Студент: Куликов А.В.
Группа: 208
Номер по списку: 9

Тема: Синтаксический анализ.

Лабораторнаые работы N6-7.

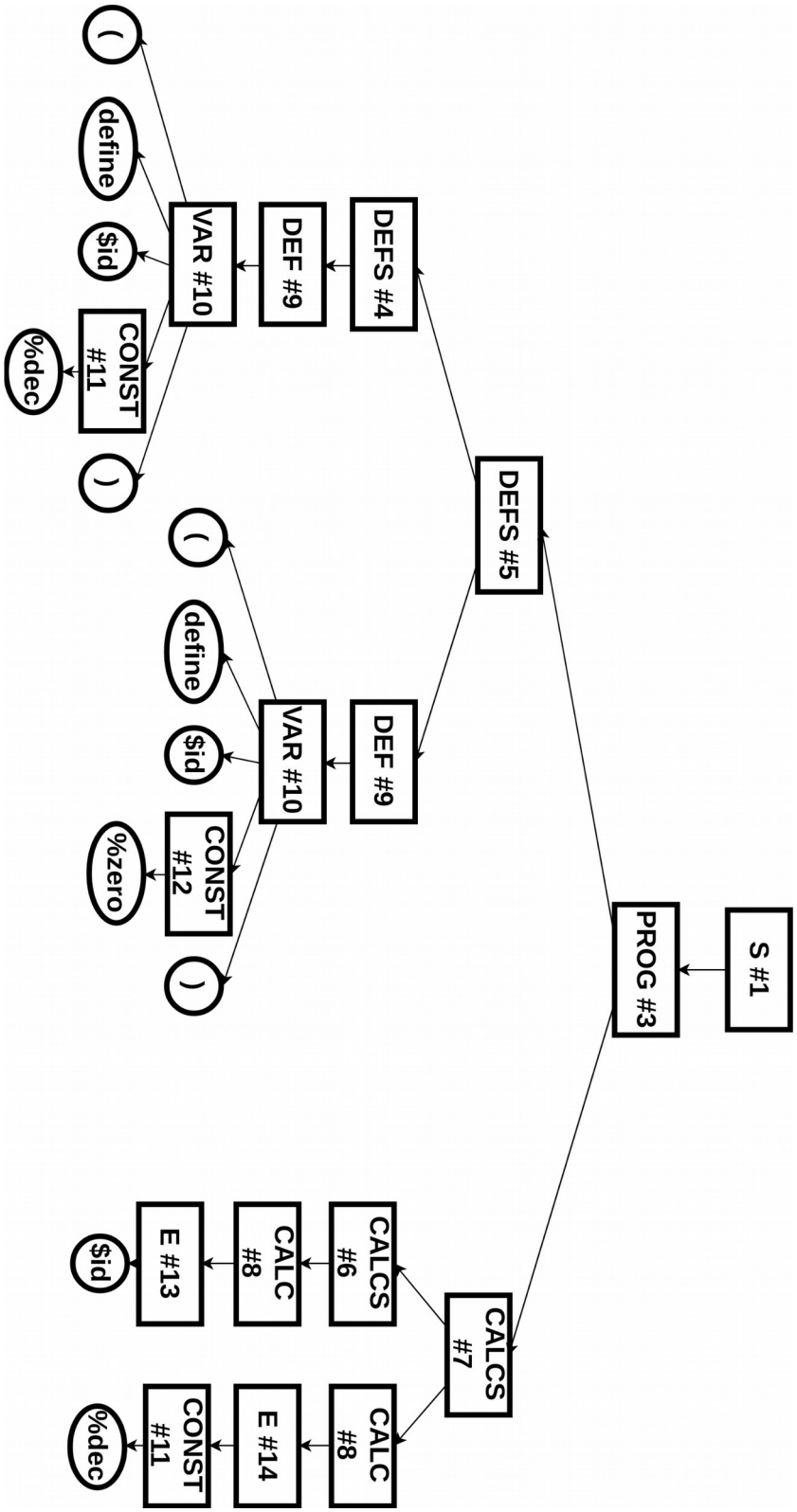
Вариант. 9.

v09=>
(define $id $dec) (define $id $zero) $id $dec

Распечатка грамматики.

```
# $v09
  $id   $dec  $zero  $bool
    (     )  define
#
S ->            PROG #1
PROG ->         DEFS CALCS #2
DEFS ->         DEF #3 |
                DEFS DEF #4
CALCS ->        CALC #5 |
                CALCS CALC #6
CALC ->         E #7
DEF ->          VAR #8
VAR ->          ( define $id CONST ) #9
CONST ->        $dec #10 |
                $zero #11
     E ->       $id #12 |
                CONST #13
```

Дерево разбора.



