

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Институт космических и информационных технологий  
институт  
Программная инженерия  
кафедра

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №4**  
Синтаксический анализ контекстно-свободных языков  
тема

Преподаватель

подпись, дата

**Кузнецов А. С.**

инициалы, фамилия

Студент **КИ23-17/26, 032320511**  
номер группы, зачетной книжки

подпись, дата

**Степанов И. Д.**  
инициалы, фамилия

Красноярск 2025

## **СОДЕРЖАНИЕ**

1 Задачи .....	3
2 Ход работы.....	3
2.1 Описание 12 варианта задания .....	3
2.2 Построение КСГ .....	3
2.3 Распознавание тестовых цепочек методом Кока-Янгера-Касами .....	6
3 Заключение .....	13

## **1 Задачи**

1 Ознакомиться с теоретическими сведениями о нормальной форме Хомского и универсальных алгоритмах синтаксического анализа.

2 Получить у преподавателя собственный вариант задания с описанием контекстно-свободного языка, распознаваемого алгоритмом Кока-Янгера-Касами.

3 Используя изученные механизмы, определите в системе JFLAP согласно постановке задачи соответствующую КСГ. Определенный таким образом язык должен анализироваться алгоритмом Кока-Янгера-Касами. В случае невозможности создания КСГ это должно доказываться формально.

4 Написать отчет и представить его на проверку вместе с полученными JFLAP-моделями.

## **2 Ход работы**

### **2.1 Описание 12 варианта задания**

Необходимо с использованием системы JFLAP, построить грамматику, определяющую заданный язык для анализа его методом Кока-Янгера-Касами, или формально доказать невозможность этого.

Язык оператора присваивания, в правой части которого задано арифметическое выражение. Элементами выражений являются римские числа (при реализации достаточно знаков I, x, v, i), имена переменных из одного символа (от a до d), знаки операций и скобки для изменения порядка вычисления подвыражений. Операции (в сторону уменьшения приоритета): унарные, бинарные, присваивание.

### **2.2 Построение КСГ в JFLAP**

КСГ в НФХ представлена на рисунках 1-3.

LHS	
S	→ VP
P	→ GE
E	→ CF
R	→ RR
R	→ i
R	→ v
R	→ x
E	→ RR
F	→ RR
T	→ RR
E	→ b
E	→ l
E	→ i
E	→ v
E	→ VL
E	→ EO
O	→ BT
B	→ +
E	→ EN
N	→ CT
F	→ CF
T	→ CF
C	→ -

Рисунок 1 – Начало КСГ в НФХ

LHS	
C	$\rightarrow -$
F	$\rightarrow \text{a}$
F	$\rightarrow  $
T	$\rightarrow x$
F	$\rightarrow i$
T	$\rightarrow d$
F	$\rightarrow v$
F	$\rightarrow c$
T	$\rightarrow a$
T	$\rightarrow b$
T	$\rightarrow c$
T	$\rightarrow  $
T	$\rightarrow i$
E	$\rightarrow x$
F	$\rightarrow x$
F	$\rightarrow VL$
T	$\rightarrow VL$
T	$\rightarrow HK$
T	$\rightarrow v$
T	$\rightarrow TJ$
T	$\rightarrow TM$
E	$\rightarrow TM$
M	$\rightarrow DF$

Рисунок 2 – Продолжение КСГ в НФХ

LHS	
M	→ DF
D	→ /
V	→ VL
L	→ GV
G	→ =
V	→ a
V	→ b
V	→ c
V	→ d
R	→ I
F	→ HK
E	→ HK
K	→ EI
H	→ [
I	→ ]
E	→ d
F	→ d
E	→ a
E	→ TJ
J	→ AF
A	→ *
F	→ b
E	→ c

Рисунок 3 – Окончание КСГ в НФХ

### 2.3 Распознавание тестовых цепочек методом Кока-Янгера-Касами

Перехваты экранов распознавания тестовых цепочек методом Кока-Янгера-Касами представлены на рисунках 4-15.

