Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Рязанский государственный радиотехнический университет

имени В.Ф. Уткина»

Кафедра АСУ

Отчёт о практической работе №7

«Метод ELECTRE I»

По дисциплине

«Математические основы принятия решений»

Выполнилa

ст. гр. 135

Кузнецова М.А.

Проверил:

Челебаева Ю.А.

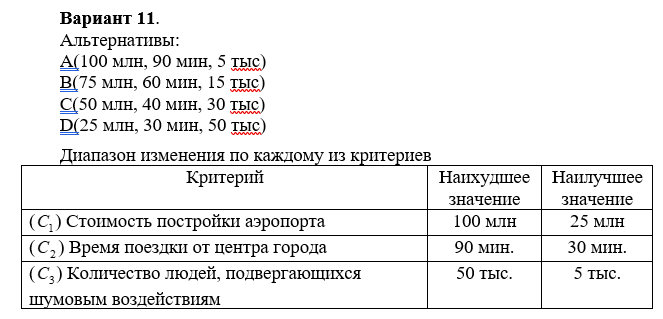
Челебаев С.В.

Рязань 2023

**Цель работы**

Реализовать метод ELECTRE I для своей предметной области на ЯВУ.

**Вариант задания**



Листинг

using System;

using System.Drawing;

using System.Linq;

using System.Windows.Forms;

namespace ELECTRE

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

dgV\_Alternativa.RowCount = 4;

dgV\_Alternativa.ColumnCount = 3;

dgV\_Soglasie.RowCount = 4;

dgV\_Soglasie.ColumnCount = 4;

dgV\_Nesoglasie.RowCount = 4;

dgV\_Nesoglasie.ColumnCount = 4;

dgV\_Alternativa.Rows[0].Cells[0].Value = 100;

dgV\_Alternativa.Rows[0].Cells[1].Value = 90;

dgV\_Alternativa.Rows[0].Cells[2].Value = 5;

dgV\_Alternativa.Rows[1].Cells[0].Value = 75;

dgV\_Alternativa.Rows[1].Cells[1].Value = 60;

dgV\_Alternativa.Rows[1].Cells[2].Value = 15;

dgV\_Alternativa.Rows[2].Cells[0].Value = 50;

dgV\_Alternativa.Rows[2].Cells[1].Value = 40;

dgV\_Alternativa.Rows[2].Cells[2].Value = 30;

dgV\_Alternativa.Rows[3].Cells[0].Value = 25;

dgV\_Alternativa.Rows[3].Cells[1].Value = 30;

dgV\_Alternativa.Rows[3].Cells[2].Value = 50;

nud\_w1.Text = "3";

nud\_w2.Text = "2";

nud\_w3.Text = "1";

}

int[,] matrix = new int[4, 3]; //Матрица альтернатив

int[] w = new int[3]; //Матрица весов

double[] L = new double[3]; //Матрица длин шкал

double[,] C\_matrix = new double[4, 4]; //Матрица согласия

double[,] D\_matrix = new double[4, 4]; //Матрица несогласия

public void Row\_Compare\_MC(int[,] mtr)

{

int c1 = 0;

int c2 = 0;

int n = 0; //Номер строки

double sum\_w = w[0] + w[1] + w[2]; //Сумма весов

for (int i = 0; i < 4; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if ((i1 == 0) & (i == 0) & (n == 0)) continue;

if (((i1 == 1) & (i == 1) & (n == 1))) continue;

if (((i1 == 2) & (i == 2) & (n == 2))) continue;

if ((i1 == 3) & (i == 3) & (n == 3)) continue;

int sum\_row = 0;

for (int j = 0; j < 3; j++)

{

if (mtr[i, j] <= mtr[i1, j])

{

sum\_row += w[j];

}

}

if (c2 > 3)

{

c1 += 1;

c2 = 0;

}

if (c1 == c2)

{

C\_matrix[c1, c2] = 0;

c2 += 1;

}

C\_matrix[c1, c2] = sum\_row / sum\_w;

c2 += 1;

}

n += 1;

}

}

public void Row\_Compare\_MD(int[,] mtr)

