МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСВЕННЫЙ УНИВЕРСИТЕТ

Кафедра САПР

ОТЧЁТ

о выполнении лабораторной работы №3

по дисциплине «Лингвистическое и программное обеспечение»

Выполнили: ст.гр. 21BA1

Ишенбеков Н.

Жанибекова А.

Женишбек к Э.

Проверил:

Финогеев А.А.

Пенза, 2023г.

**Цель работы:** разработать программу формирования файла постфиксной записи.

**БНФ реализуемого языка:**

<Программа> ::= <Объявление переменных> <Описание вычислений>

<Описание вычислений> ::= Begin <Список присваиваний> End

<Объявление переменных> ::= Var  <Список переменных> ;

<Список переменных> ::= <Идент> | <Идент> , <Список переменных>

<Список присваиваний>::= <Присваивание> |

                                             <Присваивание> <Список присваиваний>

<Присваивание> ::= <Идент> := <Выражение> ;

<Выражение> ::= <Ун.оп.> <Подвыражение> | <Подвыражение>

<Подвыражение> :: = ( <Выражение> ) | <Операнд> |

                                    < Подвыражение > <Бин.оп.> <Подвыражение>

<Ун.оп.> ::= "-"

<Бин.оп.> ::= "-" | "+" | "\*" | "/"

<Операнд> ::= <Идент> | <Const>

<Идент> ::= <Буква> <Идент> | <Буква>

<Const> ::= <Цифра> <Const>  | <Цифра>

**Листинг программы:**

#include <iostream>

#include <fstream>

#include <locale>

#include <cctype>

#include <string>

#include <cmath>

#include <vector>

#include <stack>

using namespace std;

//Классы

//11-Var

//12-Begin

//13-End

//2-идентификаторы

//3-константа

//4-символ присваивания

//5-бинарные операции сложение и вычитание

//6-бинарные операции умножение и деление

//7-унарные операции

//8-символы разделения

//9-скобка открывающая

//10-скобка закрывающая

using namespace std;

struct lexem {

char name[10];

int type;

int line;

}lex;

stack <lexem> opstk;

int per\_count = 0;

int in\_str\_count = 0;

fstream fi, fo;

//Функция приоритета операции Возвращает приоритет лексем

bool precedence(lexem one, lexem two)

{

if (one.type == 7) return true;

if (one.type == 9)return false;

else if (two.type == 10) return true;

if (two.type == 9)return false;

if (one.type == 7) return true;

else if (two.type == 7)return false;

if (one.type == 6) return true;

else if (two.type == 6)return false;

if (one.type == 5)return true;

else if (two.type == 5)return false;

return false;

}

void read()

{

fi >> lex.name >> lex.type >> lex.line;

}

void PostfixOutput()

{

int number\_str = 1;

int flag = 0, flag2 = 0;

fi.open("C:\\21VA1\\4 term\\LIPO\\LIPO 1\\lixem.txt", ios\_base::in);

fo.open("postfix.txt", ios::out);

if (!fi.is\_open() && !fo.is\_open())

{

cout << "No file!";

exit(2);

}

read();

if (lex.line == 1 && lex.name[0] == 'V') read();

while (lex.line != 2 && lex.name[0] != 'B')

{

if (lex.type == 2)fo << lex.name << " ";

read();

}

fo << ";";

while (flag != 1)

{

if (lex.type == 13 && lex.name[0] == 'E') { flag = 1; goto end; }

else flag2 = 0;

while (flag2 == 0)

{

read();

if (lex.type == 13 && lex.name[0] == 'E') { flag = 1; goto end; }

if (lex.line != number\_str)

{

number\_str++;

fo << "\n";

}

if (number\_str == 2) { number\_str = 3; }

if (lex.type == 2 || lex.type == 3) fo << lex.name << " ";

else {

while (!opstk.empty() && precedence(opstk.top(), lex))

{

fo << opstk.top().name << " ";

opstk.pop();

}

if ((opstk.empty() || (lex.name[0] != ')'))) {

if (lex.type == 7)

{

lex.name[0] = '~';

opstk.push(lex);

}

else if (lex.type != 8) opstk.push(lex);

}

else {

if (opstk.top().type != 9) fo << opstk.top().name << " ";

opstk.pop();

}

}

if (lex.type == 8)flag2 = 1;

}

while (!opstk.empty()) {

if (opstk.top().type != 9)

fo << opstk.top().name << " ";

opstk.pop();

}

}

end: {fo << "\n"; }

fi.close();

fo.close();

}

void main()

{

setlocale(LC\_ALL, "russian");

PostfixOutput();

system("pause");

return;

}

**Результат работы программы:**



