МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

кафедра «Вычислительная техника»

**Лабораторная работа №1**

по дисциплине: «Теория автоматов».

Вариант №12

Выполнил:

студент гр. ИВТАПбд-21

Кондратьев П.С.

Проверил:

преподаватель кафедры ВТ

Лылова А.В.

г. Ульяновск, 2018

**Задание:**

Требуется реализовать автомат указанного типа для разбора и преобразования циклов, написанных на языке, соответствующих требованию варианта задания. Вариант задания также содержит типы входного и выходного циклов. Достаточно обрабатывать один цикл в случае множественных или вложенных циклов во входных данных.

Входной язык: Pascal

Входной цикл: for

Выходной цикл: repeat

Модель автомата: Мили

**Исходный код программы:**

string S = "S0 =>";

ifstream f;

ofstream g;

string programm = "";

f.open("test.pas");

g.open("newTest.pas");

S += "S1 => ";

while (!f.eof()) {

programm += f.get();

}

programm.resize(programm.length() - 1);

S += "S2 => ";

vector<string> arr;

string delim("\n");

size\_t prev = 0;

size\_t next;

size\_t delta = delim.length();

while ((next = programm.find(delim, prev)) != string::npos) {

//Отладка-start

string tmp = programm.substr(prev, next - prev);

//Отладка-end

arr.push\_back(programm.substr(prev, next - prev));

prev = next + delta;

}

//Отладка-start

string tmp = programm.substr(prev);

//Отладка-end

arr.push\_back(programm.substr(prev));

S += "S3 => ";

string index = arr[3].substr(7, 1); //вырезаем индекс i

int id = arr[7].find("to");

string index1 = arr[7].substr(13, id - 14); //вырезаем количество i

string index2 = arr[3].substr(4, 1); //вырезаем индекс n

int id1 = arr[6].find(";");

string index3 = arr[6].substr(9, id1 - 9); //вырезаем количество n

string index4 = arr[6].substr(4, 1); //вырезаем инцилизация n

string index5 = arr[7].substr(18, 1); //вырезаем n в цикле

string index6 = arr[7].substr(8, 1); //вырезаем i в цикле

S += "S4 => ";

g << arr[0] + "\n" << arr[1] + "\n" << arr[2] + "\n";

g << "\t" + index2 + ", " << index + ": integer;";

g << arr[4] + "\n" << arr[5] + "\n" << "\t" + index4 + " := " << index3 + ";\n";

g << "\t" + index + " := " + index1 + ";\n";

g << "\trepeat\n";

g << arr[8] + "\n";

S += "S5 => ";

if (stol(index3) > stol(index1)) {

g << "\t\t" + index6 + ":= " << index6 + " + 1;\n";

g << "\tuntil " + index6 + " > " + index2 + ";\n";

} else {

g << "\t\t" + index6 + ":= " << index6 + " - 1;\n";

g << "\tuntil " + index6 + " < " + index2 + ";\n";

}

g << arr[9];

f.close();

g.close();

S += "S0";

cout << "Состояния:\n" + S;

**Блок-схема и граф:**

****

**Кодирование:**

|  |  |
| --- | --- |
| Y | Соответствующие функции |
| Y1 | ifstream f;  ofstream g;  string programm = "";  f.open("test.pas");  g.open("newTest.pas"); |
| Y2 | programm += f.get(); |
| Y3 | programm.resize(programm.length() - 1);  vector<string> arr;  string delim("\n");  size\_t prev = 0;  size\_t next;  size\_t delta = delim.length(); |
| Y4 | string tmp = programm.substr(prev, next - prev);  arr.push\_back(programm.substr(prev, next - prev));  prev = next + delta; |
| Y5 | string tmp = programm.substr(prev);  arr.push\_back(programm.substr(prev)); |
| Y6 | string index = arr[3].substr(7, 1); //вырезаем индекс i  int id = arr[7].find("to");  string index1 = arr[7].substr(13, id - 14); //вырезаем количество i  string index2 = arr[3].substr(4, 1); //вырезаем индекс n  int id1 = arr[6].find(";");  string index3 = arr[6].substr(9, id1 - 9); //вырезаем количество n  string index4 = arr[6].substr(4, 1); //вырезаем инцилизация n  string index5 = arr[7].substr(18, 1); //вырезаем n в цикле  string index6 = arr[7].substr(8, 1); //вырезаем i в цикле |
| Y7 | g << arr[0] + "\n" << arr[1] + "\n" << arr[2] + "\n";  g << "\t" + index2 + ", " << index + ": integer;";  g << arr[4] + "\n" << arr[5] + "\n" << "\t" + index4 + " := " << index3 + ";\n";  g << "\t" + index + " := " + index1 + ";\n";  g << "\trepeat\n";  g << arr[8] + "\n"; |
| Y8 | g << "\t\t" + index6 + ":= " << index6 + " + 1;\n";  g << "\tuntil " + index6 + " > " + index2 + ";\n"; |
| Y9 | g << "\t\t" + index6 + ":= " << index6 + " - 1;\n";  g << "\tuntil " + index6 + " < " + index2 + ";\n"; |
| F | !f.eof() |
| F1 | (next = programm.find(delim, prev)) != string::npos |
| F2 | stol(index3) > stol(index1) |

**Контрольный пример:**

program qq;

var

n, i: integer;

begin

n := 10;

for i := 0 to n do

writeln(i);

end.

**Результат работы программы:**

program qq;

var

n, i: integer;

begin

n := 10;

i := 0;

repeat

writeln(i);

i:= i + 1;

until i > n;

end.

**Вывод состояний:**

S0 => S1 => S2 => S3 => S4 => S5 => S0