МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра вычислительной техники



**ОТЧЁТ**

**по лабораторной работе №3**

***«*Синтаксис языков программирования. Формальные грамматики *»***

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

Вариант: 24443144

Выполнила:Проверил:

Студентка гр. АВТ-709 *« к.т.н., доцент »*

*« Васильева Н.В. » « Малявко А.А. »*

«\_\_\_» \_\_\_\_\_\_ 2020г.«\_\_\_» \_\_\_\_\_\_ 2020 г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (подпись)

Новосибирск

2020

Цели 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.

**Вариант: 24443144**

1. *Идентификаторы*

|  |
| --- |
| **2** |
| <бБ><пЦ><бБ>  ( d23U, N1q, x15y, …) |

1. *Константы*

|  |
| --- |
| **4** |
| целые по основанию 10 и указанному в константе;  вещественные;  символьные |

1. *Объявления примитивных типов (целое, вещественное, символьное):*

|  |
| --- |
| **4** |
| **card[inal][\_u]**  **double**  **litera** |

1. *Оператор присваивания:*

|  |
| --- |
| **4** |
| **put**<В> **to** <И> ; |

1. *Условный оператор:*

|  |
| --- |
| **3** |
| **when** <ЛВ> **then** <ОБ> **[ other** <ОБ> **]** |

1. *Оператор цикла:*

|  |
| --- |
| **1** |
| **loop** <ОБ> **until** <ЛВ> |

1. *Оператор переключателя*

|  |
| --- |
| **4** |
| **??** <В> **{ ?** <К> **:** <ОБ> **[gout;]** … **[ ?~** **:** <ОБ> **]** **}** |

1. *Формат псевдокода*

|  |
| --- |
| **4** |
| Триады <Код> <ОП> <Р> |

# Краткое описание метаязыка формальных грамматик

# Формальная грамматика G состоит из:

* алфавита терминальных слов **At,**
* алфавита нетерминальных слов **An,**
* начального нетерминального символа **S**,
* системы правил подстановки **P**.

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

Алфавит терминальных слов – конечное множество всех слов языка, порождаемого данной грамматикой. Пример: идентификатор, константа, +, -, / и тд

Алфавит нетерминальных слов – конечное множество названий синтаксических конструкций. Пример: <выражение>, <предложение>, FUNCTION

Начальный нетерминальный символ – один из нетерминальных символов, обычно обозначается наиболее общая синтаксическая конструкция, например: <правильная программа>.

Системы правил подстановки – конечное множество пар цепочек вида α:β

# Свойства грамматик из примеров и символов этих грамматик

Свойства контекстно-свободных грамматик:

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

Х–>μXη,

где μ,η - произвольные цепочки символов.

Если хотя бы в одном таком выводе цепочка μ/η пуста, то нетерминал Х называется леворекурсивным/праворекурсивным. Рекурсия бывает прямой и косвенной.

* Грамматика называется **однозначной**, если любое правильное предложение порождаемого ею языка имеет единственное дерево грамматического разбора, иначе неоднозначная.

Свойства символов грамматики:

* Нетерминальный символ называется **аннулируемым**, если из него может быть выведена пустая цепочка символов, иначе неаннулируемым.
* Символ называется **недостижимым**, если он не появляется ни в одной цепочке символов, выводимого из начального нетерминала грамматики.
* 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

Матрица полного отношения предшествования содержит в каждой строке отметки как для терминальных, так и нетерминальных символов. В действительности интерес представляют только терминалы-предшественники нетерминальных символов.



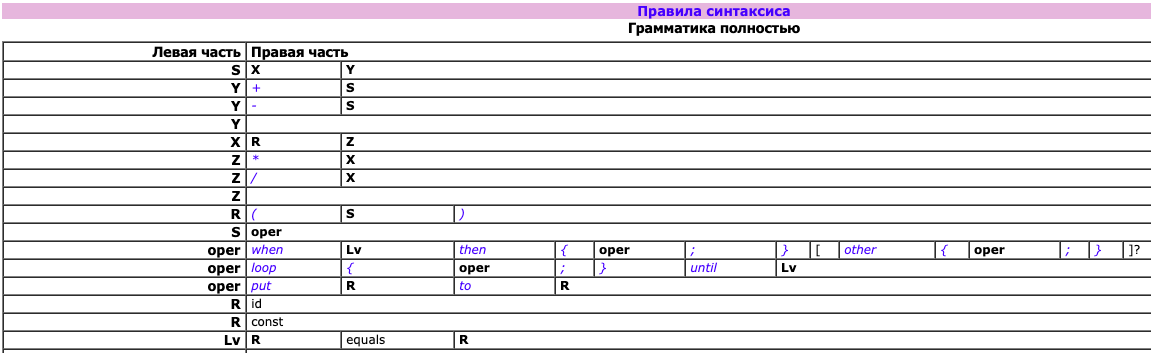
*Рисунок 1. Матрица отношения предшествования*