{

int c1 = 0;

int c2 = 0;

int n = 0; //Номер строки

double[] d = new double[3]; //Индекс несогласия

double res = 0.0;

for (int i = 0; i < 4; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if ((i1 == 0) & (i == 0) & (n == 0)) continue;

if (((i1 == 1) & (i == 1) & (n == 1))) continue;

if (((i1 == 2) & (i == 2) & (n == 2))) continue;

if ((i1 == 3) & (i == 3) & (n == 3)) continue;

Array.Clear(d, 0, 2);

for (int j = 0; j < 3; j++)

{

if (mtr[i, j] > mtr[i1, j])

{

res = (mtr[i, j] - mtr[i1, j]) / L[j];

d[j] = res;

}

}

if (c2 > 3)

{

c1 += 1;

c2 = 0;

}

if (c1 == c2)

{

D\_matrix[c1, c2] = 0;

c2 += 1;

}

D\_matrix[c1, c2] = d.Max();

c2 += 1;

}

n += 1;

}

}

//Создание матриц согласия и несогласия

private void btn\_Create\_Mtrxs\_Click\_1(object sender, EventArgs e)

{

if ((Convert.ToString(nud\_l1.Value) == "0") ||

(Convert.ToString(nud\_l2.Value) == "0") ||

(Convert.ToString(nud\_l3.Value) == "0") ||

(Convert.ToString(nud\_w1.Value) == "0") ||

(Convert.ToString(nud\_w2.Value) == "0"))

{

MessageBox.Show("Заполните все длины шкал!", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

//Заполнение матрицы из таблицы

for (int i = 0; i < 4; i++)

{

for (int j = 0; j < 3; j++)

{

matrix[i, j] = Convert.ToInt32(dgV\_Alternativa.Rows[i].Cells[j].Value);

}

}

//Заполнение матрицы весов критериев

w[0] = 3;

w[1] = 2;

w[2] = 1;

//Заполнение матрицы длин шкал

L[0] = Convert.ToInt32(nud\_l1.Value);

L[1] = Convert.ToInt32(nud\_l2.Value);

L[2] = Convert.ToInt32(nud\_l3.Value);

Row\_Compare\_MC(matrix);

Row\_Compare\_MD(matrix);

var n = dgV\_Soglasie.RowCount;

var m = dgV\_Soglasie.ColumnCount;

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

{

for (int j = 0; j < m; j++)

{

dgV\_Soglasie[j, i].Value = Math.Round(C\_matrix[i, j], 2).ToString(); //Вывод матрицы согласия

dgV\_Nesoglasie[j, i].Value = Math.Round(D\_matrix[i, j], 2).ToString(); //Вывод матрицы несогласия

}

}

}

//Расчет лучшей альтернативы

private void button2\_Click\_1(object sender, EventArgs e)

{

if (tb\_sogl.Text == "" || tb\_nesogl.Text == "")

{

MessageBox.Show("Заполните все уровни!", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

//Задание уровней согласия и несогласия

double a = Convert.ToDouble(tb\_sogl.Text);

double l = Convert.ToDouble(tb\_nesogl.Text);

int[] sum\_a = new int[4];

int[] sum\_l = new int[4];

var n = dgV\_Soglasie.RowCount;

var m = dgV\_Soglasie.ColumnCount;

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

{

for (int j = 0; j < m; j++)

{

if (Convert.ToDouble(dgV\_Soglasie[j, i].Value) >= a)

{

dgV\_Soglasie[j, i].Style.BackColor = Color.Green;

sum\_a[i] += 1;

}

if (Convert.ToDouble(dgV\_Nesoglasie[j, i].Value) <= l && Convert.ToDouble(dgV\_Soglasie[j, i].Value) != 0)

{

dgV\_Nesoglasie[j, i].Style.BackColor = Color.Green;

sum\_l[i] += 1;

}

}

}

int max1 = sum\_a[0];

int max2 = sum\_l[0];

int maxind1 = 0;

int maxind2 = 0;

for (int i = 0; i < 4; i++)

{

if (sum\_a[i] > max1)

{

max1 = sum\_a[i];

maxind1 = i;

}

if (sum\_l[i] > max2)

{

max2 = sum\_l[i];

maxind2 = i;

}

}

if (maxind1 == maxind2)

