты заданий) в грамматику G_1 без лишних символов.
2. Преобразовать грамматику G_1 в грамматику G_2 без ε -правил.
3. Преобразовать грамматику G_1 в грамматику G_3 без цепных правил.
4. Преобразовать грамматику G_1 в грамматику G_4 без левой рекурсии.
5. Преобразовать грамматику G_1 в грамматику G_5 без несаморекурсивных нетерминалов.
6. Получить грамматику G_6 , эквивалентную грамматике G_1 , в которой правая часть каждого правила состоит либо из одного терминала, либо двух нетерминалов.
7. Получить грамматику G_7 , эквивалентную грамматике G_1 , в которой правая часть каждого правила начинается терминалом.
8. Получить грамматику G_8 , эквивалентную грамматике G_1 , в которой правая часть каждого не ε -правила начинается терминалом и любые два правила с одинаковой левой частью различаются первым символом в правой части.
9. Получить грамматику G_9 , эквивалентную грамматике G_1 , в которой правая часть каждого правила не содержит двух стоящих рядом нетерминала.
10. Получить грамматику G_{10} , эквивалентную грамматике G_1 , в которой любой символ занимает либо только крайнюю правую позицию в правых частях правил, либо находится левее самого правого символа в правых частях правил.

В а р и а н т ы з а д а н и й

Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5
1. $A \rightarrow aBBb$	1. $T \rightarrow aRMb$	1. $E \rightarrow EabE$	1. $S \rightarrow EabS$	1. $B \rightarrow EabB$
2. $A \rightarrow B$	2. $T \rightarrow R$	2. $E \rightarrow BaD$	2. $S \rightarrow SaD$	2. $E \rightarrow BaDb$
3. $B \rightarrow Bab$	3. $T \rightarrow \varepsilon$	3. $D \rightarrow \varepsilon$	3. $S \rightarrow \varepsilon$	3. $E \rightarrow B$
4. $B \rightarrow AaCD$	4. $M \rightarrow Q$	4. $D \rightarrow aEAb$	4. $D \rightarrow EbAb$	4. $A \rightarrow aEAb$
5. $B \rightarrow \varepsilon$	5. $M \rightarrow Sbb$	5. $D \rightarrow bD$	5. $D \rightarrow S$	5. $A \rightarrow b$
6. $C \rightarrow aBaE$	6. $S \rightarrow aRQb$	6. $D \rightarrow C$	6. $D \rightarrow SB$	6. $A \rightarrow E$
7. $C \rightarrow bb$	7. $S \rightarrow RQ$	7. $C \rightarrow BCa$	7. $E \rightarrow Ba$	7. $D \rightarrow BDa$
8. $C \rightarrow \varepsilon$	8. $R \rightarrow Rab$	8. $C \rightarrow CEb$	8. $E \rightarrow Eab$	8. $D \rightarrow DEb$
9. $D \rightarrow Eda$	9. $R \rightarrow SaTQ$	9. $A \rightarrow abC$	9. $A \rightarrow ab$	9. $A \rightarrow abD$
10. $D \rightarrow DEb$	10. $R \rightarrow \varepsilon$	10. $A \rightarrow Abb$	10. $A \rightarrow Bab$	10. $B \rightarrow DaB$
11. $E \rightarrow abD$	11. $Q \rightarrow SQa$	11. $B \rightarrow aECb$	11. $B \rightarrow aEb$	11. $B \rightarrow D$
12. $E \rightarrow Eb$	12. $Q \rightarrow Qb$	12. $B \rightarrow D$	12. $B \rightarrow Ea$	12. $B \rightarrow \varepsilon$
Вариант 6	Вариант 7	Вариант 8	Вариант 9	Вариант 10
1. $A \rightarrow abBb$	1. $A \rightarrow EabE$	1. $T \rightarrow abETP$	1. $E \rightarrow EabC$	1. $S \rightarrow RMb$
2. $A \rightarrow C$	2. $A \rightarrow B$	2. $T \rightarrow aDE$	2. $E \rightarrow aDC$	2. $S \rightarrow ab$
3. $B \rightarrow Bab$	3. $B \rightarrow \varepsilon$	3. $T \rightarrow D$	3. $D \rightarrow \varepsilon$	3. $S \rightarrow T$
4. $B \rightarrow AaCD$	4. $B \rightarrow AEAb$	4. $D \rightarrow DTA b$	4. $D \rightarrow aEAb$	4. $M \rightarrow Q$
5. $B \rightarrow \varepsilon$	5. $D \rightarrow bDD$	5. $D \rightarrow b$	5. $D \rightarrow bB$	5. $M \rightarrow S$
6. $C \rightarrow BaE$	6. $D \rightarrow C$	6. $E \rightarrow \varepsilon$	6. $D \rightarrow C$	6. $T \rightarrow aRQb$
7. $C \rightarrow ab$	7. $C \rightarrow BCa$	7. $P \rightarrow BCa$	7. $C \rightarrow Bca$	7. $T \rightarrow RQ$
8. $D \rightarrow aEA$	8. $C \rightarrow CEb$	8. $P \rightarrow Cb$	8. $C \rightarrow \varepsilon$	8. $R \rightarrow RabR$
9. $D \rightarrow ADa$	9. $E \rightarrow abC$	9. $C \rightarrow abC$	9. $A \rightarrow abC$	9. $R \rightarrow STQ$
10. $D \rightarrow Eb$	10. $E \rightarrow Bbb$	10. $A \rightarrow Bbb$	10. $A \rightarrow DBb$	10. $R \rightarrow \varepsilon$
11. $E \rightarrow DaE$	11. $E \rightarrow aECb$	11. $B \rightarrow aECb$	11. $B \rightarrow Beb$	11. $Q \rightarrow SQa$
12. $E \rightarrow AE$	12. $E \rightarrow D$	12. $B \rightarrow D$	12. $B \rightarrow DCB$	12. $Q \rightarrow Qb$

