Лабораторная работа №3

«Построение синтаксического анализатора»

**Задание:**

Разработать программу реализующую синтаксический анализатор для следующего языка

**Оператор присваивания:**

<ид> := <ар. выр.>;

**Условный оператор:**

if (<лог. выр.>)<оператор> [else <оператор> ]

**Оператор цикла:**

ЦИКЛ <совокупность операторов> ПОКА\_НЕ <лог.выр>;

**Арифметическое выражение:**

<E> ::= <T> <E-список>

<E-список> ::= -<T> <E-список>

<E-список>::= !

<T> ::= <F> <T-список>

<T-список> ::= / <F><T-список>

<T-список> ::= !

<F> ::= <Id>

<F> ::= <Int>

**Логическое выражение**

<лог. выр.> := <F><лог. опер.><F>

< лог. опер.> := >

< лог. опер.> := <

**Пример программы:**

d:= 23 - 1;

a:= 3 - 254 / d;

c:= 200 / d -367 / a;

if (d > c) a = d - 2; else a:= d / 3;

h = 10;

ЦИКЛ

h:=h-1;

ПОКА\_НЕ h<0;

Возможны следующие виды лексем:

- идентификаторы

- ключевые слова

- целые числа

- разделители

**LL(1)-грамматика:**

1. <S> -> <совокупность операторов>
2. <совокупность операторов>-> <оператор><совокупность операторов>
3. <совокупность операторов>-> ε
4. <Оператор> ->Id:=<E>;
5. <Оператор> -> if **(**<лог.выр>**)** <Оператор> <if -список>
6. <Оператор> -> ЦИКЛ <совокупность операторов> ПОКА\_НЕ <лог.выр>
7. <if-список> -> else <Оператор>
8. <if-список> -> ε
9. <Лог. выражение> -> <F>< Лог. оператор ><F>
10. <Лог. оператор> -> **>**
11. <Лог. оператор> -> <
12. <T> -> <F> <T-список>
13. <E-список> -> **-** <T> <E-список>
14. <E-список> -> ε
15. <E>-> <T><E-список>
16. <T-список>-> **/** <F><T-список>
17. <T-список> -> ε
18. <F> -> Id
19. <F> -> Int

**Множество выбора для LL(1)-грамматики:**

Выбор(1) = {Id, if, ЦИКЛ, -|}

Выбор(2) = {Id, if, ЦИКЛ }

Выбор(3) = {-|}

Выбор(4) = {Id}

Выбор(5) = {if}

Выбор(6) = {ЦИКЛ}

Выбор(7) = {else}

Выбор(8) = {if, ЦИКЛ, Id, -| }

Выбор(9) = {Id, Int}

Выбор(10) = {>}

Выбор(11) = {<}

Выбор(12) = {Int, Id}

Выбор(13) = {-}

Выбор(14) = {;}

Выбор(15) = {Int, Id}

Выбор(16) = {/}

Выбор(17) = {-, ;}

Выбор(18) = {Id}

Выбор(19) = {Int}

**Управляющая таблица МП-автомата:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *Id(1)* | *Int(2)* | /(10) | -(11) | :=(12) | >(13) | < (14) | ( (15) | ) (16) | ; (17) | If(101) | else(102) | ЦИКЛ(103) | ПОКА\_НЕ(104) | -|(99) |
| <S> 24 | #1 | - | - | - | - | - | - | - | - | - | #1 | - | #1 | - | #1 |
| <Сов.оп.> 29 | #2 | - | - | - | - | - | - | - | - | - | #2 | - | #2 | #3 | #3 |
| <Оператор> 26 | #4 | - | - | - | - | - | - | - | - | - | #5 | - | #6 | - | - |
| <E.> 30 | #12 | #12 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| <Лог.выр> 27 | #8 | #8 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| <if-список> 25 | #3 | - | - | - | - | - | - | - | - | - | #3 | #7 | #3 | - | #3 |
| <Лог. опер> 28 | - | - | - | - | - | #9 | #9 | - | - | - | - | - | - | - | - |
| <F> 34 | #9 | #9 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| <T.> 32 | #10 | #10 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| <T-список> 33 | - | - | #13 | #3 | - | - | - | - | - | #3 | - | - | - | - | - |
| <E-список> 31 | - | - | - | #11 | - | - | - | - | - | #3 | - | - | - | - | - |
| ПОКА\_НЕ 104 | - | - | - | - | - | - | - | - | - | - | - | - | - | #9 | - |
| ( 15 | - | - | - | - | - | - | - | #9 | - | - | - | - | - | - | - |
| ) 16 | - | - | - | - | - | - | - | - | #9 | - | - | - | - | - | - |
| Id 1 | #9 | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| := 12 | - | - | - | - | #9 | - | - | - | - | - | - | - | - | - | - |
| ; 17 | - | - | - | - | - | - | - | - | - | #9 | - | - | - | - | - |
| ∇ 0 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | Доп |