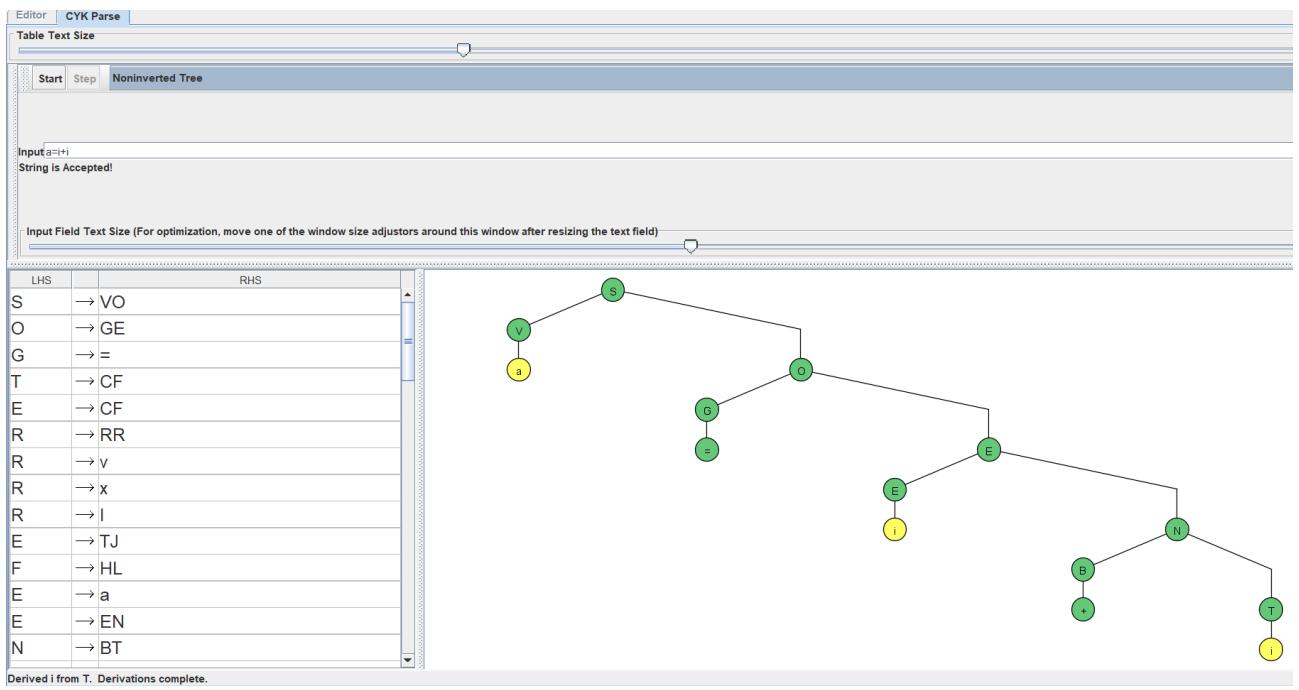


Рисунок 4 – Распознавание цепочки ‘ $a=i+i$ ’