{

label\_result.Text = "";

if (maxind1 == 0)

label\_result.Text += "Наилучшая альтернатива - A";

if (maxind1 == 1)

label\_result.Text += "Наилучшая альтернатива - B";

if (maxind1 == 2)

label\_result.Text += "Наилучшая альтернатива - C";

if (maxind1 == 3)

label\_result.Text += "Наилучшая альтернатива - D";

}

else label\_result.Text = "Необходимо продолжить \nвычисления." +

"На данный момент \n" +

"НЕТ лучшей альтернативы.";

}

private void tb\_sogl\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!(Char.IsDigit(e.KeyChar)))

{

if (e.KeyChar != (char)Keys.Back && (e.KeyChar != ','))

{

e.Handled = true;

}

}

}

private void tb\_nesogl\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!(Char.IsDigit(e.KeyChar)))

{

if (e.KeyChar != (char)Keys.Back && (e.KeyChar != ','))

{

e.Handled = true;

}

}

}

private void nud\_l1\_ValueChanged(object sender, EventArgs e)

{

}

}

}

**Результаты выполнения программы.**

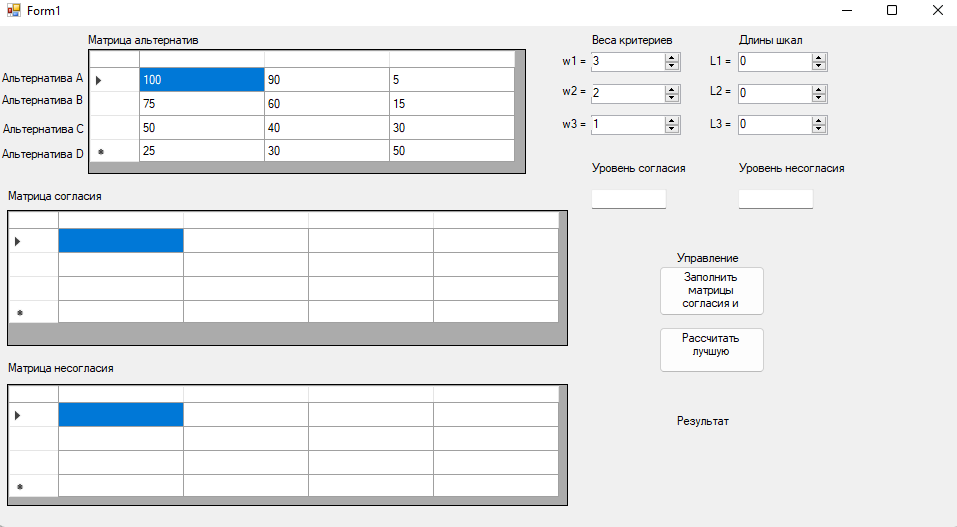


Рисунок 1 – Приветственная форма программы

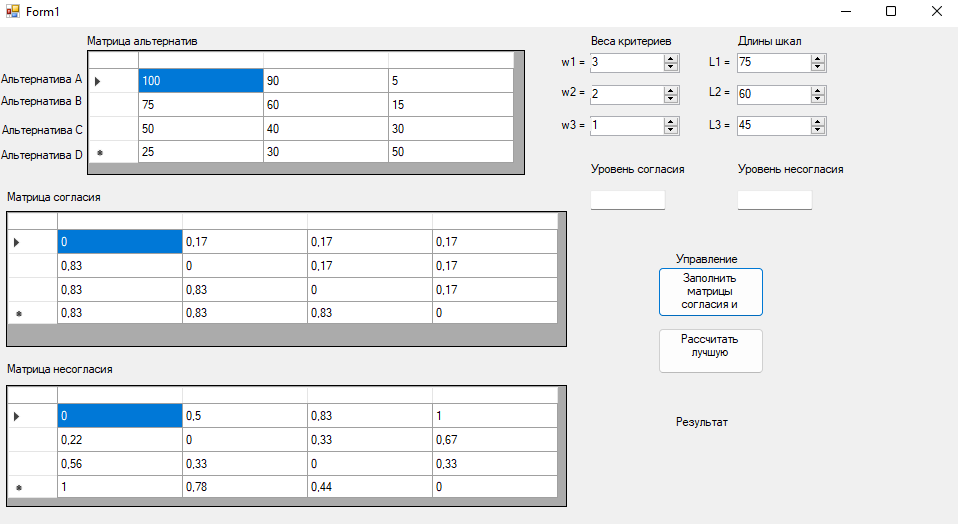


Рисунок 2 – Заполненные матрицы согласия и несогласия

Из рисунка 2 видно, что уровень согласия = 0.83, а уровень несогласия = 0,22. Введем их и рассчитаем лучшую альтернативу (рисунок 3).

