МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра вычислительной техники



**Лабораторная Работа №3**

**по дисциплине:** *Теория формальных языков и компиляторов*

**на тему:** *«Синтаксис языков программирования. Формальные грамматики»*

Вариант № 31233114

|  |  |
| --- | --- |
| Выполнила: | Проверил: |
| Студентка гр. *АВТ-709*, *АВТФ* | *К.т.н., доцент* |
| *Черданцева Кристина* | *Малявко Александр Антонович* |
| «\_\_\_» \_\_\_\_\_\_ 2020г. | «\_\_\_» \_\_\_\_\_\_ 2020г. |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| (подпись) | (подпись) |

Новосибирск

2020

**Оглавление**

[Цель работы 3](#_Toc36652743)

[Краткое описание метаязыка формальных грамматик 3](#_Toc36652744)

[Свойства грамматик из примеров и символов этих грамматик 3](#_Toc36652745)

[Матричное представление отношений предшествования и последования для символов изученных грамматик 4](#_Toc36652746)

[Разработанная грамматика 6](#_Toc36652747)

[Краткое описание разработанной грамматики 7](#_Toc36652748)

[Выводы 7](#_Toc36652749)

# **Цель работы**

Изучение основных понятий метаязыка формальных грамматик, свойств грамматик и нетерминальных символов, рекурсивности и однозначности грамматик, недостижимости, бесплодности, аннулируемости и рекурсивности нетерминальных символов, отношений предшествования и последования между символами, приобретение навыков эквивалентных преобразований формальных грамматик, освоение технологий разработки формальных грамматик.

# **Краткое описание метаязыка формальных грамматик**

Формальной грамматикой G называют совокупность

G = {At, An, S, P},

состоящая:

* из алфавита терминальных символов At;
* алфавита нетерминальных символов An;
* начального нетерминального символа S;
* системы правил подстановки P.

Алфавит терминальных символов At есть конечное множество всех слов языка, порождаемого данной грамматикой. Понятие «терминальный» в данном случае означает неразложимость, элементарность таких символов с точки зрения синтаксических правил.

Алфавит нетерминальных символов An есть конечное множество названий синтаксических конструкций. Нетерминальные символы используются только в метаязыке, на котором описывается язык программирования, никакой нетерминальный символ не может появиться в тексте правильной программы.

Начальный нетерминальный символ S есть один из нетерминальных символов. Этим символом обычно обозначается наиболее общая синтаксическая конструкция.

Система правил подстановки P есть конечное множество пар цепочек вида α : β, причем цепочка α должна содержать хотя бы один нетерминальный символ.

Каждая пара цепочек называется правилом подстановки и определяет возможный способ замены левой части правила на его правую часть.

# **Свойства грамматик из примеров и символов этих грамматик**

**Свойства грамматик**

1. Рекурсивность

Нетерминальный символ Х называется рекурсивным, если из него могут быть выведены цепочки, содержащие сам этот символ, т. е.

Х ⇒ μ Х η,

где μ и η – произвольные цепочки символов.

Грамматика называется рекурсивной, если рекурсивен хотя бы один нетерминальный символ, и нерекурсивной в противном случае.

Примеры рекурсивности из грамматик Ga1 и Ga2:

S: X Y

Y: +S

S: S + T

S: S – T

1. Однозначность

Грамматика называется однозначной, если любое правильное предложение порождаемого ею языка имеет единственное дерево грамматического разбора, и неоднозначной в противном случае.

**Свойства символов грамматик**

1. Аннулируемость

Нетерминальный символ называется аннулируемым, если из него может быть выведена пустая цепочка символов. В противном случае нетерминальный символ называется неаннулируемым.

1. Недостижимость

Символ называется недостижимым, если он не появляется ни в одной цепочке символов, выводимой из начального нетерминала грамматики.

1. Бесплодность

Нетерминальный символ называется бесплодным, если из него не может быть выведена ни одна цепочка, состоящая только из терминальных символов.