К о н т р о л ь н ы е в о п р о с ы

1. Какие символы грамматики называются лишними? Как их устранить?

2. Что такое аннулирующий нетерминал? Как найти множество аннулирующих нетерминалов?

3. Какие правила называются цепными? В чём заключается «опасность» цепных правил? Как их устранить?

4. Какие правила называются самолеворекурсивными? Приведите пример леворекурсивной грамматики без самолеворекурсивных правил.

5. Дайте определения грамматикам, заданным в нормальных формах Хомского и Грейбах.

6. Дайте определение операторной грамматики.

Лабораторная работа № 3

Регулярные языки и конечные распознаватели

Цель работы: изучить основные способы задания регулярных языков, способы построения, алгоритмы преобразования, анализа и реализации конечных распознавателей.

З а д а н и е

1. Построить минимальный детерминированный конечный распознаватель заданного языка (см. варианты заданий).

2. Написать программу-распознаватель компиляционного и интерпретационного типа.

Исходные данные: строка.

Результат: “допустить” — если строка представляет собой цепочку заданного языка;

“отвергнуть” — в противном случае.

3. Написать программу, которая оставляет в исходном текстовом файле только те строки, которые представляют собой цепочки заданного языка.

4. Написать программу, которая исключает из исходного текстового файла строки, являющиеся цепочками заданного языка.

В а р и а н т ы з а д а н и й**Вариант 1**

Язык строковых констант.

Строковая константа может состоять из одной или более частей, соединённых между собой знаком + . Каждая часть начинается и заканчивается апострофом. Между апострофами могут быть любые символы, но если нужно включить апостроф, то его нужно повторить дважды. Строковая константа может содержать коды символов. Код символа — не более чем трёхзначное беззнаковое восьмеричное число, записанное после знака # . Последовательность кодов — часть константы.

Вариант 2

Язык строковых констант.

Строковая константа — последовательность символов, заключённая в кавычки. Последовательности символов `\n`, `\t`, `\b`, `\r`, `\f`, `\'`, `\"`, `\\` и символ `\`, за которым следует трёхразрядное восьмеричное число, считаются одним символом, а одиночный символ `\` недопустим.

Вариант 3

Язык прилагательных.

Язык прилагательных содержит прилагательные, независимо от рода и падежа. Принадлежность слова множеству прилагательных определяется окончанием. Кроме этого слова должны содержать не менее одной гласной и не менее одной согласной в части слова до окончания, последовательности согласных не должны превышать трёх букв, а последовательности гласных, не включая окончание — двух букв.

Вариант 4

Язык управляющих строк.

