|  |  |  |
| --- | --- | --- |
| Лабораторная работа №1  «Теория автоматов и формальных языков»  Лексический анализатор. | ФИО | Маркосян К.Г.  Окороджи Д.Ф |
| Группа | ПРИН – 367 |
| Преподаватель | Ерофеев А.А. |
| Срок сдачи |  |
| Оценка |  |

Задание.

Написать с помощью генератора лексических анализаторов flex программу, осуществляющую лексический разбор языка Кумир с выводом на экран перечня лексем (с их семантическими значениями).

Список поддерживаемых лексем

1.

алг – главный алгоритм

алг [тип\_возвращаемого\_значения] <имя\_алгоритма> [(<список аргументов>)] – вспомогательный алгоритм

дано – условие применимости алгоритма

надо – цель выполнения алгоритма

нач – начало тела алгоритм

кон – конец тела алгоритм

арг – передаваемые аргументы

рез – результирующие аргументы

аргрез – обновляемые аргументы

; - перенос строки

знач – возвращаемое значение функции

Типы величин:

* цел
* вещ
* лог
* сим
* лит
* таб
* целтаб
* вещтаб
* логтаб
* симтаб
* литтаб

и – логическое и

или – логическое или

не – логическое не

да – истина

нет – ложь

утв – проверка условия внутри алгоритма

выход – вынужденное завершение цикла или алгоритма

ввод – ввод данных

вывод – вывод данных

нс – перевод строки \n

если то иначе все (всё) – синтаксис ветвлений

выбор при иначе все (всё) – синтаксис множественного ветвления

Синтаксис циклов:

* нц
* кц
* кц\_при
* раз
* пока
* для
* от
* до
* шаг

2.

Имя – набор слов, разделённых пробелами. Первое слово не может начинаться с цифры. Ни одно слово не может быть ключевым.

Словарными символами являются

∙ буквы (кириллические и латинские, прописные и строчные)

∙ цифры

∙ два специальных знака: @ \_

3.

**Целочисленные константы:**

|цел| < 231 = 2147483647

Целочисленные шестнадцатеричные константы предваряются символом $

Примеры: 123, -100000, $100.

**Вещественные константы:**

|вещ| < 21023

Вещественные константы можно записывать в десятичной и экспоненциальной форме. В качестве разделителя в экспоненциальной записи можно использовать любой вариант буквы e: строчный или прописной, латинский или кириллический.

Примеры: 1.23, -0.56, 1e+4, 5E-7.

**Логические константы:**

да, нет

**Символьные и литеральные константы:**

‘ ’ или “ ”, внутри ‘ ’ не может быть ‘, внутри “ “ не может быть “.

**Задание таблиц:**

тип: цел таб, целтаб, вещь таб, вещтаб и т.д.

границы задаются после имени в квадратных скобках через двоеточие

Примеры:

цел таб а[-5:5]

вещтаб tab[1:4, 1:12]

4.

Присвоение

а := б

Арифметические операции:

|  |  |
| --- | --- |
| сложение | x + y |
| вычитание | x - y |
| умножение | x \* y |
| деление | x / y |
| возведение в степень | x \*\* y |

Операции сравнения:

|  |  |
| --- | --- |
| равно | x = y |
| не равно | x <> y |
| меньше | x<y |
| больше | x>y |
| меньше или равно | x<=y |
| Больше или равно | x>=y |

Логические операции

|  |  |
| --- | --- |
| Конъюнкция | А и Б |
| Дизъюнкция | А или Б |
| Отрицание | не А |

Операции для работы со строками

|  |  |
| --- | --- |
| Слияние | а+б |
| Вырезка | а[3:5] |
| взятие символа | а[3] |
| Равно | а = б |
| не равно | а <> б |

Стандартные функции для работы с числами

|  |  |
| --- | --- |
| корень квадратный | sqrt(x) |
| абсолютная величина | abs(x) и iabs(x) |
| знак числа (-1, 0 или 1) | sign(x) |
| синус | sin(x) |
| косинус | cos(x) |
| тангенс | tg(x) |
| котангенс | ctg(x) |
| арксинус | arcsin(x) |
| арккосинус | arccos(x) |
| арктангенс | arctg(x) |
| арккотангенс | arcctg(x) |
| натуральный логарифм | ln(x) |
| десятичный логарифм | lg(x) |
| степень числа 𝑒 (𝑒 ≈ 2.718181) | exp(x) |
| минимум из чисел x и y | min(x,y) |
| максимум из чисел x и y | max(x,y) |
| остаток от деления x на y (x, y — целые) | mod(x,y) |
| частное от деления x на y (x, y — целые) | div(x,y) |
| целая часть числа x | int(x) |
| случайное число в диапазоне от 0 до x | rnd(x) |