Скриншот запуска парсера для свого варианта задания с трассировкой.

```
Input gramma name>v09
Gramma:v09.txt
Source>(define probe 10) (define test 0) func 150
Source:temp.ss
   1|(define probe 10) (define test 0) func 150
   2|

_________________
KAV2019

______________________
  <-  (
  <-  define
  <-  $id
  <-  $dec
   CONST -> $dec #10
  <-  )
     VAR -> ( define $id CONST ) #9
     DEF -> VAR #8
    DEFS -> DEF #3
  <-  (
  <-  define
  <-  $id
  <-  $zero
   CONST -> $zero #11
  <-  )
     VAR -> ( define $id CONST ) #9
     DEF -> VAR #8
    DEFS -> DEFS DEF #4
  <-  $id
       E -> $id #12
    CALC -> E #7
   CALCS -> CALC #5
  <-  $dec
   CONST -> $dec #10
       E -> CONST #13
    CALC -> E #7
   CALCS -> CALCS CALC #6
    PROG -> DEFS CALCS #2
       S -> PROG #1
Good sourse!

________________________
Source>
```

Скриншот запуска парсера для свого файла coin.ss из лабораторной №3 в грамматике mlisp19 без трассировки.

```
Input gramma name>mlisp19
Gramma:mlisp19.txt
Source>coin19
Source:coin19.ss
   1|(define VARIANT 9)
   2|(define LAST-DIGIT-OF-GROUP-NUMBER 8)
   3|(define LARGEST-COIN 20)
   4|
   5|(define (implication? x? y?) (not (and x? (not y?))))
   6|
   7|(define (cc amount largest-coin) (cond ((or (= amount 0) (= largest-coin 1)) 1)
   8|                                  ((implication? (>= amount 0) (= largest-coin 0)) 0)
   9|                                  (else (+ (cc amount (next-coin largest-coin))
  10|                                     (cc (- amount largest-coin) largest-coin))) )
  11|)
  12|
  13|(define (count-change amount) (cc amount LARGEST-COIN))
  14|
  15|(define (next-coin coin) (cond ((= coin 20) 15)
  16|                         ((= coin 15) 10)
  17|                         ((= coin 10) 5)
  18|                         ((= coin 5) 3)
  19|                         ((= coin 3) 2)
  20|                         (else 1) )
  21|)
  22|
  23|(define (GR-AMOUNT) (remainder (+ (* 100 LAST-DIGIT-OF-GROUP-NUMBER) VARIANT) 137))
  24|
  25|(display " KAV variant ")
  26|(display VARIANT) (newline)
  27|(display " 1-2-3-5-10-15-20") (newline)
  28|(display "count__change for 100 \t= ")
  29|(display (count-change 100)) (newline)
  30|(display "count__change for ");
  31|(display (GR-AMOUNT))
  32|(display " \t= ")
  33|(display (count-change (GR-AMOUNT)))(newline)
  34|
_________________
Good sourse!

_________________________
Source>
```

Скриншоты запуска парсера для файлов id.ss и idq.ss из лабораторной №5 в грамматике mlisp19 **без** трассировки.

```
Gramma:mlisp19.txt
Source>id
Source:id.ss
   1|; id.ss
   2|"$id"
   3| good
   4| eE
   5| count-change
   6| !
   7| A!
   8| abc123
   9|
  10| a---b ;01
  11| a--!b ;02
  12| a--?b ;03
  13| a-!-b ;04
  14| a-!!b ;05
  15| a-!?b ;06
  16| a-?-b ;07
  17| a-?!b ;08
  18| a-??b ;09
  19| a!--b ;10
  20| a!-!b ;11
  21| a!-?b ;12
  22| a!!-b ;13
  23| a!!!b ;14
  24| a!!?b ;15
  25| a!?-b ;16
  26| a!?!b ;17
  27| a!??b ;18
  28| a?--b ;19
  29| a?-!b ;20
  30| a?-?b ;21
  31| a?!-b ;22
  32| a?!!b ;23
  33| a?!?b ;24
  34| a??-b ;25
  35| a??!b ;26
  36| a???b ;27
  37|
  38| 123abc    ;?
  39| -c     ;?
  40|-!      ;?
  41|-  ;-
  42|
-----------------
Syntax: unknown token!
  18| a-??b ;09
      ^


-------------------------
Source>
```

```
Input gramma name>mlisp19
Gramma:mlisp19.txt
Source>idq
Source:idq.ss
   1|; idq.ss
   2|"$idq"
   3| ?
   4| What?
   5| good-enough?
   6| DF--G?
   7| -a?
   8| -1?
   9|
  10| ???
  11| ?!?
  12| ?-?
  13| !??
  14| !!?
  15| !-?
  16| -??
  17| -!?
  18| --?
  19|
  20|
  21| 123abc?   ;?
  22| ?b             ;?
  23|
-----------------
Syntax: unknown token!
  11| ?!?
      ^


-------------------------
Source>
```

Выводы .

В ходе выполнения данных лабораторных работ я познакомился с процессом синтаксического разбора выражений использованием КС-грамматик.

Синтаксический разбор – следующий этап после лексического разбора, не менее важная часть компилятора.

С несколько похожей задачей я уже сталкивался ранее, при работе с арифметическими выражениями. Здесь же задача более общая и, следовательно, более сложная.

Самостоятельной работы в данном задании было немного. Ее суть, по видимому, заключалась в именно понимании. И какое-то понимание, процесса синтаксического разбора, определенно, достигнуто.

Задания лабораторных работ выполнены в полном объеме.