Министерство образования Республики Беларусь

УО «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

ОТЧЕТ

по лабораторной работе № 1

по предмету

Языки программирования на тему:

РАСПОЗНАВАНИЕ И ПОИСК ЛЕСЕМ

Вариант 5

Выполнили : Цыбулько К.Д.

Студенты гр. 851002 Трухан К.А.

Проверила : Шостак Е. В.

Минск 2019

**Регулярное выражение**

Константы:

DigBin=[’0’,’1’]

DigOx=[’2’-’7’]

DigHex=[’8’,’9’]

LetHex=[’A’-’F’]

Bin=[’b’]

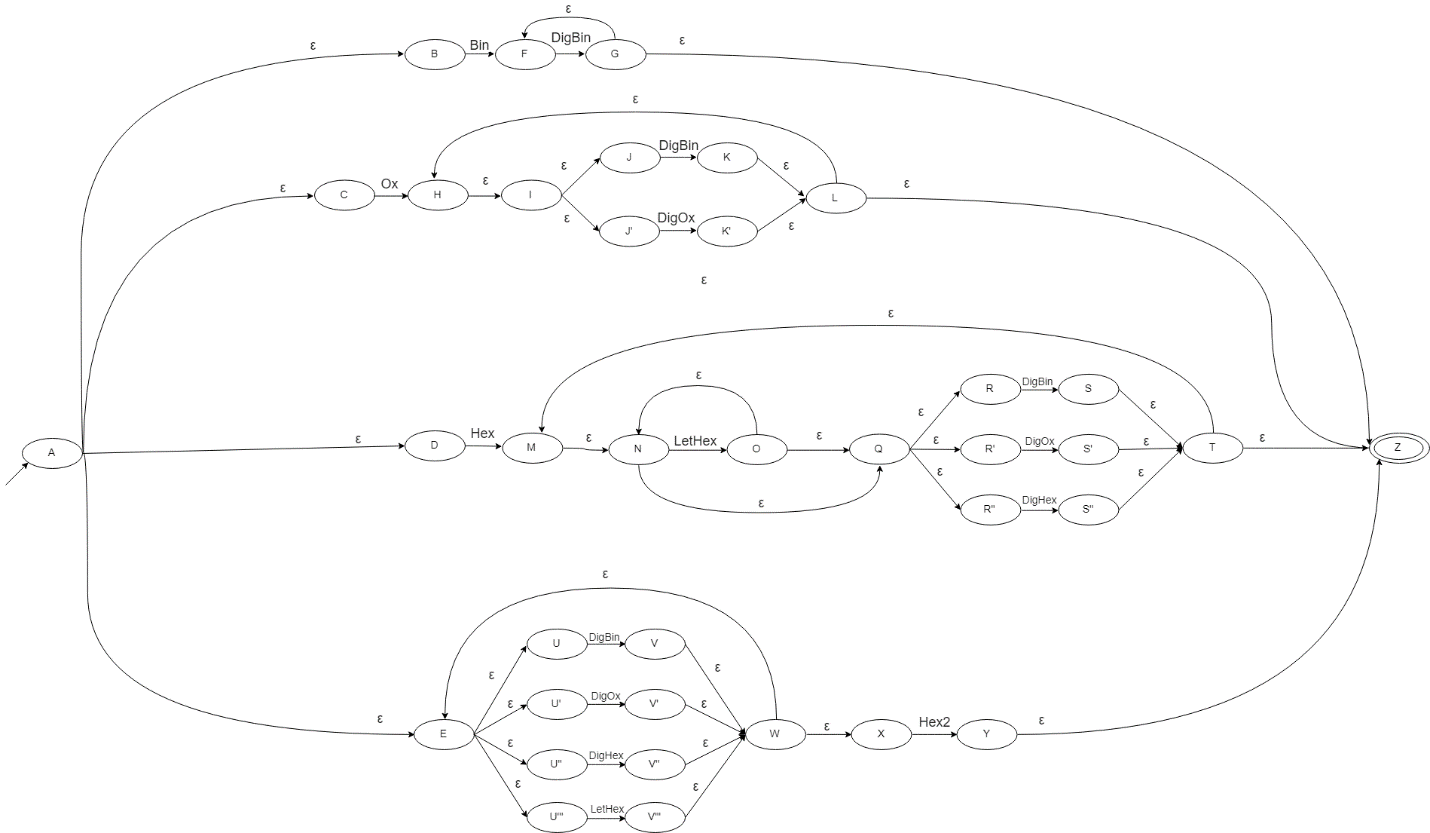
Ox=[’o’,’O’]