Управляющая строка начинается и заканчивается кавычками, содержит обычные символы и спецификации. Спецификация начинается символом `%` и заканчивается символом преобразования. Между `%` и символом преобразования может находиться формат, представляющий собой последовательность цифр, возможно, разделённых на две части точкой, перед которой или за которой может находиться знак минус. Символами преобразования являются: `D`, `O`, `X`, `U`, `C`, `S`, `E`, `F`, `G`. Символом `%` может начинаться только спецификация.

Вариант 5

Язык параметров оператора вывода.

Параметры оператора вывода заключаются в круглые скобки и разделяются запятыми. Параметр может представлять собой целочисленную или строковую константу или переменную, за которыми может следовать формат.

Строковая константа представляет собой последовательность символов, заключённую в кавычки. Целочисленная константа может быть десятичной, которая начинается не с нуля, восьмеричной, которая начинается с нуля, или шестнадцатеричной, которая начинается с буквы

Н. Константе может предшествовать знак + или – . Формат, следующий за константой, представляет собой двоеточие, за которым беззнаковое целое. Формат, следующий за переменной, может представлять собой формат, следующий за константой, или формат, следующий за константой, за которым следует такой же формат.

Вариант 6

Язык адресов.

Адрес представляет собой последовательность объектов, разделённых запятой.

Объект “область” представляет собой название области (слово), за которым следует слово “обл.” .

Объект “край” представляет собой название края (слово), за которым следует слово “край”.

Объект “город” представляет собой слово “г.”, за которым следует название города (последовательность слов).

Объект “район” представляет собой название района (слово), за которым следует слово “район”.

Объект “село” представляет собой слово “с.”, за которым следует название села (последовательность слов).

Объект “улица” представляет собой слово “ул.”, за которым следует название улицы (последовательность слов).

Объект “дом” представляет собой слово “д.”, за которым следует число и, возможно, буква.

Объект “квартира” представляет собой слово “кв.”, за которым следует число.

Каждый объект в адресе встречается только один раз. Начинаться адрес может объектами : “область” , “край” или “город”. Если адрес начинается объектами “область” или “край” , то за ними могут следовать объекты “город” или “район”. После объекта “район” может следовать “город” или “село”. За объектами “город” и “село” всегда следуют “улица” и “дом”, после чего может быть “квартира”.

Вариант 7

Язык директив компилятора.

Цепочка языка начинается парой символов { \$, заканчивается символом } и содержит последовательность директив, разделённых запятой. Директива представляет собой один из символов A , B , D , E , F , L , N , R , S , V , за которым следует символ + или – , или символ I ,

за которым следует имя файла (полное, начиная с имени диска и заканчивая расширением, или сокращённое), или символ M , за которым следует три беззнаковых целых числа, разделённых запятыми.

Вариант 8

Язык вещественных констант.

Вещественная константа может состоять из мантиссы и порядка, разделённых символом E . Мантисса может состоять из целой и дробной частей, разделённых точкой. Целая часть мантиссы и порядок — целые знаковые десятичные числа (знак может отсутствовать). Если отсутствует порядок и символ E , то могут отсутствовать целая или дробная часть мантиссы, но не обе сразу. Если присутствует порядок и символ E , то может отсутствовать дробная часть и точка.

Вариант 9

Язык слов, правильно разбитых на две части.

Части слова разделяются символом $-$. Слово считается правильно разбитым на две части, если:

— первая часть заканчивается последовательностью состоящей из согласной и гласной буквы, а вторая часть начинается гласной, за которой идёт хотя бы одна буква (буква y при этом рассматривается вместе с предшествующей гласной как единое целое);

— первая часть заканчивается последовательностью, состоящей из гласной и согласной буквы, а вторая часть начинается согласной и в оставшейся части имеется хотя бы одна гласная (буквы y и z вместе с предшествующей согласной рассматриваются как единое целое).

Вариант 10

Язык числовых констант.

Числовая константа может быть вещественной или целочисленной. Константа может начинаться со знака $+$ или $-$. Вещественная константа представляется только в десятичной системе счисления, а целочисленная — в задаваемой системе счисления.

Вещественная константа может быть представлена в форме с фиксированной или с плавающей точкой. В форме с фиксированной точкой константа состоит из двух частей — целой и дробной, разделённых точкой. Одна из частей может отсутствовать, но не обе сразу. В форме с плавающей точкой константа состоит из двух частей — мантиссы и

порядка, разделённых символом E . Мантисса — вещественное число в форме с фиксированной точкой, целая часть которой представляет собой одну цифру, не равную нулю. Порядок — целое число со знаком $+$ или $-$, не более чем четырёхзначное.

