МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА.

Институт радиоэлектроники и информационных технологий

Отчет

по лабораторной работе № 2

по дисциплине

«Основы теории интеллектуальных вычислительных систем»

Руководитель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мисевич П. В.

Студент:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Игнаков К. М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тихонов В. Д.

группа 19-ВМ

Работа защищена «\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

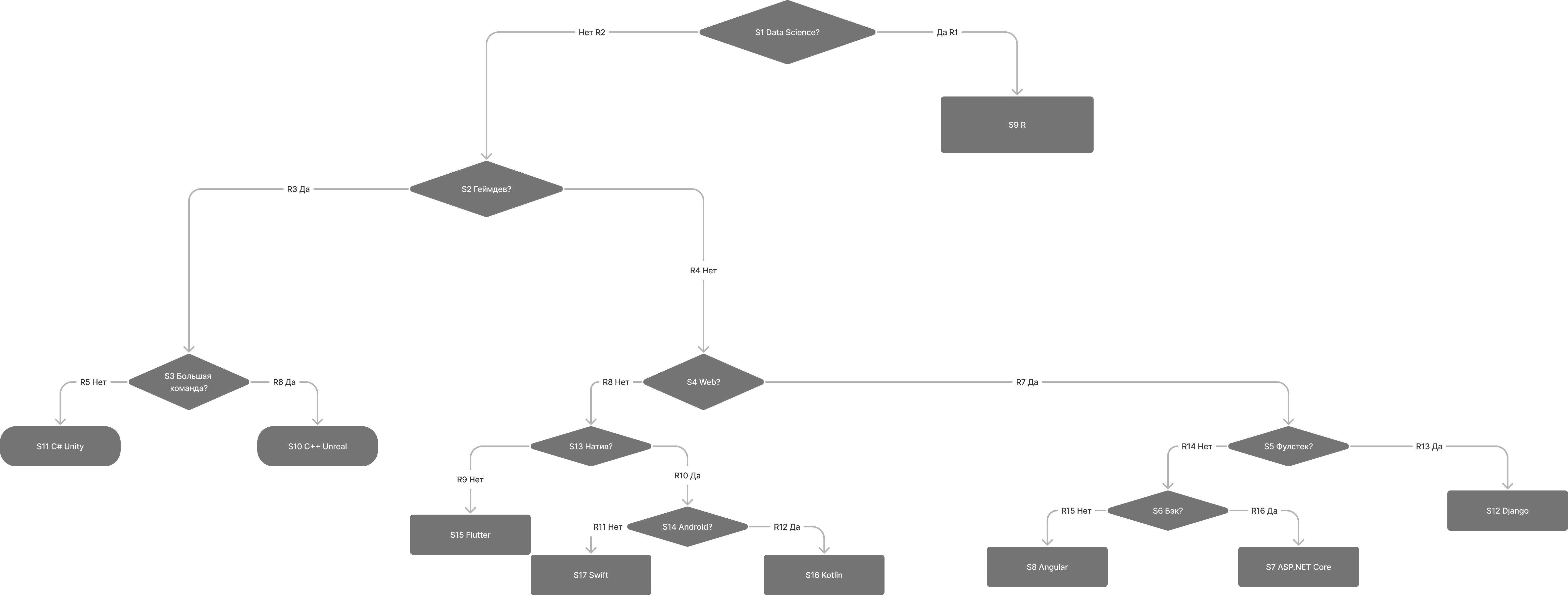
Нижний Новгород 2022

## Постановка задачи

Реализация графа вопрос-ответ программой с типовыми элементами ЭС.

## Ход работы

Дерево вопрос-ответ:



Код программы (формы, тк в ней располагаются вся логика для инкапсуляции):

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace ЛР2\_Client\_Server

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

int[] STOP\_;//

string[] S\_;

int[,] rools\_;

private void Form1\_Load(object sender, EventArgs e)

{

// массив событий порождающего сценария множества S

S\_ = new string[100];// параметр: настроеное на 100 проодукционных праввил с 3 исходами

S\_[1] = "Data Science ?";

S\_[2] = "Gamedev ?";

S\_[3] = "Big company?";

S\_[4] = "Web?";

S\_[5] = "Fullstack?";

S\_[6] = "Back-end?";

S\_[7] = "ASP.Net Core";

S\_[8] = "Angular";

S\_[9] = "R";

S\_[10] = "Unreal Engine";

S\_[11] = "C# Unity";

S\_[12] = "Python Django";

S\_[13] = "Native?";

S\_[14] = "Android?";

S\_[15] = "Flutter";

S\_[16] = "Kotlin";

S\_[17] = "Swift";

// закодированные логические правила: продукционная БЗ начало

//int[,] rools\_ = new int[100, 2];// параметр: настроеное на 100 проодукционных праввил с 3 исходами

rools\_ = new int[100, 2];// параметр: настроеное на 100 проодукционных праввил с 3 исходами

rools\_[1, 1] = 9;

rools\_[1, 0] = 2;

rools\_[2, 1] = 3;

rools\_[2, 0] = 4;

rools\_[3, 1] = 10

rools\_[3, 0] = 11

rools\_[4, 1] = 5;

rools\_[4, 0] = 13

rools\_[5, 1] = 12

rools\_[5, 0] = 6;

rools\_[6, 1] = 7;

rools\_[6, 0] = 8;

rools\_[13, 1] = 14;

rools\_[13, 0] = 15;

rools\_[14, 1] = 16;

rools\_[14, 0] = 17;