Hex=[’h’,’H’]

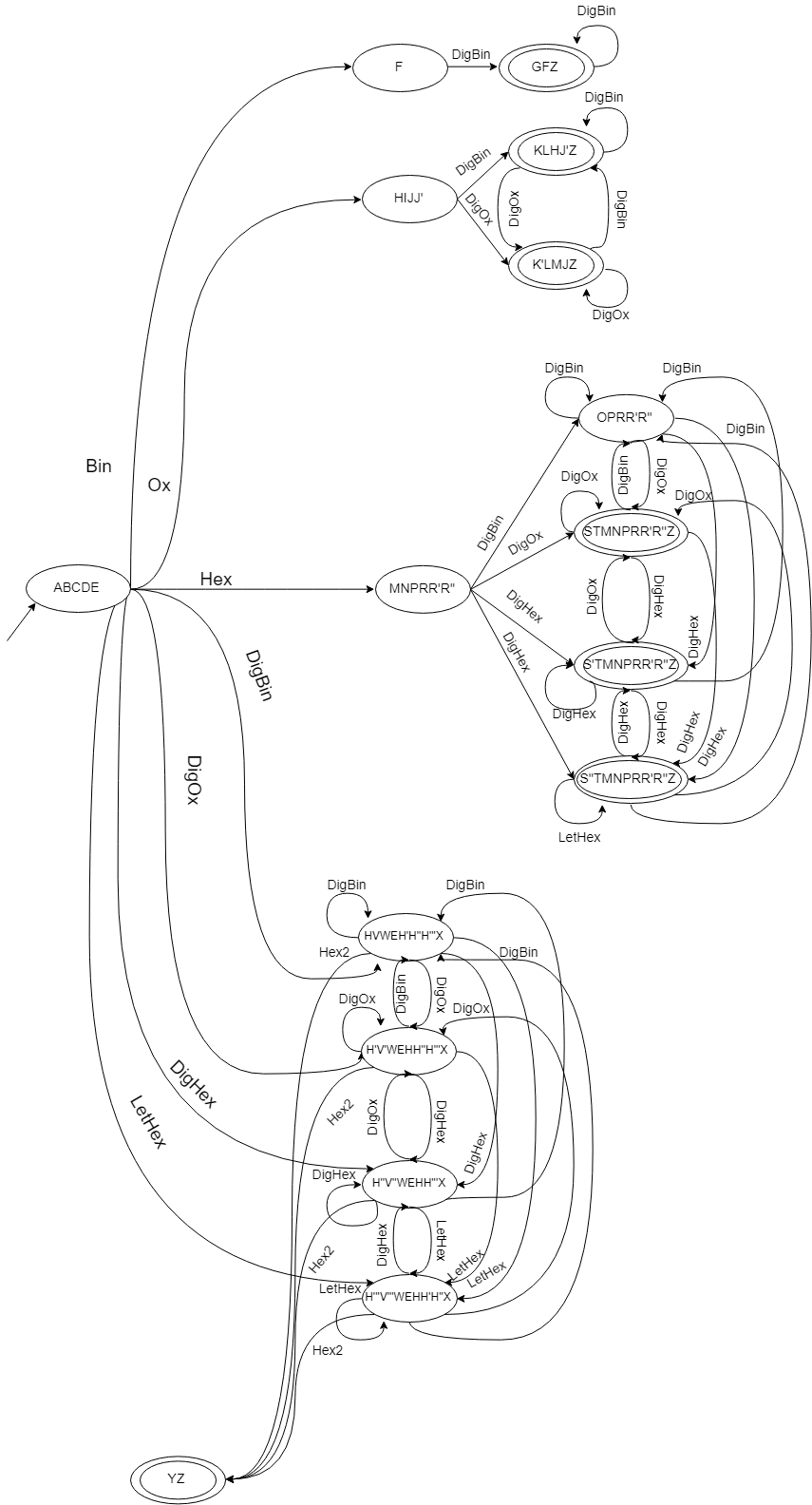
Hex2=[’0x’]