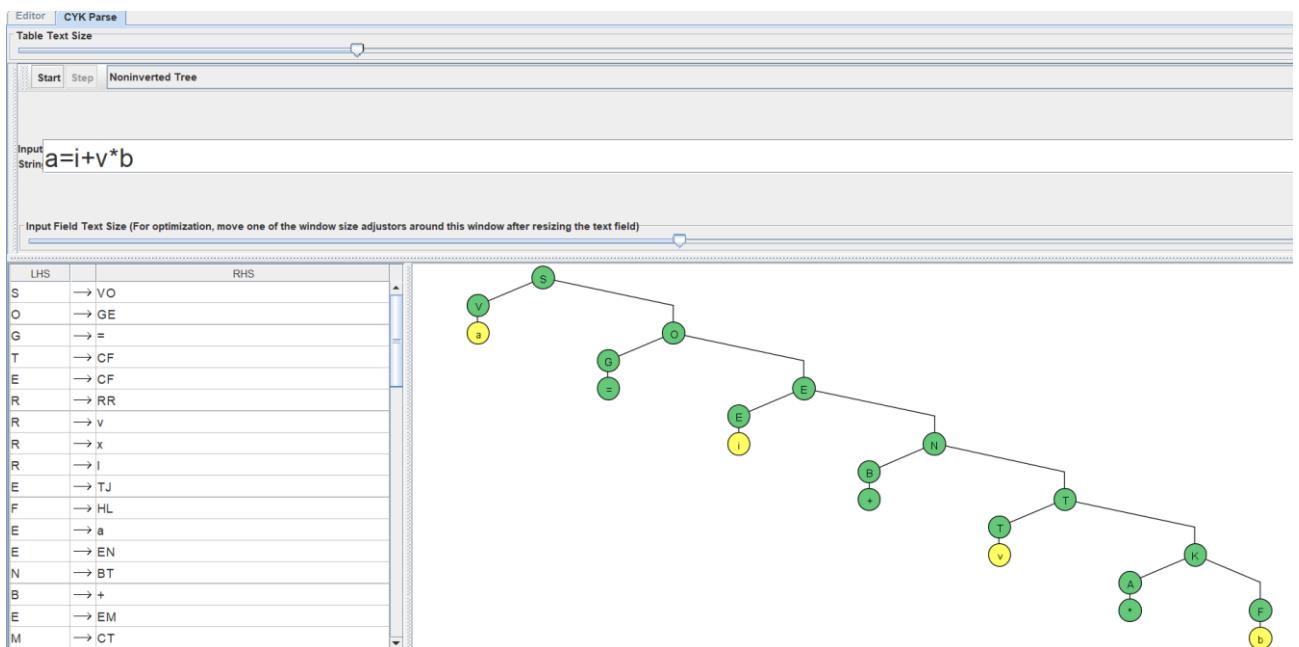


Рисунок 5 – Распознавание цепочки ‘ $a=i+v^*b$ ’

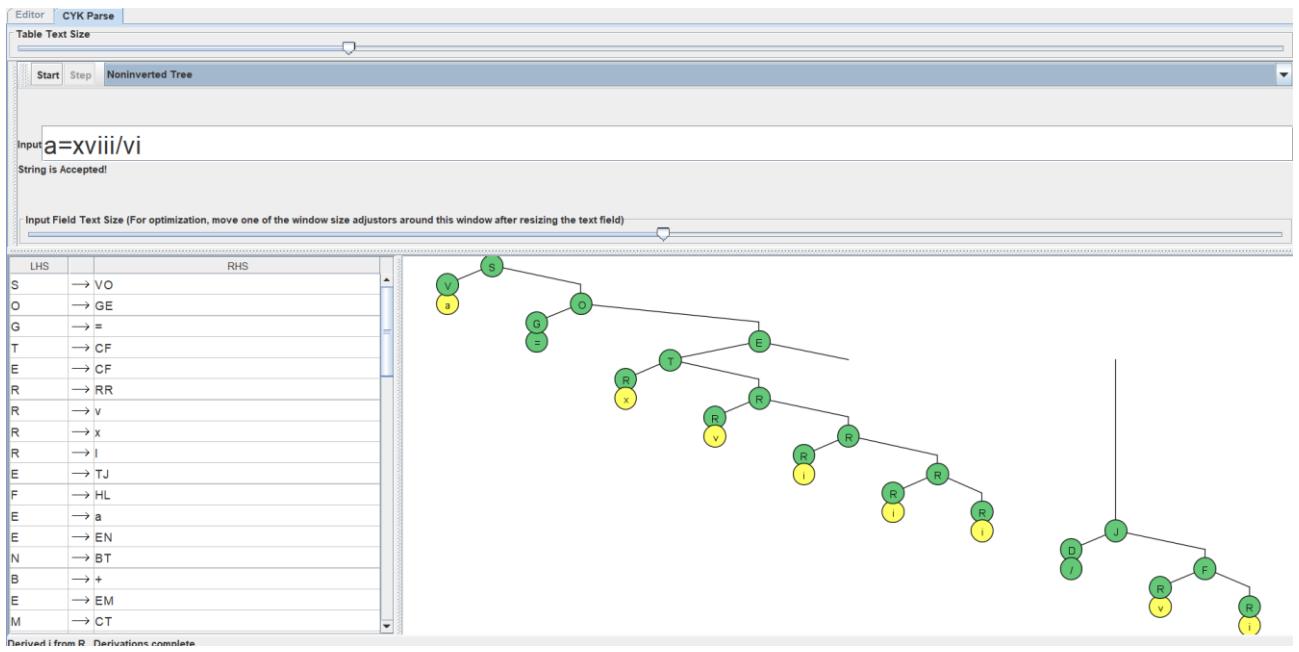


Рисунок 6 – Распознавание цепочки ‘ $a=xviii/vi$ ’