Другие встроенные алгоритмы

|  |  |
| --- | --- |
| Строковое представление целого числа | цел\_в\_лит(х) |
| Строковое представление вещественного числа | вещ\_в\_лит(х) |
| Перевод строки в целое число | лит\_в\_цел(стр, успех) |
| Перевод строки в вещественное число | лит\_в\_вещ(стр, успех) |
| Длина строки | длин(стр) |
| Код символа в таблице КОИ-8 | код(с) |
| Символ таблицы КОИ-8 | символ(х) |
| Код символа в таблице Юникод | юникод(с) |
| Символ таблицы Юникод | символ2(х) |

5.

| - однострочный комментарий

# - информация об алгоритме, располагается между **алг** и **нач**

Входной файл для FLEX

%option noyywrap

%option never-interactive

digit [0-9]

number {digit}+

hexnum "$"[0-9a-fA-F]+

float {number}"."{digit}\*

identifier [a-zA-Zа-яА-ЯёЁ@\_][a-zA-Zа-яА-ЯёЁ0-9@\_]\*

identifier2 [a-zA-Zа-яА-ЯёЁ0-9@\_]+

quote1 "\'"[^']\*"\'"

quote2 "\""[^"]\*"\""

%%

алг printf("Found Keyword: %s\n", yytext);

дано printf("Found Keyword: %s\n", yytext);

надо printf("Found Keyword: %s\n", yytext);

нач printf("Found Keyword: %s\n", yytext);

кон printf("Found Keyword: %s\n", yytext);

арг printf("Found Keyword: %s\n", yytext);

рез printf("Found Keyword: %s\n", yytext);

аргрез printf("Found Keyword: %s\n", yytext);

знач printf("Found Keyword: %s\n", yytext);

цел printf("Found Keyword: %s\n", yytext);

вещ printf("Found Keyword: %s\n", yytext);

лог printf("Found Keyword: %s\n", yytext);

сим printf("Found Keyword: %s\n", yytext);

лит printf("Found Keyword: %s\n", yytext);

таб printf("Found Keyword: %s\n", yytext);

целтаб printf("Found Keyword: %s\n", yytext);

вещтаб printf("Found Keyword: %s\n", yytext);

логтаб printf("Found Keyword: %s\n", yytext);

и printf("Found Keyword: %s\n", yytext);

или printf("Found Keyword: %s\n", yytext);

не printf("Found Keyword: %s\n", yytext);

да printf("Found Keyword: %s\n", yytext);

нет printf("Found Keyword: %s\n", yytext);

выход printf("Found Keyword: %s\n", yytext);

вывод printf("Found Keyword: %s\n", yytext);

ввод printf("Found Keyword: %s\n", yytext);

если printf("Found Keyword: %s\n", yytext);

то printf("Found Keyword: %s\n", yytext);

иначе printf("Found Keyword: %s\n", yytext);

всё printf("Found Keyword: %s\n", yytext);

выбор printf("Found Keyword: %s\n", yytext);

при printf("Found Keyword: %s\n", yytext);

нц printf("Found Keyword: %s\n", yytext);

кц printf("Found Keyword: %s\n", yytext);

кц\_при printf("Found Keyword: %s\n", yytext);

раз printf("Found Keyword: %s\n", yytext);

пока printf("Found Keyword: %s\n", yytext);

для printf("Found Keyword: %s\n", yytext);

от printf("Found Keyword: %s\n", yytext);

до printf("Found Keyword: %s\n", yytext);

шаг printf("Found Keyword: %s\n", yytext);

нс printf("Found Keyword: %s\n", yytext);

{number} printf("Found Number in decimal form: %s\n", yytext);

{hexnum} {

int d;

printf("Found number in hexadecimal form: %s", yytext);

sscanf(yytext+1, "%x", &d);

printf("(%d)\n", d);

}

{float} printf("Found number in float form: %s\n", yytext);

{identifier} printf("Found identifier: %s\n", yytext);

{quote1} printf("Found string: %s\n", yytext);

{quote2} printf("Found string with two dots: %s\n", yytext);

"," printf("Found comma\n");

"(" printf("Found opening brackets : %s\n", yytext);

")" printf("Closing brackets: %s\n", yytext);

"[" printf("Opening Braces: %s\n", yytext);

"]" printf("Closing braces: %s\n", yytext);