Целочисленная константа в десятичной системе представляет собой последовательность десятичных цифр. Целочисленная константа начинается с точки, после которой следует число $n \in \{2, 8, 10\}$, представляющее собой основание системы счисления, точка и последовательность цифр соответствующей системы счисления.

Контрольные вопросы

1. Какой язык называется регулярным?
2. Дайте определение правосторонней и левосторонней грамматики.
3. Опишите алгоритм преобразования правосторонней грамматики в автоматную правостороннюю.
4. Опишите алгоритм преобразования левосторонней грамматики в автоматную левостороннюю.
5. Приведите пример конечного распознавателя с ε -переходами. Преобразуйте его в конечный распознаватель без ε -переходов.
6. Приведите пример недетерминированного конечного распознавателя без ε -переходов. Преобразуйте его в детерминированный конечный распознаватель.
7. Постройте конечный распознаватель по правосторонней грамматике.
8. Постройте конечный распознаватель по левосторонней грамматике.
9. Приведите пример двух эквивалентных детерминированных конечных распознавателей с различным числом состояний. Проверьте их эквивалентность и минимизируйте их.
10. Приведите пример регулярного выражения. Постройте по нему детерминированный конечный распознаватель.
11. Какие регулярные выражения называются эквивалентными? Как определить эквивалентность регулярных выражений?
12. Представьте регулярным выражением язык, заданный правосторонней грамматикой.
13. Представьте левосторонней грамматикой язык, заданный регулярным выражением.
14. Напишите программу, которая табличное представление конечного детерминированного распознавателя преобразует в компиляционную программу реализации конечного распознавателя.

Лабораторная работа № 4

Нисходящая обработка контекстно-свободных языков

Цель работы: изучить и научиться применять нисходящие методы обработки формальных языков

З а д а н и е

1. Преобразовать исходную КС-грамматику в $LL(1)$ -грамматику (см. варианты заданий).
2. Определить множества ПЕРВЫХ для каждого символа $LL(1)$ -грамматики.
3. Определить множества СЛЕДУЮЩИХ для каждого символа $LL(1)$ -грамматики.
4. Определить множество ВЫБОРА для каждого правила $LL(1)$ -грамматики.
5. Написать программу-распознаватель методом рекурсивного спуска. Программа должна выводить последовательность номеров правил, применяемых при левом выводе обрабатываемой цепочки.
6. Сформировать наборы тестовых данных. Тестовые данные должны содержать цепочки, принадлежащие языку, заданному грамматикой, (допустимые цепочки) и цепочки, не принадлежащие языку. Для каждой допустимой цепочки построить дерево вывода и левый вывод. Каждое правило грамматики должно использоваться в выводах допустимых цепочек хотя бы один раз.
7. Обработать цепочки из набора тестовых данных (см. п.6) программой-распознавателем.
8. Построить нисходящий МП-распознаватель по $LL(1)$ -грамматике.
9. Написать программу-распознаватель, реализующую построенный нисходящий МП-распознаватель. Программа должна выводить на каждом шаге номер применяемого правила и промежуточную цепочку левого вывода.
10. Обработать цепочки из набора тестовых данных (см. п.6) программой-распознавателем.