# **Матричное представление отношений предшествования и последования для символов изученных грамматик**



Рис. 1. Отношение предшествования для грамматики из примера 4



Рис. 2. Отношение предшествования для грамматики из примера 3

Предшественником некоторого символа Х называется символ, с которого начинается цепочка, выводимая из Х. Считается, что любой символ является предшественником самого себя, т.е. учитываются выводы длины 0.



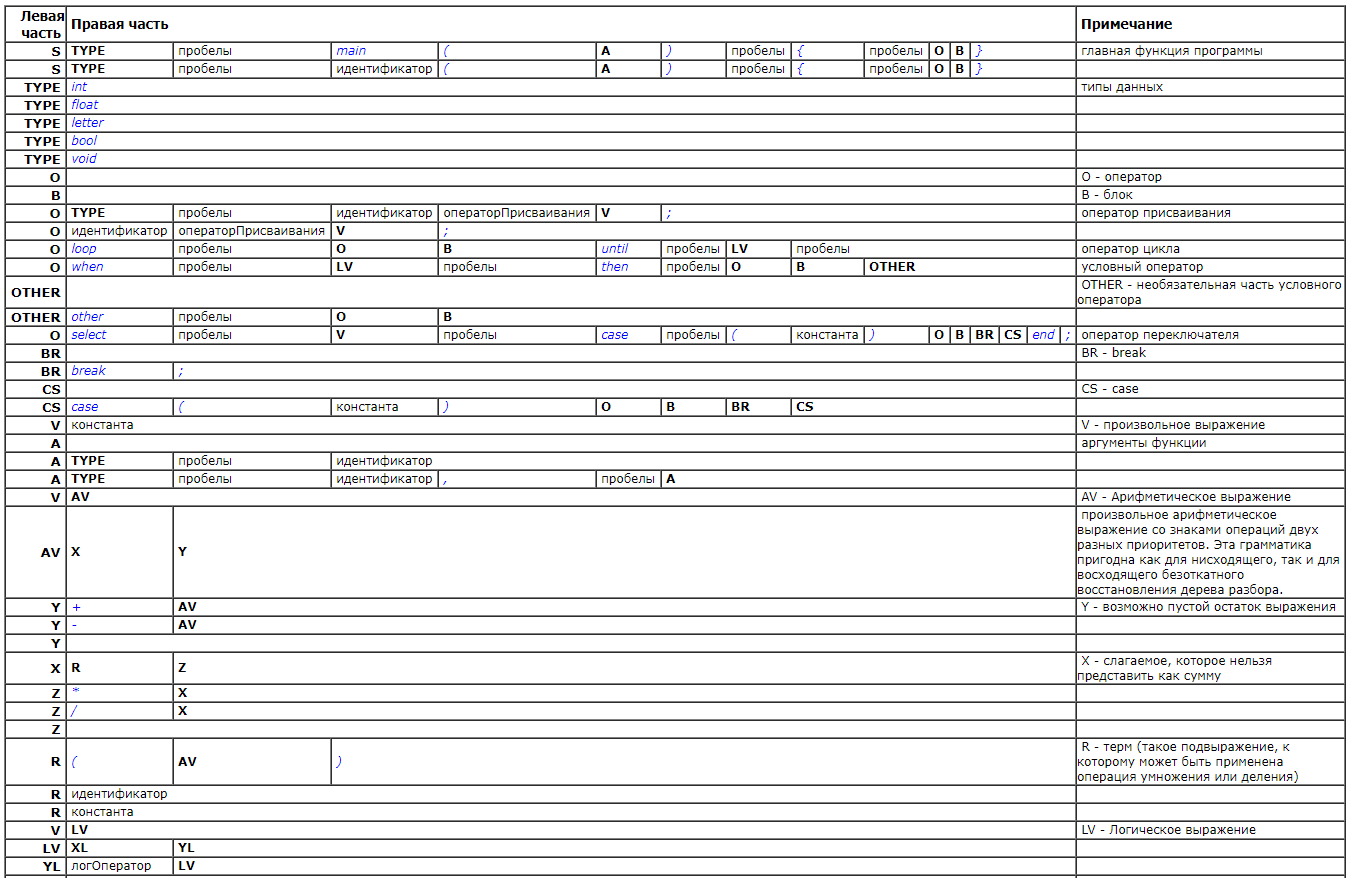
Рис. 3. Отношение последования для грамматики из примера 4