":" printf("Semi Colon\n");

":"= printf("Operation found: %s\n", yytext);

"+" printf("Operation found: %s\n", yytext);

"-" printf("Operation found: %s\n", yytext);

"\*" printf("Operation found \*: %s\n", yytext);

"/" printf("Operation found /: %s\n", yytext);

"\*\*" printf("Operation found \*\*: %s\n", yytext);

"=" printf("Operation found =: %s\n", yytext);

"<>" printf("Operation found <>: %s\n", yytext);

"<" printf("Operation found <: %s\n", yytext);

">" printf("Operation found >: %s\n", yytext);

"<=" printf("Operation found <=: %s\n", yytext);

">=" printf("Operation found >=: %s\n", yytext);

";" printf("Newline found: %s\n", yytext);

" "|"\t" printf("\n");

"|".\* printf("Commentary has been inputed: %s\n", yytext);

"#".\* printf("Commentary inputed between type: %s\n", yytext);

" "|"\n"|"\n\r"|"\r\n"|"\r"; /\*skip\*/

%%

Тестовый пример на кумир

|главная функция

алг

#тестовый пример

нач

цел целое

вещ float

лог true 5ili, true 5 ili

лит s1, s2

сим сим\_в@л

сим\_в@л := 'a'

сим\_в@л:="b"

s2:='edew@\_"'

s2 := "rkofr;',"

целое:=вернуть\_вв; float:= 23.3

типы\_и\_циклы('строка', s1, целое) | типы и циклы

| условия и операции

кон

алг цел вернуть

надо | вернуть

нач

знач := 3

кон

алг типы\_и\_циклы (арг лит первый , рез лит второй , аргрез цел третий)

дано первый = "строка"

нач

симтаб таблица[1:100, 2:200]

целтаб просто так [-5:0]

цел таб то\_же самое [-4:1, 50:55, 7:34]

литтаб текст[0:10]

вещтаб вещтаблица[0:1]

логтаб тф [50:60]

тф [53] := да

тф [54] := нет

тф [55] := тф[53] и тф[54] или тф[53] и (не тф[54] или тф[53])

ввод текст[0]

вывод текст[0]

ввод текст[1], текст[2]

вывод текст[1], нс, текст [2]

алг условия

нач

цел а, б, в

вещ г, д, е, ж

а := 123; б:= -10000; в :=$FF

г := 1.23; д:= -0.56; е :=1е+4; ж := 5E-7

выбор

при а = 2: а := 3

при а = 3: а := 4

при а = 4: а := 5; б := 4

иначе а := 6

всe

кон

Результат

Commentary has been inputed: |главная функция

Found Keyword: алг

Commentary inputed between type: #тестовый пример

Found Keyword: нач

Found Keyword: цел

Found identifier: целое

Found Keyword: вещ

Found identifier: float

Found Keyword: лог

Found identifier: true

Found Number in decimal form: 5

Found identifier: ili

Found comma

Found identifier: true

Found Number in decimal form: 5

Found identifier: ili

Found Keyword: лит

Found identifier: s1

Found comma

Found identifier: s2

Found Keyword: сим

Found identifier: сим\_в@л

Found identifier: сим\_в@л

Operation found: :=

Found string: 'a'

Found identifier: сим\_в@л

Operation found: :=

Found string with two dots: "b"

Found identifier: s2

Operation found: :=

Found string: 'edew@\_"'

Found identifier: s2

Operation found: :=

Found string with two dots: "rkofr;',"

Found identifier: целое

Operation found: :=

Found identifier: вернуть\_вв

Newline found: ;

Found identifier: float

Operation found: :=

Found number in float form: 23.3

Found identifier: типы\_и\_циклы

Found opening brackets : (

Found string: 'строка'

Found comma

Found identifier: s1

Found comma

Found identifier: целое

Closing brackets: )

Commentary has been inputed: | типы и циклы

Commentary has been inputed: | условия и операции

Found Keyword: кон

Found Keyword: алг

Found Keyword: цел

Found identifier: вернуть

Found Keyword: надо

Commentary has been inputed: | вернуть

Found Keyword: нач

Found Keyword: знач

Operation found: :=

Found Number in decimal form: 3

Found Keyword: кон

Found Keyword: алг

Found identifier: типы\_и\_циклы

Found opening brackets : (

Found Keyword: арг

Found Keyword: лит

Found identifier: первый

Found comma

Found Keyword: рез

Found Keyword: лит

Found identifier: второй

Found comma

Found Keyword: аргрез

Found Keyword: цел

Found identifier: третий

Closing brackets: )