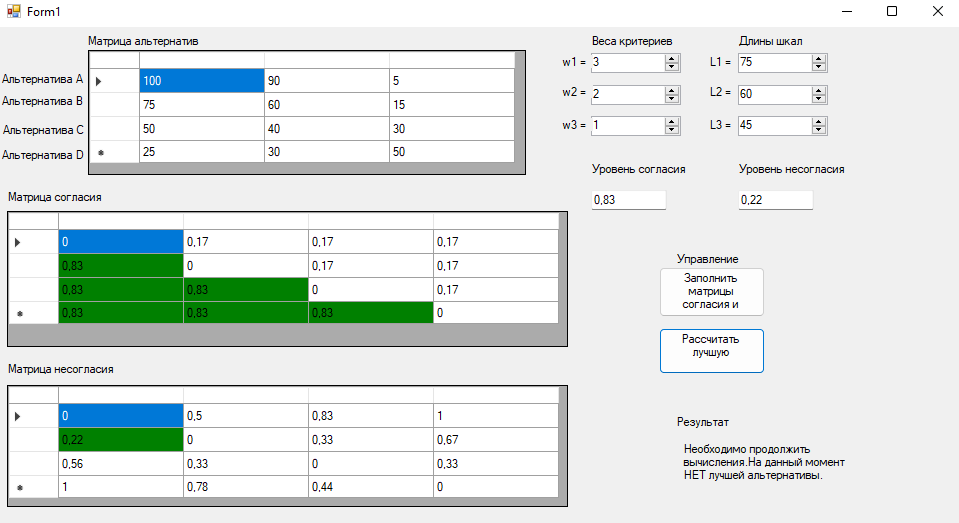


Рисунок 3 – Результат расчета лучшей альтернативы

Видно, из рисунка 3, что требуется продолжать расчет, следовательно введем новые уровни согласия (0.17) и несогласия (0.33). Результат представлен на рисунке 4.

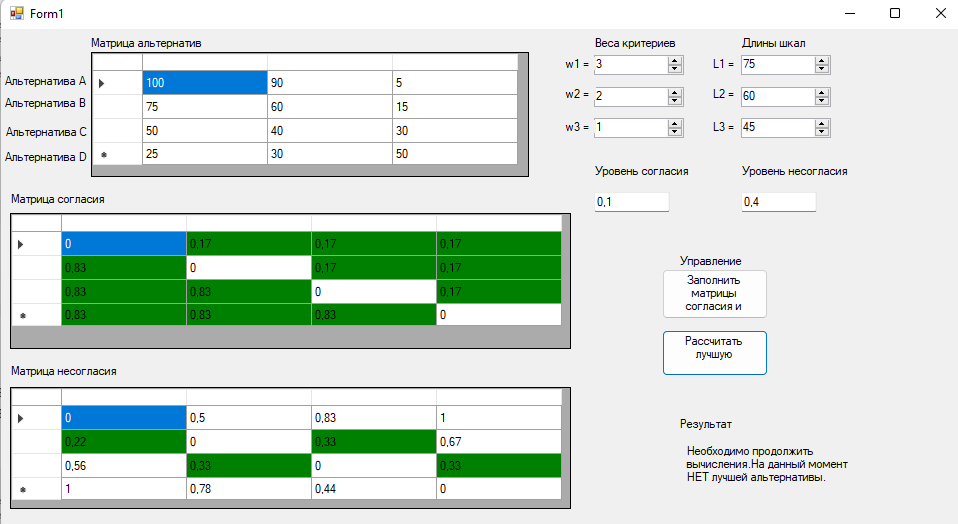


Рисунок 4 – Результат расчета наилучшей альтернативы

**Вывод:** был реализован метод ELECTRE I для своей предметной области на ЯВУ.