Рис. 4. Отношение последования для грамматики из примера 3

Символ Y называется последователем символа Х, если хотя бы в одной цепочке η, выводимой из начального нетерминала грамматики, символ Y непосредственно следует за Х.

# **Разработанная грамматика**



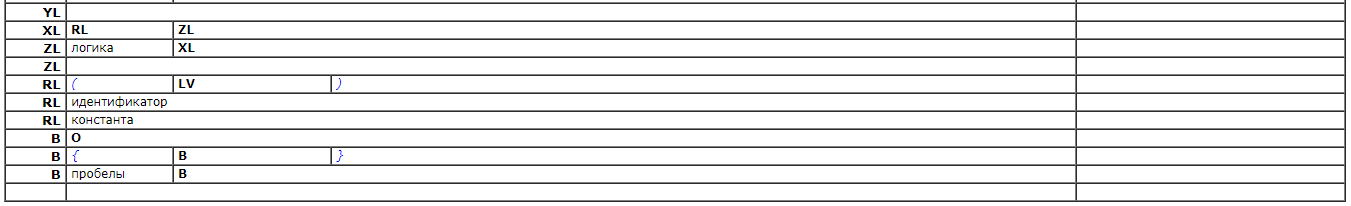


Рис. 5. Грамматика заданного языка

# **Краткое описание разработанной грамматики**

Начальный нетерминальный символ – S. В качестве S могут быть использованы только функции, либо функция main, либо функция с произвольным названием.

Далее описана грамматика для оператора присваивания, цикла, условного оператора и оператора переключателя. Во всех операторах могут присутствовать выражения (V). Выражения представляют из себя константы, арифметические выражения или логические выражения.

Блок операторов еще не доработан. Пока корректно работает только один оператор в программе.

В грамматике описаны типы данных, которые можно использовать в программе. В лексику еще не внесены слова для логического типа данных.