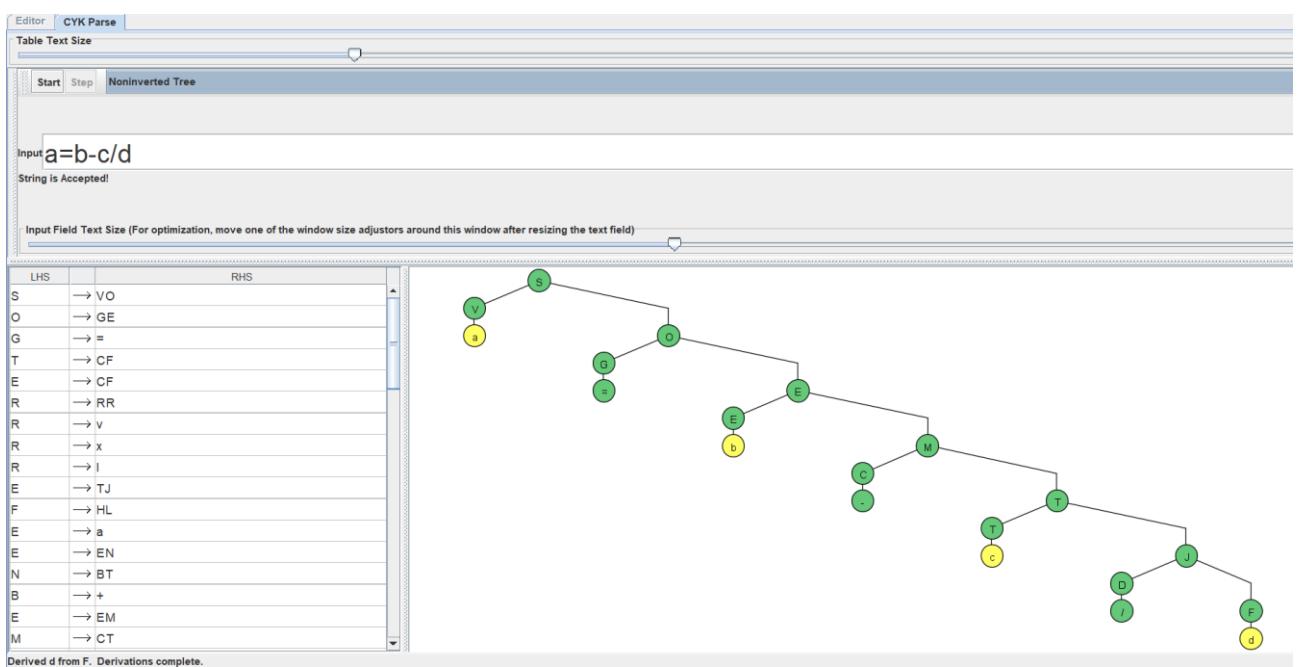


Рисунок 7 – Распознавание цепочки ‘ $a=b-c/d$ ’