В а р и а н т ы з а д а н и й *

Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5
1. $S \rightarrow OS$	1. $S \rightarrow S;O$	1. $S \rightarrow OS$	1. $S \rightarrow S;O$	1. $S \rightarrow SO$
2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$
3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow a[S]$	3. $O \rightarrow \{Y[S]\}$	3. $O \rightarrow \{[S]Y\}$
4. $O \rightarrow Y[S][S]$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow a[S][S]$	4. $O \rightarrow a=E$	4. $O \rightarrow a=E$
5. $O \rightarrow a=Y$	5. $O \rightarrow a=E$	5. $O \rightarrow a=E$	5. $Y \rightarrow (Y,Y)$	5. $Y \rightarrow (Y Y)$
6. $Y \rightarrow Y T$	6. $Y \rightarrow a=a$	6. $E \rightarrow E+T$	6. $Y \rightarrow \&(Y,Y)$	6. $Y \rightarrow (Y\&Y)$
7. $Y \rightarrow T$	7. $Y \rightarrow a<a$	7. $E \rightarrow T$	7. $Y \rightarrow !(Y)$	7. $Y \rightarrow !(Y)$
8. $T \rightarrow T\&P$	8. $Y \rightarrow !(Y)$	8. $T \rightarrow T*P$	8. $Y \rightarrow a$	8. $Y \rightarrow a$
9. $T \rightarrow P$	9. $E \rightarrow (E+E)$	9. $T \rightarrow P$	9. $E \rightarrow EE+$	9. $E \rightarrow +(E,E)$
10. $P \rightarrow (Y)$	10. $E \rightarrow (E*E)$	10. $P \rightarrow (E)$	10. $E \rightarrow EE^*$	10. $E \rightarrow *(E,E)$
11. $P \rightarrow !(Y)$	11. $E \rightarrow a$	11. $P \rightarrow a$	11. $E \rightarrow a$	11. $E \rightarrow a$
12. $P \rightarrow a$				
Вариант 6	Вариант 7	Вариант 8	Вариант 9	Вариант 10
1. $S \rightarrow OS$	1. $S \rightarrow S;O$	1. $S \rightarrow O;S$	1. $S \rightarrow OS$	1. $S \rightarrow S;O$
2. $S \rightarrow O$	2. $S \rightarrow O;$	2. $S \rightarrow O;$	2. $S \rightarrow O$	2. $S \rightarrow O;$
3. $O \rightarrow \{[S]Y\}$	3. $O \rightarrow Y[S]$	3. $O \rightarrow a[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$
4. $O \rightarrow \{Y[S]\}$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow a[S][S]$	4. $O \rightarrow \{Y[S]\}$	4. $O \rightarrow Y[S][S]$
5. $O \rightarrow a=E$	5. $O \rightarrow \{[S]Y\}$	5. $O \rightarrow a=E$	5. $O \rightarrow a=E$	5. $O \rightarrow \{[S]Y\}$
6. 6. $Y \rightarrow a=a$	6. $O \rightarrow \{Y[S]\}$	6. $E \rightarrow E+T$	6. $Y \rightarrow a=a$	6. $O \rightarrow \{Y[S]\}$
7. $Y \rightarrow a<a$	7. $O \rightarrow a=Y$	7. $E \rightarrow T$	7. $Y \rightarrow a<a$	7. $O \rightarrow a=Y$
8. $Y \rightarrow !(Y)$	8. $Y \rightarrow (Y Y)$	8. $T \rightarrow T*P$	8. $Y \rightarrow !(Y)$	8. $Y \rightarrow (Y Y)$
9. $E \rightarrow +(E,E)$	6. $Y \rightarrow (Y\&Y)$	9. $T \rightarrow P$	9. $E \rightarrow (E+E)$	6. $Y \rightarrow (Y\&Y)$
10. $E \rightarrow *(E,E)$	7. $Y \rightarrow !(Y)$	10. $P \rightarrow (E)$	10. $E \rightarrow (E*E)$	7. $Y \rightarrow !(Y)$
11. $E \rightarrow a$	11. $Y \rightarrow (a=a)$	11. $P \rightarrow -(E)$	11. $E \rightarrow -(E)$	11. $Y \rightarrow a$
	12. $Y \rightarrow (a<a)$	12. $P \rightarrow a$	12. $E \rightarrow a$	

(*) Во всех вариантах символ a обозначает букву латинского алфавита.

К о н т р о л ь н ы е в о п р о с ы

1. Дайте определение $LL(1)$ -грамматике.
2. Из каких элементов состоит множество ПЕРВЫХ для символа грамматики?
3. Из каких элементов состоит множество СЛЕДУЮЩИХ для символа грамматики?
4. Как найти множество ПЕРВЫХ для цепочки, состоящей из терминалов и нетерминалов?
5. Как найти множество ВЫБОРА для правила грамматики?
6. В чем суть метода рекурсивного спуска?