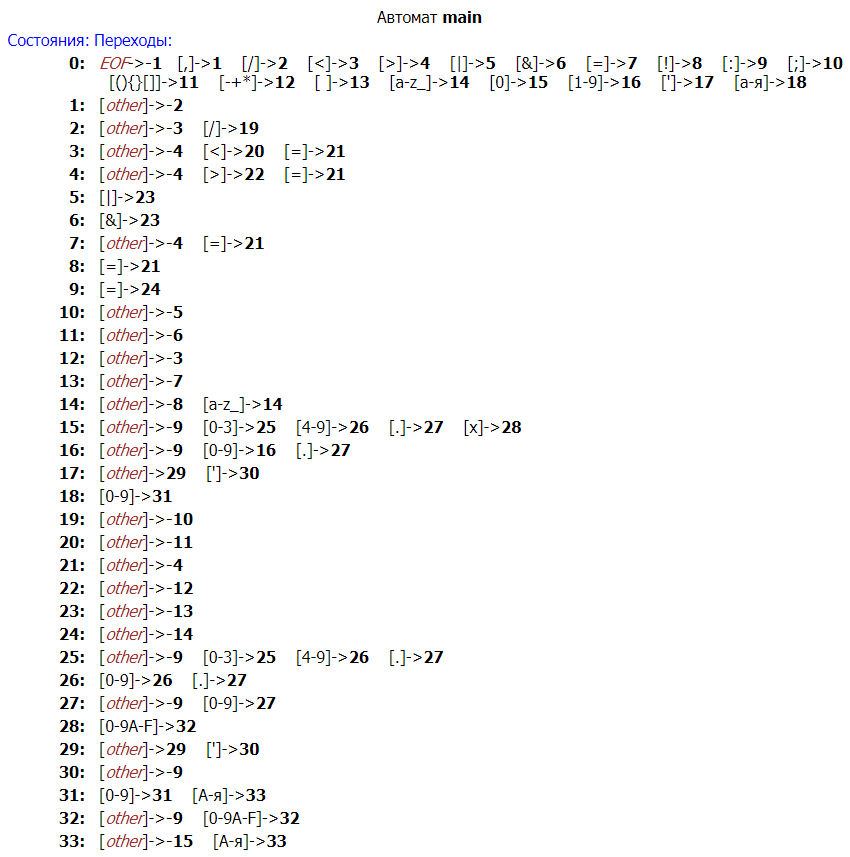
# **Выводы**

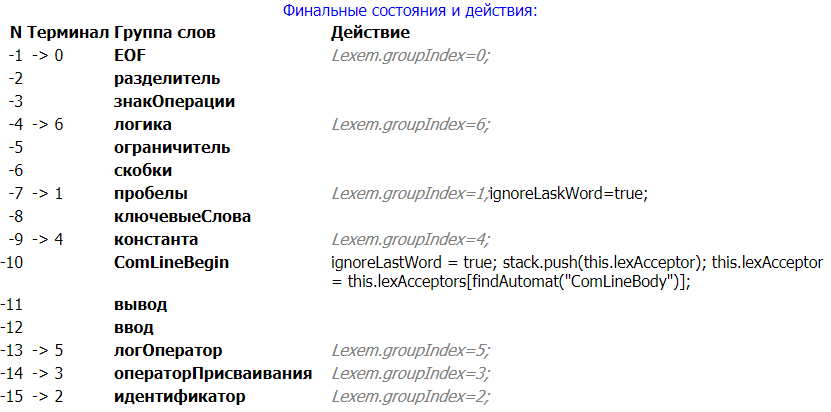
Изучены основные понятия метаязыка формальных грамматик, свойства грамматик и нетерминальных символов, рекурсивности и однозначности грамматик, недостижимости, бесплодности, аннулируемости и рекурсивности нетерминальных символов, отношений предшествования и последования между символами, освоены технологии разработки формальных грамматик.

**Защита лабораторной работы №2**

История разбора лексических автоматов по обработке фрагмента int а16а, а4а := 0;.