Для решения задачи синтаксического анализа интерес представляют только множества последователей нетерминальных символов, для их вычисления приходится определять множества последователей всех символов грамматики.

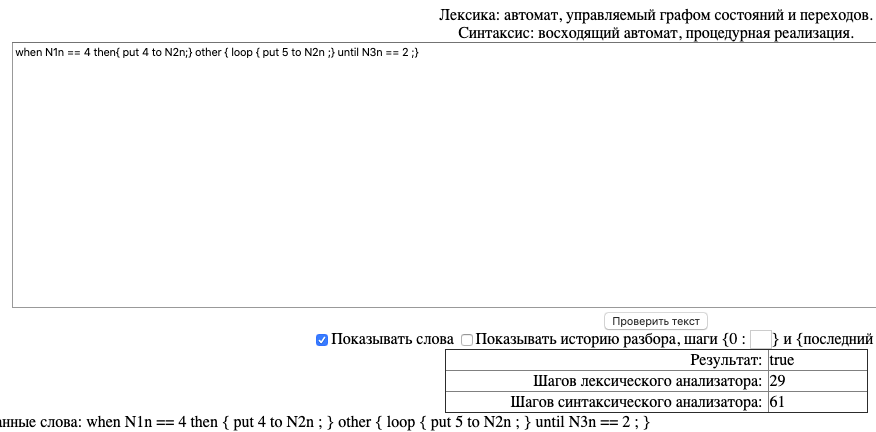


*Рисунок 2. Матрица отношения последования*

# Расширение одной из изученных грамматик до части языка, заданного на курсовую работу



*Рисунок 3. Расширение одной из изученных грамматик*

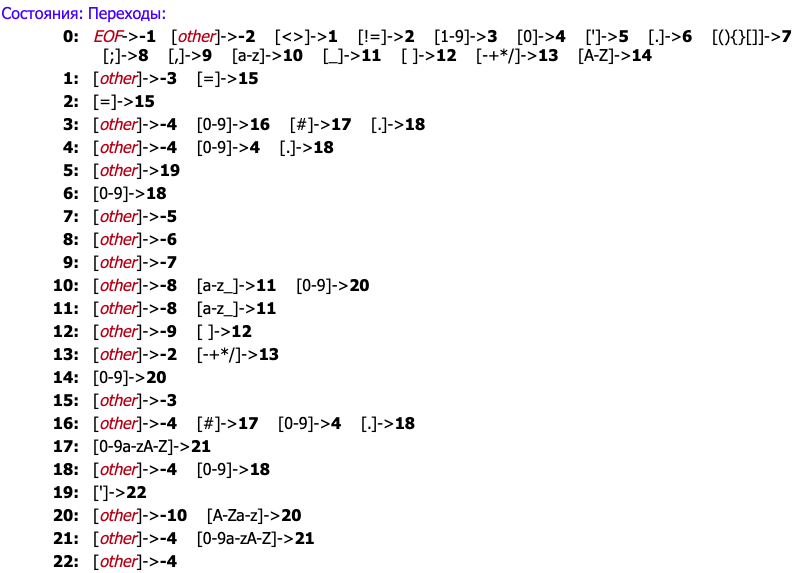


*Рисунок 4. Построенный транслятор для проверки правильности написанных операторов*

# Вывод

Изучили основные понятия 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, приобрели навыкы эквивалентных преобразований формальных грамматик, освоили технологи разработки формальных грамматик.