Лабораторная работа № 5
**Восходящая обработка контекстно-свободных языков
методом «перенос-опознание»**

Цель работы: изучить и научиться применять восходящие методы обработки формальных языков типа «перенос-опознание»

З а д а н и е

1. Преобразовать исходную КС-грамматику (см. варианты заданий) в грамматику без правой рекурсии.
2. Определить множества ПЕРВЫХ для каждого символа грамматики.
3. Определить множества СЛЕДУЮЩИХ для каждого символа грамматики.
4. Построить управляющую таблицу восходящего МП-распознавателя типа «перенос-опознание». Если не удастся построить управляющую таблицу МП-распознавателя без конфликтов типа «перенос-опознание», то преобразовать грамматику в грамматику, допускающую построение МП-распознавателя без конфликтов типа «перенос-опознание» и выполнить п.п. 2 — 4.
5. Построить процедуру опознания в виде конечного автомата.
6. Написать программу-распознаватель, реализующую построенный восходящий МП-распознаватель. После выполнения свертки программа должна выводить номер правила, по которому выполнялась свертка, и строку, состоящую из содержимого магазина и необработанной части входной цепочки.
7. Сформировать наборы тестовых данных. Тестовые данные должны содержать цепочки, принадлежащие языку, заданному грамматикой, (допустимые цепочки) и цепочки, не принадлежащие языку. Для каждой допустимой цепочки построить дерево вывода и правый вывод. Каждое правило грамматики должно использоваться в выводах допустимых цепочек хотя бы один раз.
8. Обработать цепочки из набора тестовых данных программой-распознавателем.

В а р и а н т ы з а д а н и й *

Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5
1. $S \rightarrow OS$	1. $S \rightarrow OS$	1. $S \rightarrow S; O$	1. $S \rightarrow SO$	1. $S \rightarrow S; O$
2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$
3. $O \rightarrow \{[S]Y\}$	3. $O \rightarrow a[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow \{[S]Y\}$	3. $O \rightarrow \{Y[S]\}$
4. $O \rightarrow \{Y[S]\}$	4. $O \rightarrow a[S][S]$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow a=E$	4. $O \rightarrow a=E$
5. $O \rightarrow a=E$	5. $O \rightarrow a=E$	5. $O \rightarrow a=E$	5. $Y \rightarrow (Y Y)$	5. $Y \rightarrow (Y, Y)$
6. $Y \rightarrow a=a$	6. $E \rightarrow E+T$	6. $Y \rightarrow a=a$	6. $Y \rightarrow (Y \& Y)$	6. $Y \rightarrow \&(Y, Y)$
7. $Y \rightarrow a < a$	7. $E \rightarrow T$	7. $Y \rightarrow a < a$	7. $Y \rightarrow !(Y)$	7. $Y \rightarrow !(Y)$
8. $Y \rightarrow !(Y)$	8. $T \rightarrow T^*P$	8. $Y \rightarrow !(Y)$	8. $Y \rightarrow a$	8. $Y \rightarrow a$
9. $E \rightarrow +(E, E)$	9. $T \rightarrow P$	9. $E \rightarrow (E+E)$	9. $E \rightarrow +(E, E)$	9. $E \rightarrow EE+$
10. $E \rightarrow *(E, E)$	10. $P \rightarrow (E)$	10. $E \rightarrow (E^*E)$	10. $E \rightarrow *(E, E)$	10. $E \rightarrow EE^*$
11. $E \rightarrow a$	11. $P \rightarrow a$	11. $E \rightarrow a$	11. $E \rightarrow a$	11. $E \rightarrow a$
Вариант 6	Вариант 7	Вариант 8	Вариант 9	Вариант 10
1. $S \rightarrow O; S$	1. $S \rightarrow O; S$	1. $S \rightarrow S; O$	1. $S \rightarrow S; O$	1. $S \rightarrow OS$
2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$
3. $O \rightarrow Y[S]$	3. $O \rightarrow a[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$
4. $O \rightarrow Y[S][S]$	4. $O \rightarrow a[S][S]$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow \{Y[S]\}$
5. $O \rightarrow a=Y$	5. $O \rightarrow a=E$	5. $O \rightarrow \{[S]Y\}$	5. $O \rightarrow \{[S]Y\}$	5. $O \rightarrow a=E$
6. $Y \rightarrow Y T$	6. $E \rightarrow E+T$	6. $O \rightarrow \{Y[S]\}$	6. $O \rightarrow \{Y[S]\}$	6. $Y \rightarrow a=a$
7. $Y \rightarrow T$	7. $E \rightarrow T$	7. $O \rightarrow a=Y$	7. $O \rightarrow a=Y$	7. $Y \rightarrow a < a$
8. $T \rightarrow T \& P$	8. $T \rightarrow T^*P$	8. $Y \rightarrow (Y Y)$	8. $Y \rightarrow (Y Y)$	8. $Y \rightarrow !(Y)$
9. $T \rightarrow P$	9. $T \rightarrow P$	6. $Y \rightarrow (Y \& Y)$	6. $Y \rightarrow (Y \& Y)$	9. $E \rightarrow (E+E)$
10. $P \rightarrow (Y)$	10. $P \rightarrow (E)$	7. $Y \rightarrow !(Y)$	7. $Y \rightarrow !(Y)$	10. $E \rightarrow (E^*E)$
11. $P \rightarrow !(Y)$	11. $P \rightarrow -(E)$	11. $Y \rightarrow (a=a)$	11. $Y \rightarrow a$	11. $E \rightarrow -(E)$
12. $P \rightarrow a$	12. $P \rightarrow a$	12. $Y \rightarrow (a < a)$		12. $E \rightarrow a$