Number=((Bin(DigBin)+)|(Ox(DigBin|DigOx)+)|(Hex((LetHex)\*(DigBin|DigOx|DigHex)+)+)|

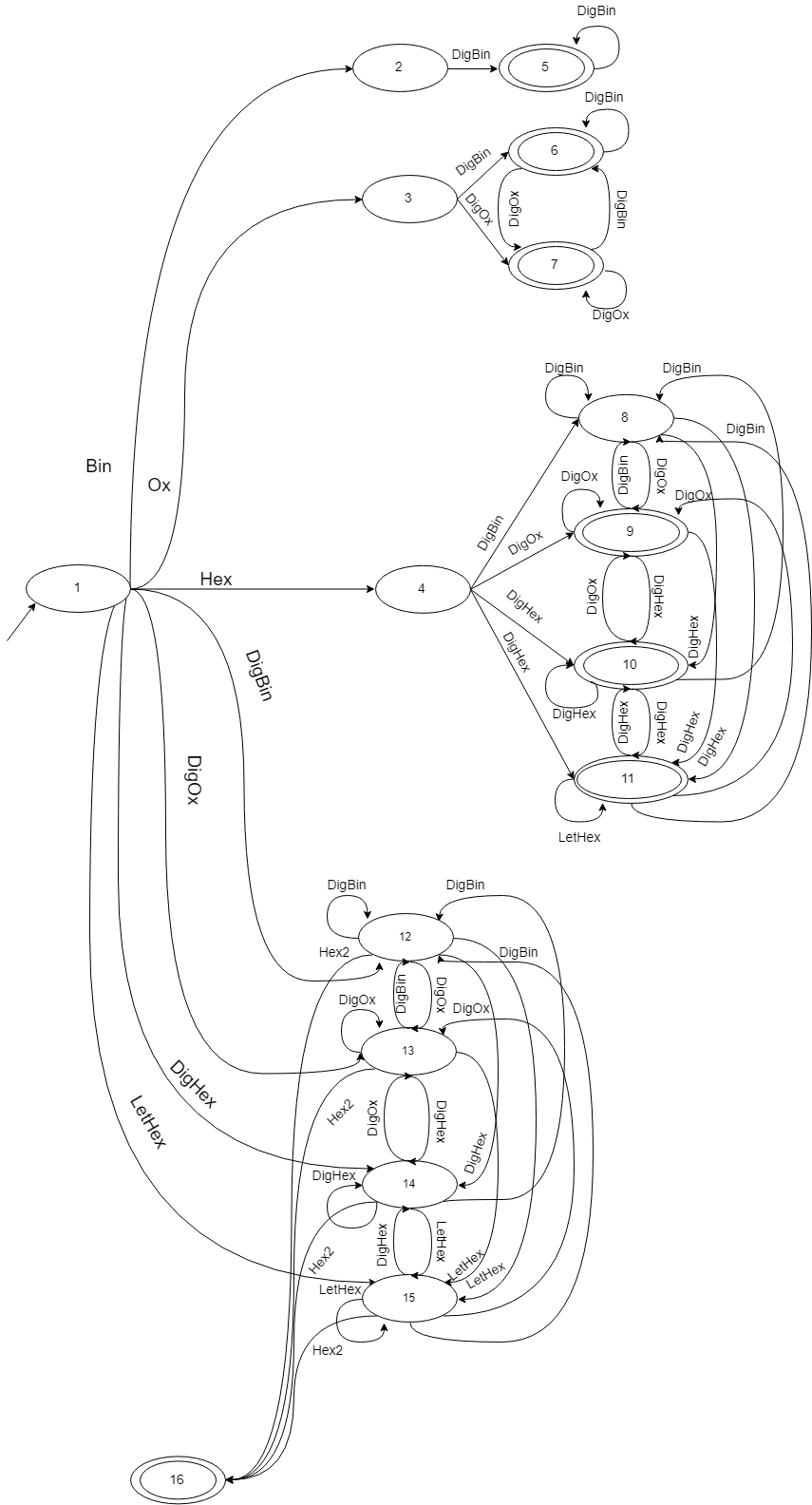
((DigBin|DigOx|DigHex|LetHex)+Hex2) )