Found Keyword: дано

Found identifier: первый

Operation found =: =

Found string with two dots: "строка"

Found Keyword: нач

Found identifier: симтаб

Found identifier: таблица

Opening Braces: [

Found Number in decimal form: 1

Semi Colon

Found Number in decimal form: 100

Found comma

Found Number in decimal form: 2

Semi Colon

Found Number in decimal form: 200

Closing braces: ]

Found Keyword: целтаб

Found identifier: просто

Found identifier: так

Opening Braces: [

Operation found: -

Found Number in decimal form: 5

Semi Colon

Found Number in decimal form: 0

Closing braces: ]

Found Keyword: цел

Found Keyword: таб

Found identifier: то\_же

Found identifier: самое

Opening Braces: [

Operation found: -

Found Number in decimal form: 4

Semi Colon

Found Number in decimal form: 1

Found comma

Found Number in decimal form: 50

Semi Colon

Found Number in decimal form: 55

Found comma

Found Number in decimal form: 7

Semi Colon

Found Number in decimal form: 34

Closing braces: ]

Found identifier: литтаб

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 0

Semi Colon

Found Number in decimal form: 10

Closing braces: ]

Found Keyword: вещтаб

Found identifier: вещтаблица

Opening Braces: [

Found Number in decimal form: 0

Semi Colon

Found Number in decimal form: 1

Closing braces: ]

Found Keyword: логтаб

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 50

Semi Colon

Found Number in decimal form: 60

Closing braces: ]

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 53

Closing braces: ]

Operation found: :=

Found Keyword: да

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 54

Closing braces: ]

Operation found: :=

Found Keyword: нет

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 55

Closing braces: ]

Operation found: :=

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 53

Closing braces: ]

Found Keyword: и

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 54

Closing braces: ]

Found Keyword: или

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 53

Closing braces: ]

Found Keyword: и

Found opening brackets : (

Found Keyword: не

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 54

Closing braces: ]

Found Keyword: или

Found identifier: тф

Opening Braces: [

Found Number in decimal form: 53

Closing braces: ]

Closing brackets: )

Found Keyword: ввод

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 0

Closing braces: ]

Found Keyword: вывод

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 0

Closing braces: ]

Found Keyword: ввод

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 1

Closing braces: ]

Found comma

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 2

Closing braces: ]

Found Keyword: вывод

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 1

Closing braces: ]

Found comma

Found Keyword: нс

Found comma

Found identifier: текст

Opening Braces: [

Found Number in decimal form: 2

Closing braces: ]

Found Keyword: алг

Found identifier: условия

Found Keyword: нач

Found Keyword: цел

Found identifier: а

Found comma

Found identifier: б

Found comma

Found identifier: в

Found Keyword: вещ

Found identifier: г

Found comma

Found identifier: д

Found comma

Found identifier: е

Found comma

Found identifier: ж

Found identifier: а

Operation found: :=

Found Number in decimal form: 123

Newline found: ;

Found identifier: б

Operation found: :=

Operation found: -

Found Number in decimal form: 10000

Newline found: ;

Found identifier: в

Operation found: :=

Found number in hexadecimal form: $FF(255)

Found identifier: г

Operation found: :=

Found number in float form: 1.23

Newline found: ;

Found identifier: д

Operation found: :=

Operation found: -

Found number in float form: 0.56

Newline found: ;

Found identifier: е

Operation found: :=

Found Number in decimal form: 1

Found identifier: е

Operation found: +

Found Number in decimal form: 4

Newline found: ;

Found identifier: ж

Operation found: :=

Found Number in decimal form: 5

Found identifier: E

Operation found: -

Found Number in decimal form: 7

Found Keyword: выбор

Found Keyword: при

Found identifier: а

Operation found =: =

Found Number in decimal form: 2

Semi Colon

Found identifier: а

Operation found: :=

Found Number in decimal form: 3

Found Keyword: при

Found identifier: а

Operation found =: =

Found Number in decimal form: 3

Semi Colon

Found identifier: а

Operation found: :=

Found Number in decimal form: 4

Found Keyword: при

Found identifier: а

Operation found =: =

Found Number in decimal form: 4

Semi Colon

Found identifier: а

Operation found: :=

Found Number in decimal form: 5

Newline found: ;

Found identifier: б

Operation found: :=

Found Number in decimal form: 4

Found Keyword: иначе

Found identifier: а

Operation found: :=

Found Number in decimal form: 6

Found identifier: все

Found Keyword: кон