(*) Во всех вариантах символ a обозначает букву латинского алфавита.

К о н т р о л ь н ы е в о п р о с ы

1. Что такое основа и основывающее правило?
2. Опишите принцип работы восходящего МП-распознавателя типа «перенос-опознание».
3. По каким классам грамматик можно построить детерминированный МП-распознаватель типа «перенос-опознание»?
4. Сформулируйте правила построения управляющей таблицы МП-распознавателя типа «перенос-опознание».
5. Что такое отношение ПОД и как его можно использовать для разрешения конфликтов типа «опознание-опознание»?

Лабораторная работа № 6

Восходящая обработка контекстно-свободных языков

методом «перенос-свертка»

Цель работы: изучить и научиться применять восходящие методы обработки формальных языков типа «перенос-свертка»

З а д а н и е

1. Пополнить исходную КС-грамматику (см. варианты заданий).
2. Построить граф переходов на множестве $LR(0)$ -ситуаций. Граф переходов представить в виде таблицы. Ниже приведен пример фрагмента таблицы графа переходов:

Номер ситуации	Ситуация	Символ перехода	Ситуация перехода
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	$E \rightarrow E + T$ $T \rightarrow T * P$ $T \rightarrow P$ $P \rightarrow (E)$ $P \rightarrow a$	T P $($ a	9 3 4 5
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	$E \rightarrow E + T$ $T \rightarrow T * P$	*	7
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Если в ситуациях присутствуют конфликты (например, в ситуации 9 в приведенной выше таблице имеется конфликт типа «перенос-свертка»), то найти способы разрешения конфликтов.

3. Построить таблицу восходящего МП-распознавателя типа «перенос-свертка».

4. Написать программу-распознаватель, обрабатывающую таблицу восходящего МП-распознавателя типа «перенос-свертка». После выполнения свертки программа должна выводить номер правила, по которому выполнялась свертка.

5. Сформировать наборы тестовых данных.

6. Обработать цепочки из набора тестовых данных программой-распознавателем.