<S>∇

#1 –Заменить(<Совокупность операторов>), Держать

#2 – Заменить(<Оператор>< Совокупность операторов >), Держать

#3 – Вытолкнуть, Держать

#4 – Заменить(=<E>;), Сдвиг

#5 – Заменить((<лог.выр>) <Оператор><if-список>), Сдвиг

#6 – Заменить(<Совокупность операторов> ПОКА\_НЕ <лог.выражение>), Сдвиг

#7 – Заменить(<Оператор>), Сдвиг

#8 – Заменить(<F><лог.опер.><F>), Держать

#9 – Вытолкнуть, Сдвиг

#10 –Заменить(<F><T-список>), Держать

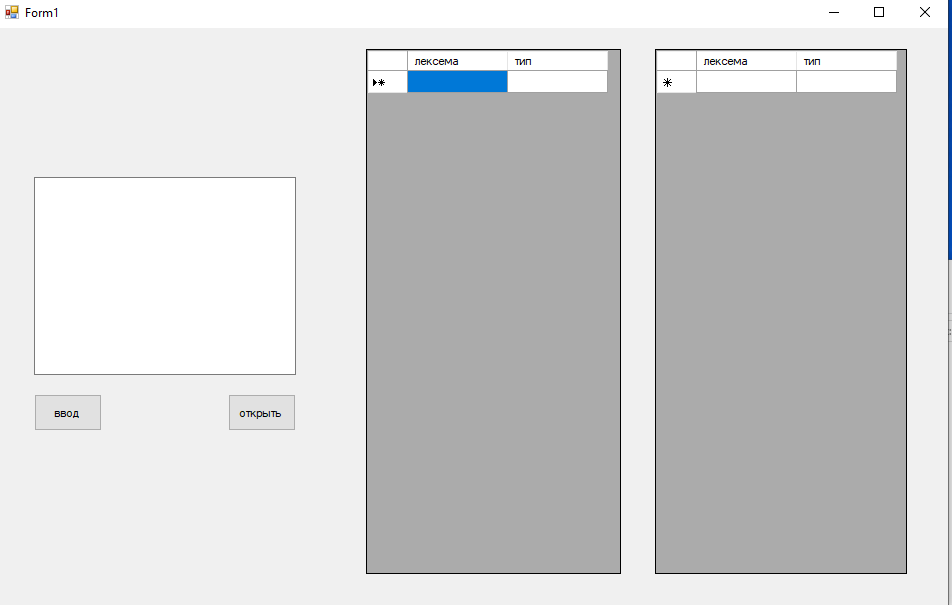
#11 – Заменить(<T><E-список>), Сдвиг

#12 – Заменить(<T><E-список>), Держать

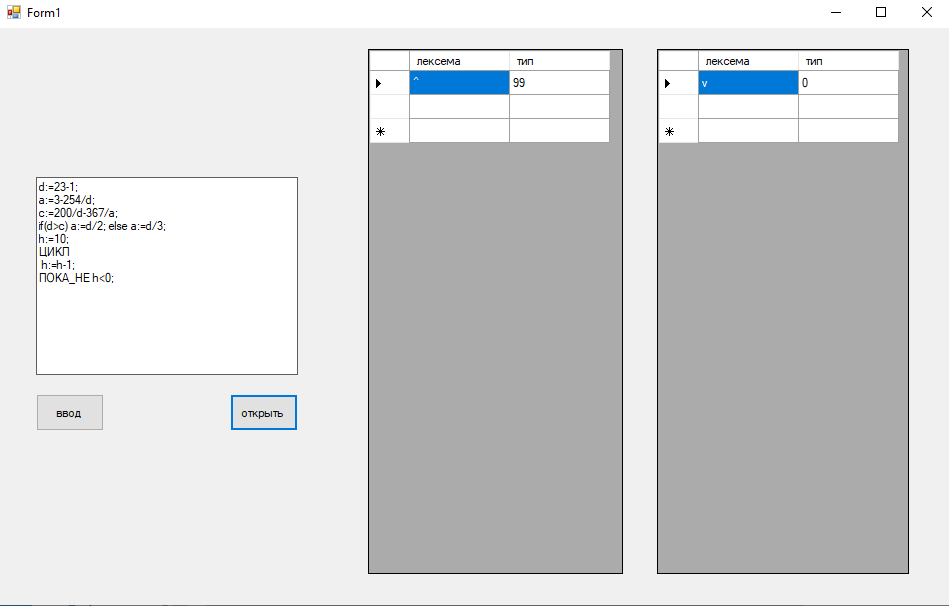