# Приложения



**put N1n/I1i to N1n**

p – это [a-z], значит переходим в состояние 10

u – это [a-z], переходим в состояние 11

t - это [a-z], переходим в состояние 11

[ ] -> -8 (финальное состояние keywords), начинаем с 0, [ ] -> 12

N -> -9 (финальное состояние space), начинаем с 0, N -> 14

1 -> 20

n -> 20

/ -> -10 (финальное состояние id), начинаем с 0, / -> 13

I -> -2 (финальное состояние singOperation), начинаем с 0, I ->14

1 -> 20

i -> 20

[ ] -> -10 (финальное состояние id), начинаем с 0, [ ] -> 12

t -> -9 (финальное состояние space), начинаем с 0, t ->10

o -> 11

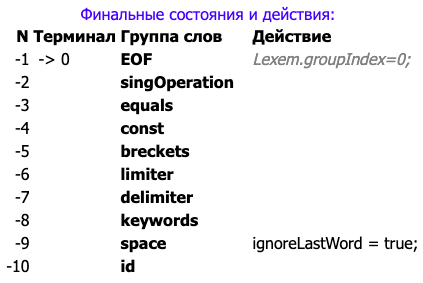
[ ] -> -8 (финальное состояние keywords), начинаем с 0, [ ] -> 12

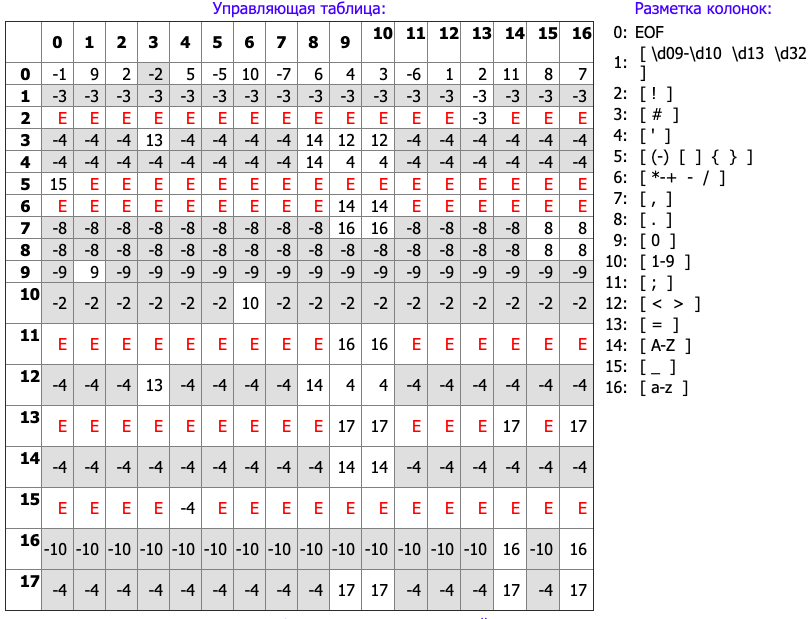
N -> -9 (финальное состояние space), начинаем с 0, N -> 14

1 -> 20

n -> 20

EOF -> -> -10 (финальное состояние id), начинаем с 0, EOF -> -1 (финальное состояние EOF)





**put N1n/I1i to N1n**

p – это [a-z], значит ищем в 0 строке колонку 16, в этой колонке стоит цифра 7, переходим к 7-ой строке

u - это [a-z], ищем в 7-ой строке колонку 16, видим цифру 8, переходим к 8-ой строке.

t-> 8

[ ]-> -8 (финальное состояние keywords), начинаем вновь с 0 строки, [ ]-> 9

N-> -9 (финальное состояние space), начинаем вновь с 0 строки, N-> 11

1-> 16

n-> 16

/ -> -10 (финальное состояние id), начинаем с 0 строки, / -> 10

I -> -2 (финальное состояние singOperation ), начинаем с 0 строки, I -> 11

1-> 16

i -> 16

[ ] -> -10 (финальное состояние id), начинаем с 0 строки, [ ] -> 9

t -> -9 (финальное состояние space), начинаем вновь с 0 строки, t -> 7

o -> 8

[ ] -> -8 (финальное состояние keywords), начинаем вновь с 0 строки, [ ]-> 9

N-> -9 (финальное состояние space), начинаем вновь с 0 строки, N-> 11

1-> 16

n-> 16 (в 16 строчке 0 колонка – EOF -> -10 финальное состояние id)

EOF -> -1 (финальное состояние EOF)