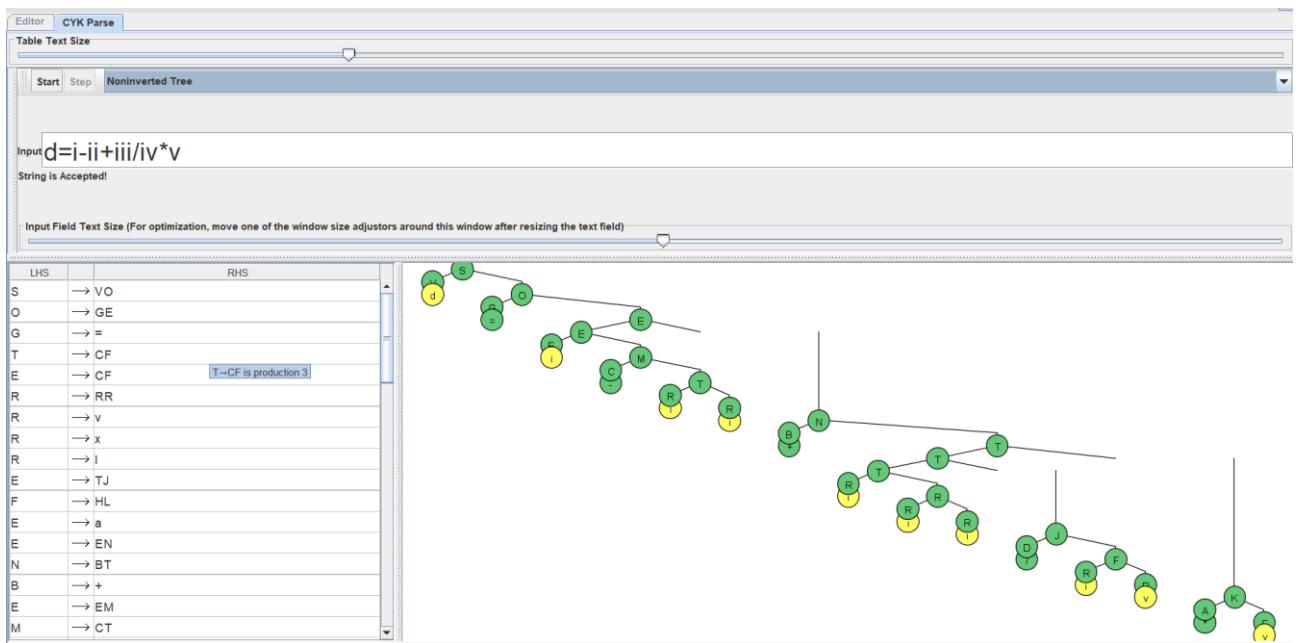


Рисунок 8 – Распознавание цепочки ‘ $d=i-ii+iii/iv^*v$ ’