В а р и а н т ы з а д а н и й *

Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5
1. $S \rightarrow OS$	1. $S \rightarrow OS$	1. $S \rightarrow SO$	1. $S \rightarrow S;O$	1. $S \rightarrow S;O$
2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O;$
3. $O \rightarrow a[S]$	3. $O \rightarrow \{[S]Y\}$	3. $O \rightarrow \{[S]Y\}$	3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$
4. $O \rightarrow a[S][S]$	4. $O \rightarrow \{Y[S]\}$	4. $O \rightarrow a=E$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow Y[S][S]$
5. $O \rightarrow a=E$	5. $O \rightarrow a=E$	5. $Y \rightarrow (Y Y)$	5. $O \rightarrow a=E$	5. $O \rightarrow \{[S]Y\}$
6. $E \rightarrow E+T$	6. $Y \rightarrow a=a$	6. $Y \rightarrow (Y&Y)$	6. $Y \rightarrow a=a$	6. $O \rightarrow \{Y[S]\}$
7. $E \rightarrow T$	7. $Y \rightarrow a < a$	7. $Y \rightarrow !(Y)$	7. $Y \rightarrow a < a$	7. $O \rightarrow a=Y$
8. $T \rightarrow T^*P$	8. $Y \rightarrow !(Y)$	8. $Y \rightarrow a$	8. $Y \rightarrow !(Y)$	8. $Y \rightarrow (Y Y)$
9. $T \rightarrow P$	9. $E \rightarrow +(E,E)$	9. $E \rightarrow +(E,E)$	9. $E \rightarrow (E+E)$	6. $Y \rightarrow (Y&Y)$
10. $P \rightarrow (E)$	10. $E \rightarrow *(E,E)$	10. $E \rightarrow *(E,E)$	10. $E \rightarrow (E^*E)$	7. $Y \rightarrow !(Y)$
11. $P \rightarrow a$	11. $E \rightarrow a$	11. $E \rightarrow a$	11. $E \rightarrow a$	11. $Y \rightarrow a$
Вариант 6	Вариант 7	Вариант 8	Вариант 9	Вариант 10
1. $S \rightarrow O;S$	1. $S \rightarrow O;S$	1. $S \rightarrow OS$	1. $S \rightarrow S;O$	1. $S \rightarrow S;O$
2. $S \rightarrow O;$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O$	2. $S \rightarrow O;$
3. $O \rightarrow a[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow Y[S]$	3. $O \rightarrow \{Y[S]\}$	3. $O \rightarrow Y[S]$
4. $O \rightarrow a[S][S]$	4. $O \rightarrow Y[S][S]$	4. $O \rightarrow \{Y[S]\}$	4. $O \rightarrow a=E$	4. $O \rightarrow Y[S][S]$
5. $O \rightarrow a=E$	5. $O \rightarrow a=Y$	5. $O \rightarrow a=E$	5. $Y \rightarrow (Y,Y)$	5. $O \rightarrow \{[S]Y\}$
6. $E \rightarrow E+T$	6. $Y \rightarrow Y T$	6. $Y \rightarrow a=a$	6. $Y \rightarrow \&(Y,Y)$	6. $O \rightarrow \{Y[S]\}$
7. $E \rightarrow T$	7. $Y \rightarrow T$	7. $Y \rightarrow a < a$	7. $Y \rightarrow !(Y)$	7. $O \rightarrow a=Y$
8. $T \rightarrow T^*P$	8. $T \rightarrow T&P$	8. $Y \rightarrow !(Y)$	8. $Y \rightarrow a$	8. $Y \rightarrow (Y Y)$
9. $T \rightarrow P$	9. $T \rightarrow P$	9. $E \rightarrow (E+E)$	9. $E \rightarrow EE+$	6. $Y \rightarrow (Y&Y)$
10. $P \rightarrow (E)$	10. $P \rightarrow (Y)$	10. $E \rightarrow (E^*E)$	10. $E \rightarrow EE^*$	7. $Y \rightarrow !(Y)$
11. $P \rightarrow -(E)$	11. $P \rightarrow !(Y)$	11. $E \rightarrow -(E)$	11. $E \rightarrow a$	11. $Y \rightarrow (a=a)$
12. $P \rightarrow a$	12. $P \rightarrow a$	12. $E \rightarrow a$		12. $Y \rightarrow (a < a)$

(*) Во всех вариантах символ a обозначает букву латинского алфавита.

К о н т р о л ь н ы е в о п р о с ы

1. Опишите принцип работы восходящего МП-распознавателя типа «перенос-свертка».

2. Чем отличается МП-распознаватель типа «перенос-свертка» от МП-распознавателя типа «перенос-опознание»?

3. По каким классам грамматик можно построить детерминированный МП-распознаватель типа «перенос-свертка»?

4. Сформулируйте правила построения граф переходов на множестве $LR(0)$ -ситуаций.

5. Какие конфликты существуют в МП-распознавателях типа «перенос-свертка» и какими способами их можно разрешить?

Учебное издание

Теория автоматов и формальных языков

*Методические указания к выполнению лабораторных работ
для студентов, обучающихся по направлению бакалавриата
09.03.04 «Программная инженерия»*

Составитель Рязанов Юрий Дмитриевич

Подписано в печать	Формат 60x84/16. Усл.печ.л. 1,4. Уч.-изд.л. 1,5.
Тираж экз.	Заказ №262 Цена
Отпечатано в Белгородском государственном технологическом университете им. В.Г.Шухова 308012, г.Белгород, Костюкова, 46	