1. **Графовый**





0: [i] → 14

14: [n] → 14

14: [t] → 14

14: [ ] → -8

-8: ключевыеСлова

0: [ ] → 13

13: [а] → -7

-7: пробелы

0: [а] → 18

18: [1] → 31

31: [6] → 31

31: [а] → 33

33: [,] → -15

-15: идентификатор

0: [,] → 1

1: [ ] → -2

-2: разделитель

0: [ ] → 13

13: [а] → -7

-7: пробелы

0: [а] → 18

18: [4] → 31

31: [а] → 33

33: [:] → -15

-15: идентификатор

0: [:] → 9

9: [=] → 24

24: [0] → -14

-14: операторПрисваивания

0: [0] → 15

15: [;] → -9

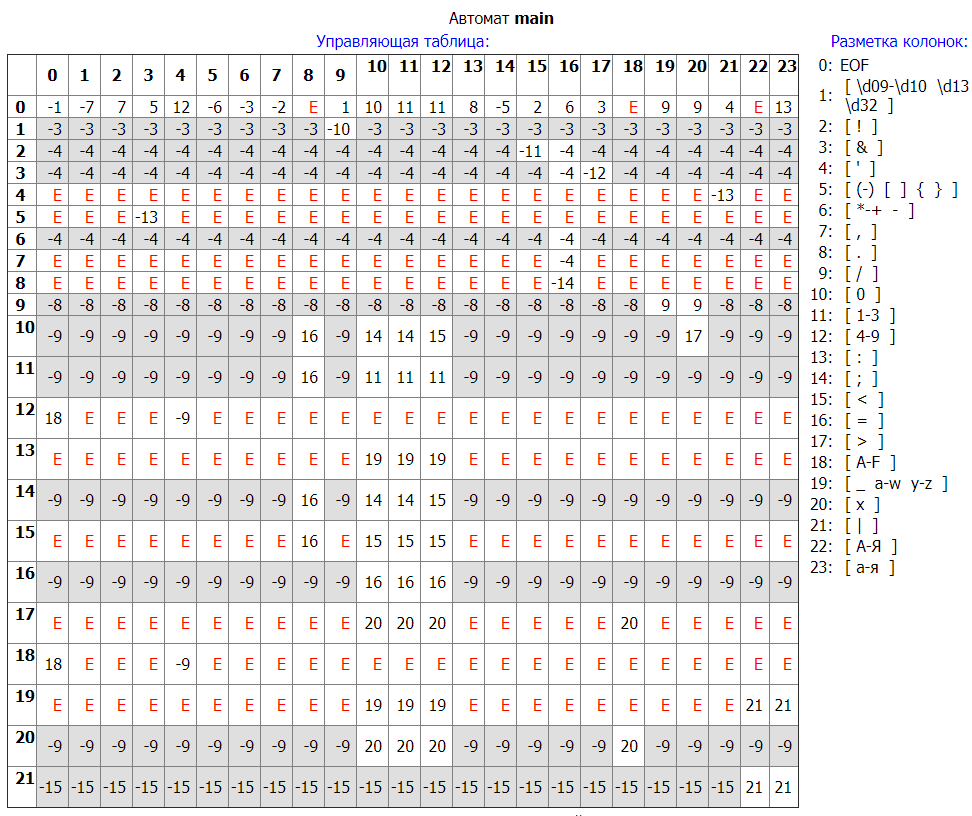
-9: константа

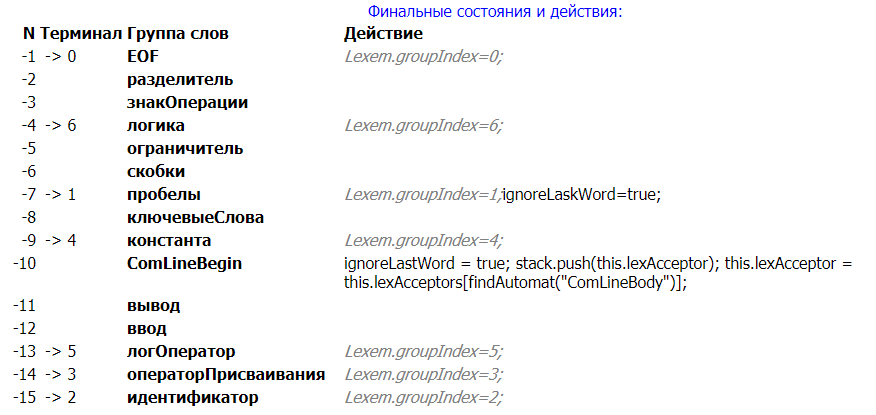
0: [;] → 10

10: → -5

-5: ограничитель

1. **Табличный**





0: [i] (19) → 9

9: [n] (19) → 9

9: [t] (19) → 9

9: [ ] (1) → -8

-8: ключевыеСлова

0: [ ] (1) → -7

-7: пробелы

0: [a] (23) → 13

13: [1] (11) → 19

19: [6] (12) → 19

19: [a] (23) → 21

21: [,] (7) → -15

-15: идентификатор

0: [,] (7) → -2

-2: разделитель

0: [ ] (1) → -7

-7: пробелы

0: [a] (23) → 13

13: [4] (12) → 19

19: [a] (23) → 21

21: [:] (13) → -15

-15: идентификатор

0: [:] (13) → 8

8: [=] (16) → -14

-14: операторПрисваивания

0: [0] (10) → 10

10: [;] (14) → -9

-9: константа

0: [;] (14) → -5

-5: ограничитель