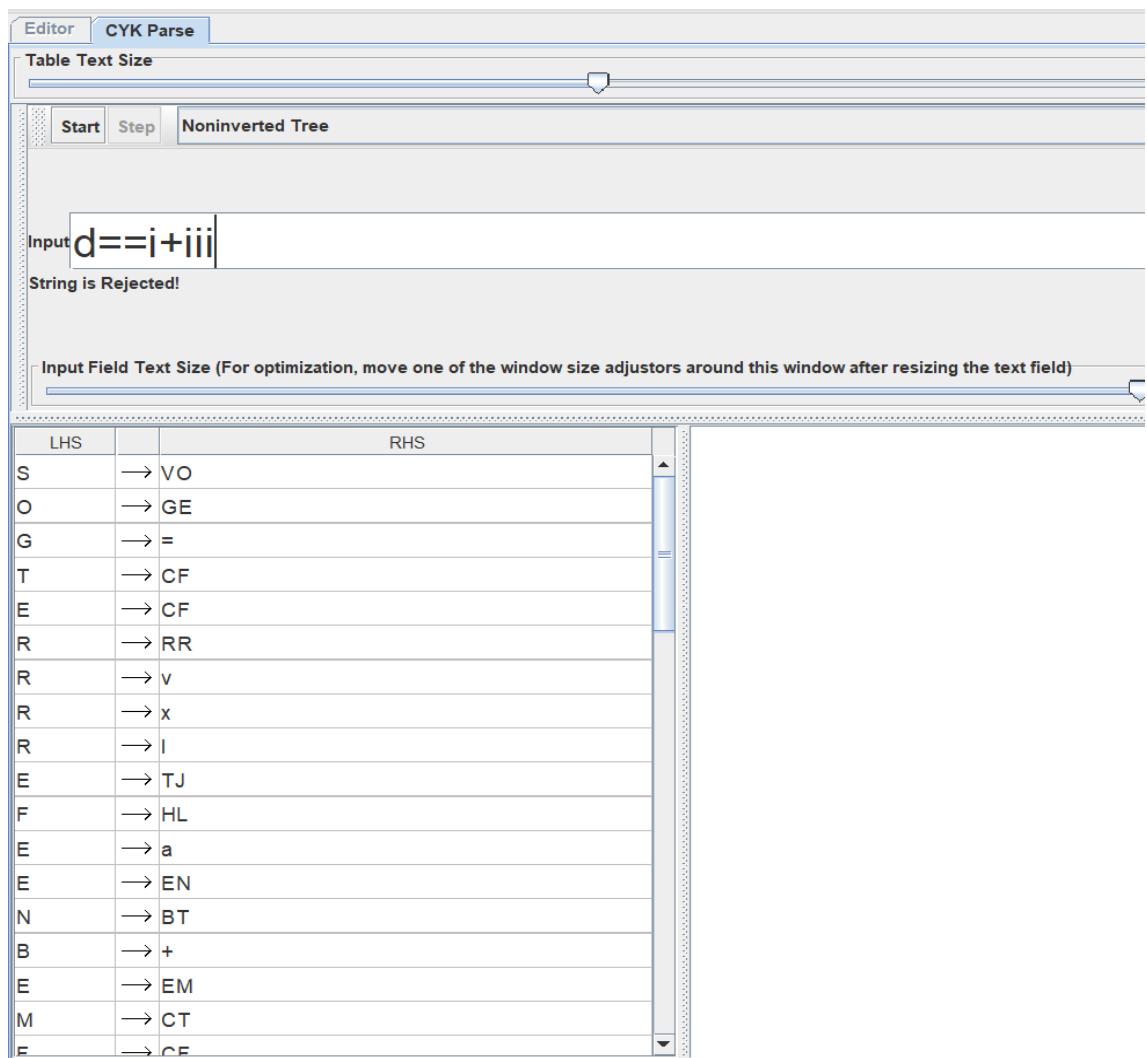


Рисунок 9 – Распознавание цепочки ‘ $d==i+iii$ ’

Editor CYK Parse

Table Text Size

Start Step Noninverted Tree

Input **d=i+t**

String is Rejected!

Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text field)

LHS	RHS
S	$\rightarrow VO$
O	$\rightarrow GE$
G	$\rightarrow =$
T	$\rightarrow CF$
E	$\rightarrow CF$
R	$\rightarrow RR$
R	$\rightarrow v$
R	$\rightarrow x$
R	$\rightarrow I$
E	$\rightarrow TJ$
F	$\rightarrow HL$
E	$\rightarrow a$
E	$\rightarrow EN$
N	$\rightarrow BT$
B	$\rightarrow +$
E	$\rightarrow EM$
M	$\rightarrow CT$
F	$\rightarrow CF$

Рисунок 10 – Распознавание цепочки ‘d=i+t’

Editor CYK Parse

Table Text Size

Start Step Noninverted Tree

Input  $d+a=a+b$

String is Rejected!

Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text field)

LHS	RHS
S	$\rightarrow VO$
O	$\rightarrow GE$
G	$\rightarrow =$
T	$\rightarrow CF$
E	$\rightarrow CF$
R	$\rightarrow RR$
R	$\rightarrow v$
R	$\rightarrow x$
R	$\rightarrow I$
E	$\rightarrow TJ$
F	$\rightarrow HL$
E	$\rightarrow a$
E	$\rightarrow EN$
N	$\rightarrow BT$
B	$\rightarrow +$
E	$\rightarrow EM$
M	$\rightarrow CT$
E	$\rightarrow CE$

Рисунок 11 – Распознавание цепочки ‘d+a=a+b’

The figure shows a screenshot of a CYK parser interface. At the top, there are tabs for "Editor" and "CYK Parse". Below the tabs, the "Table Text Size" is set to "Noninverted Tree". The main area has a toolbar with "Start" and "Step" buttons. The input string is "c=[a+iii]/iv-lxxxix\*d". A message below the input says "String Is Accepted!". Below the input field is a note: "Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text field)". To the left is a table showing the grammar rules (LHS | RHS). The right side displays a non-inverted parse tree where tokens are represented by green circles and non-terminals by yellow circles.

LHS	RHS
S	→ VO
O	→ GE
G	→ =
T	→ CF
E	→ CF
R	→ RR
R	→ v
R	→ x
R	→ i
E	→ TJ
F	→ HL
E	→ a
E	→ EN
N	→ BT
B	→ +
E	→ EM
M	→ CT

Derived d from F. Derivations complete.

Рисунок 12 – Распознавание цепочки ‘c=[a+iii]/iv-lxxxix\*d’

JFLAP : (CFG\_NFC.jff)

File Input Test Convert Help

Editor CYK Parse

Table Text Size

Start Step Noninverted Tree

Input  $c=[a+iii[iv-lxxxix^*d]$

String is Rejected!

Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text field)

LHS	RHS
S	$\rightarrow VO$
O	$\rightarrow GE$
G	$\rightarrow =$
T	$\rightarrow CF$
E	$\rightarrow CF$
R	$\rightarrow RR$
R	$\rightarrow v$
R	$\rightarrow x$
R	$\rightarrow I$
E	$\rightarrow TJ$
F	$\rightarrow HL$
E	$\rightarrow a$
E	$\rightarrow EN$
N	$\rightarrow BT$
B	$\rightarrow +$
E	$\rightarrow EM$
M	$\rightarrow CT$
F	$\rightarrow CF$

S → VO is production 0

Рисунок 13 – Распознавание цепочки ‘ $c=[a+iii[iv-lxxxix^*d$ ’

Editor CYK Parse

Table Text Size

Start Step Noninverted Tree

Input  $c=-i^*-a$

String is Accepted!

Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text field)

LHS	RHS
S	$\rightarrow VO$
O	$\rightarrow GE$
G	$\rightarrow =$
T	$\rightarrow CF$
E	$\rightarrow CF$
R	$\rightarrow RR$
R	$\rightarrow v$
R	$\rightarrow x$
R	$\rightarrow I$
E	$\rightarrow TJ$
F	$\rightarrow HL$
E	$\rightarrow a$
E	$\rightarrow EN$
N	$\rightarrow BT$
B	$\rightarrow +$
E	$\rightarrow EM$
M	$\rightarrow CT$

Derived a from F. Derivations complete.

Рисунок 14 – Распознавание цепочки ‘ $c=-i^*-a$ ’

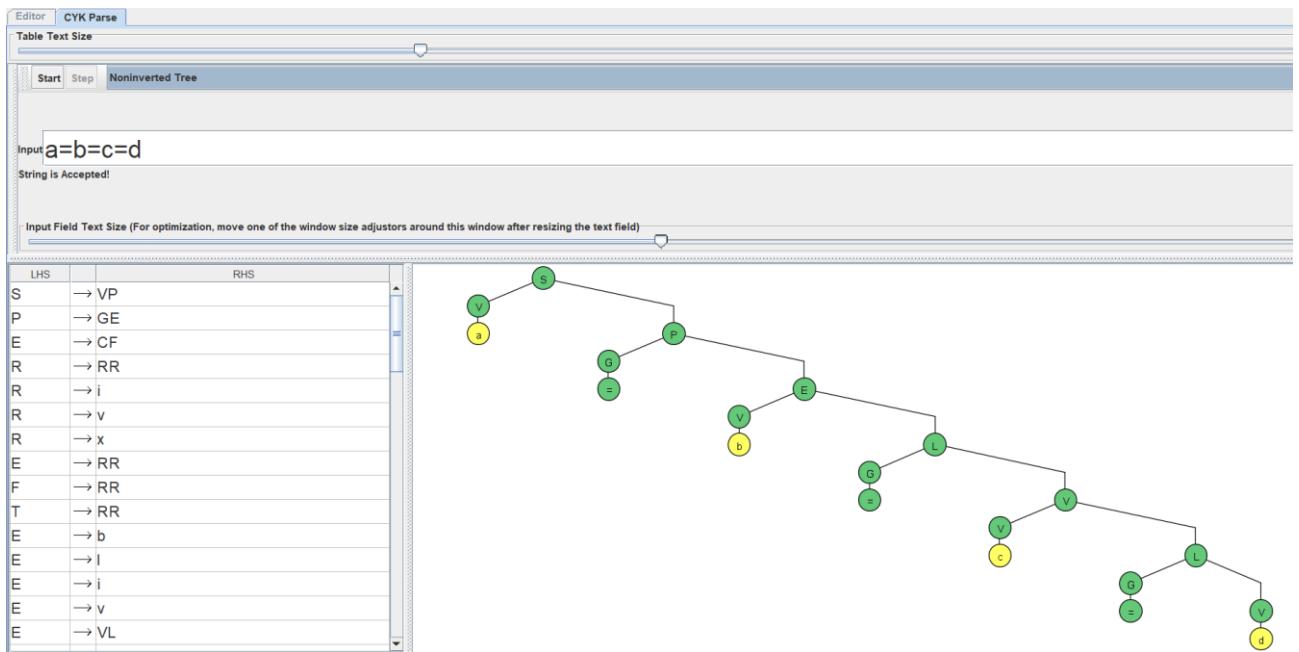


Рисунок 15 – Распознавание цепочки ‘a=b=c=d’

### 3 Заключение

В ходе работы были достигнуты все цели и выполнены все задачи.