**НКА (для регулярного выражения)**

**ДКА (для регулярного выражения)**

****

**ДКА(готовый к программной реализации)**

****

**Таблица переходов**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Error** | **DigBin** | **DigOx** | **DigHex** | **Bin** | **Ox** | **Hex** | **Hex2** | **LetHex** |
| **0** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | 0 | 12 | 13 | 14 | 2 | 3 | 4 | 0 | 15 |
| **2** | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 6 | 7 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 0 | 9 | 10 | 11 | 0 | 0 | 0 | 0 | 8 |
| **5** | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **6** | 0 | 6 | 7 | 0 | 0 | 0 | 0 | 0 | 0 |
| **7** | 0 | 6 | 7 | 0 | 0 | 0 | 0 | 0 | 0 |
| **8** | 0 | 9 | 10 | 11 | 0 | 0 | 0 | 0 | 8 |
| **9** | 0 | 9 | 10 | 11 | 0 | 0 | 0 | 0 | 8 |
| **10** | 0 | 9 | 10 | 11 | 0 | 0 | 0 | 0 | 8 |
| **11** | 0 | 9 | 10 | 11 | 0 | 0 | 0 | 0 | 8 |
| **12** | 0 | 12 | 13 | 14 | 0 | 0 | 0 | 17 | 15 |
| **13** | 0 | 12 | 13 | 14 | 0 | 0 | 0 | 17 | 15 |
| **14** | 0 | 12 | 13 | 14 | 0 | 0 | 0 | 17 | 15 |
| **15** | 0 | 12 | 13 | 14 | 0 | 0 | 0 | 17 | 15 |
| **16** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Код программы, реализующий работы ДКА**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <conio.h>

#define NumberStates 19

#define CharType 10

int ResultPriority(char c,char s) {

switch (c) {

case '0':

case '1':

return 1;

case '2':

case '3':

case '4':

case '5':

case '6':

case '7':

return 2;

case '8':

case '9':

return 3;

case 'b':

return 4;

case 'o':

case 'O':

return 5;

case 'h':

case 'H':

return 6;

case 'x': {

if (s=='0') return 7;

else return 0;

}

case 'A':

case 'B':

case 'C':

case 'D':

case 'E':

case 'F':

return 8;

default:

return 0;

}

}

int main() {

int IsFinalStates[NumberStates] = { 0,0,0,0,0,1,1,1,0,1,1,1,0,0,0,0,1};

int TransitionsArray[NumberStates][CharType] = {

{0,0,0,0,0,0,0,0,0},

{0,12,13,14,2,3,4,0,15},

{0,5,0,0,0,0,0,0,0},

{0,6,7,0,0,0,0,0,0},

{0,9,10,11,0,0,0,0,8},

{0,5,0,0,0,0,0,0,0},

{0,6,7,0,0,0,0,0,0},

{0,6,7,0,0,0,0,0,0},

{0,9,10,11,0,0,0,0,8},

{0,9,10,11,0,0,0,0,8},

{0,9,10,11,0,0,0,0,8},

{0,9,10,11,0,0,0,0,8},

{0,12,13,14,0,0,0,16,15},

{0,12,13,14,0,0,0,16,15},

{0,12,13,14,0,0,0,16,15},

{0,12,13,14,0,0,0,16,15},

{0,0,0,0,0,0,0,0,0},

};

char key[10];

while (key != "exit") {

puts("Please, enter a number in a correct postfix form: ");

char str[50];

gets(str);

int k = 0;

char buffer[50];

for (int i = 0; i <= strlen(str); i++) {

if ((str[i] != ' ') && (str[i] != '\0')) {

buffer[k] = str[i];

k++;

}

else {

buffer[k] = '\0';

int State = 1;

int j;

j = strlen(buffer) - 1;

while ((j >= 0) && (State != 0)) {

State = TransitionsArray[State][ResultPriority(buffer[j], buffer[j-1])];

if (ResultPriority(buffer[j], buffer[j - 1]) == 7) j--;

j--;

}

int Result = IsFinalStates[State];

if (Result == 1) {

puts("\nCorrect format!\n");

}

else {

puts("\nUncorrect format\n");

k = 0;

}

}

}

}

system("pause");

return 0;

}