// закодированные логические правила: продукционная БЗ окончание

//подготавливаем хранилище индексов концевых событий начало

STOP\_ = new int[100];// параметр: настроеное на 100 проодукционных праввил с 3 исходами

// при отрацательном значении массива STOP\_ событие концевое

// метим концевые собятия

for (int i = 0; i < 100; i++)

{

STOP\_[i] = 7;

}

STOP\_[9] = -7;

STOP\_[11] = -7;

STOP\_[10] = -7;

STOP\_[15] = -7;

STOP\_[17] = -7;

STOP\_[16] = -7;

STOP\_[8] = -7;

STOP\_[7] = -7;

STOP\_[12] = -7;

// метим концевые собятия

//подготавливаем хранилище индексов концевых событий окончание

// инициализация стартового события

textBox1.Text = S\_[1];

textBox3.Text = "1";

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

textBox2.Text = "1";

}

private void button2\_Click(object sender, EventArgs e)

{

textBox2.Text = "0";

}

private void button3\_Click(object sender, EventArgs e)

{

//База фактов из 2-х переменных начало

int SDB = 0;// антецедент импликации Si - левая часть правила

int RDB = 0;// антецедент импликации Rj - левая часть правила

int S\_cos\_ = 0;// косеквент импликации

//наполнение БФ информацией о состоянии предметной области начало

SDB = Int32.Parse(textBox3.Text);

RDB = Int32.Parse(textBox2.Text);

//наполнение БФ информацией о состоянии предметной области окончание

//База фактов из 2-х переменных окончание

// процедуры событий !! события отделены от правил и процедур вывода!

// интерпретатор начало // в явном виде нет цикла

S\_cos\_ = rools\_[SDB, RDB];

// textBox2.Text = S\_cos\_.ToString();

textBox4.Text = Convert.ToString(S\_cos\_);

// интерпретатор окончение

// button4\_Click(sender, e);

}

private void button4\_Click(object sender, EventArgs e)

{

//string[] S\_ = new string[100];// параметр: настроеное на 100 проодукционных праввил с 3 исходами

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* код кнопки

int S\_cos\_ = 0;// косеквент импликации

//наполнение БФ информацией о состоянии предметной области начало

S\_cos\_ = Int32.Parse(textBox4.Text);

textBox1.Text = S\_[S\_cos\_];

textBox3.Text = Convert.ToString(S\_cos\_);

if (STOP\_[S\_cos\_] < 0)

{

MessageBox.Show("концевое событие");

Environment.Exit(0);

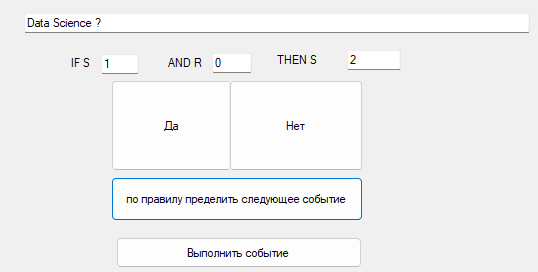
}

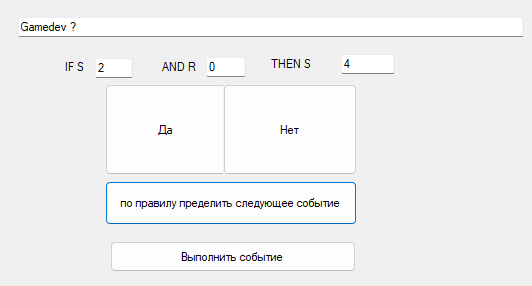
}

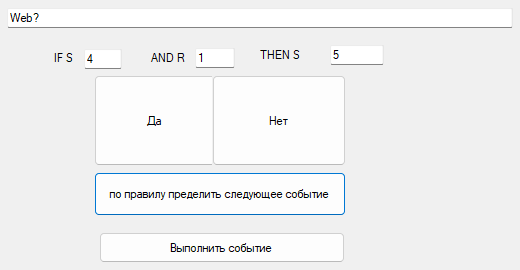
}

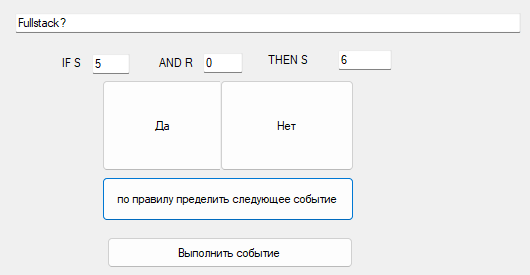
}

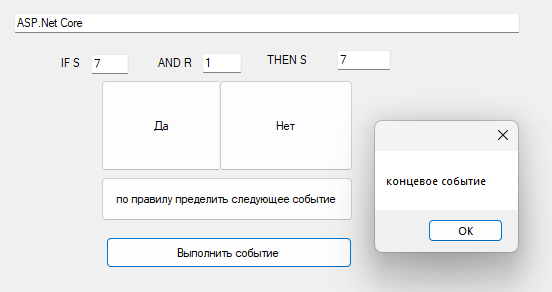
Протокол решения задачи:











## **Вывод**

Данный подход в решении задачи нуждается в небольшой справке для реализации, но гораздо удобнее и проще расширяемый для небольших деревьев, по сравнению с примером, выполненным в лабораторной работе №1. Минусы данного метода в том, что для большой структуры вопрос-ответ данный метод становится сложным в поддержке.