#13 – Заменить(<F><T-список>), Сдвиг

**Скриншоты работы программы:**

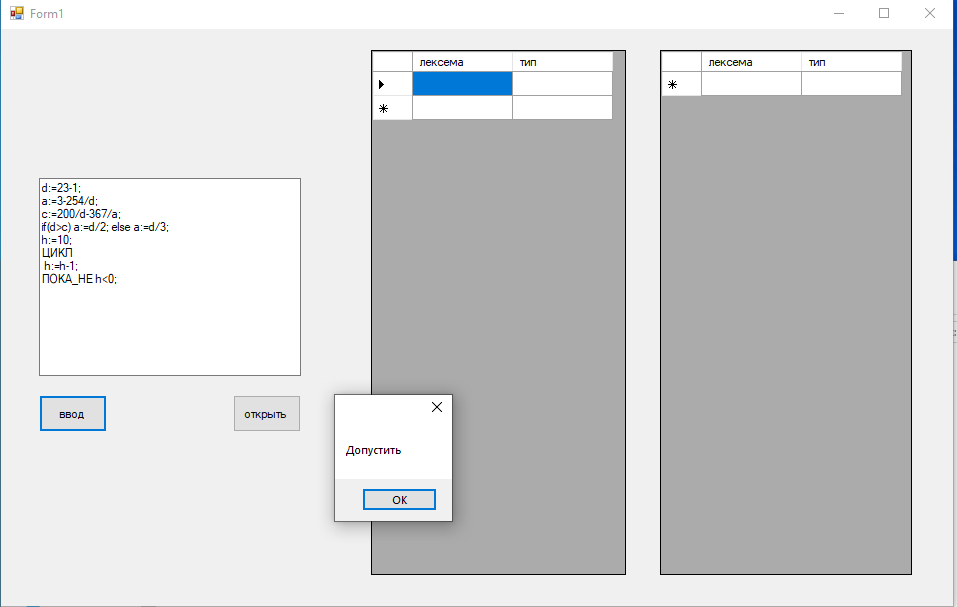
Запуск программы



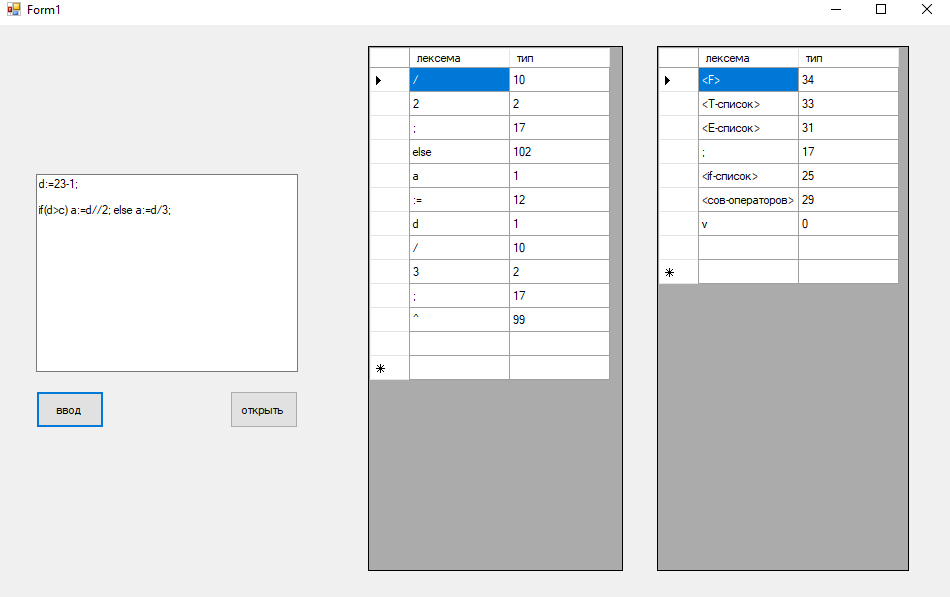
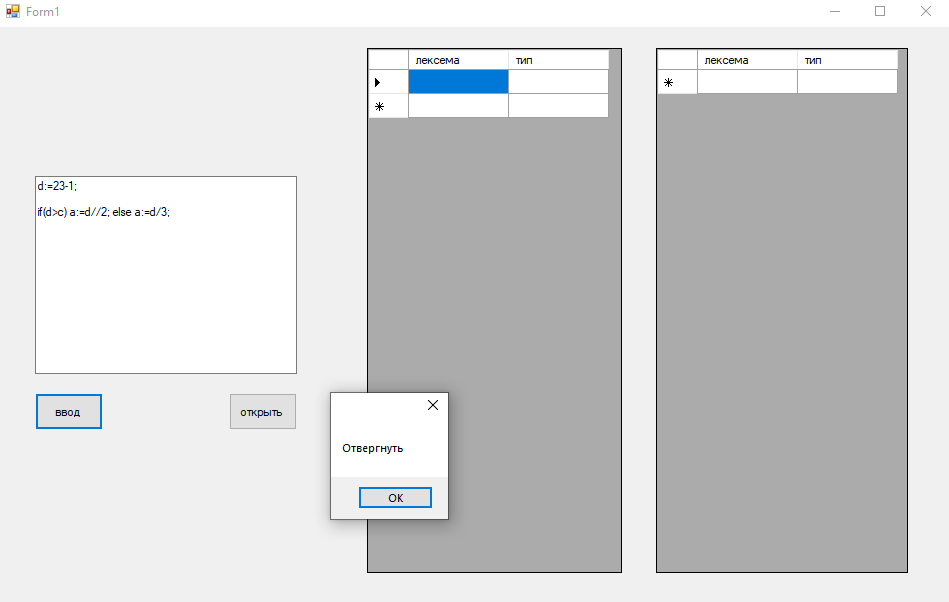
Ввод из файла



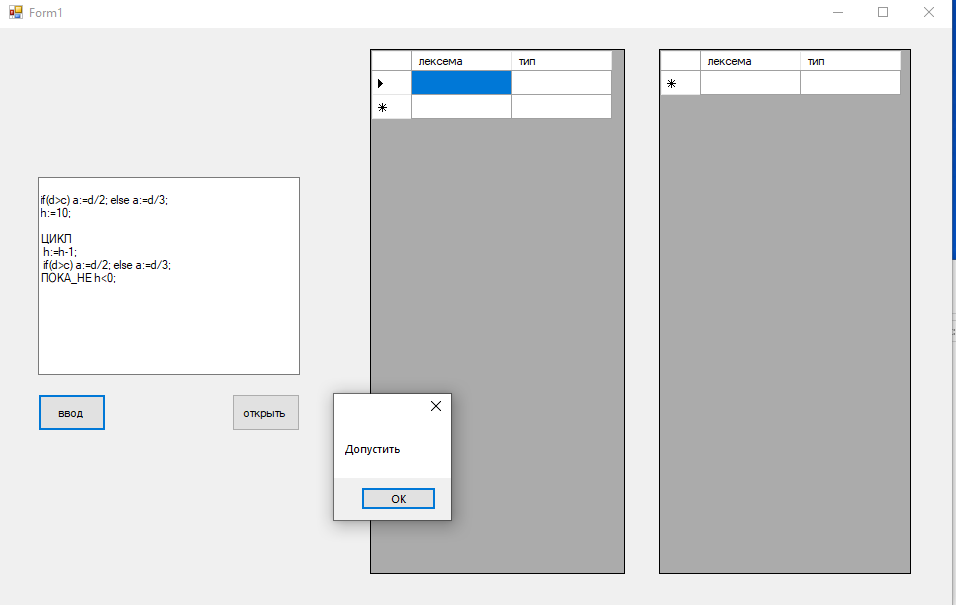
Тест 1



Тест 2(неверный)



Тест 3



Тест 4

