МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

**Лабораторная работа №2**

**Решение уравнений симплекс методом**

Выполнил студент группы ИВТ-31 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кудяшев Я.Ю./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Коржавина А.С./

Киров 2021

1. Цель работы: написать программу, решающую систему уравнений симплекс-методом.
2. Экранные формы

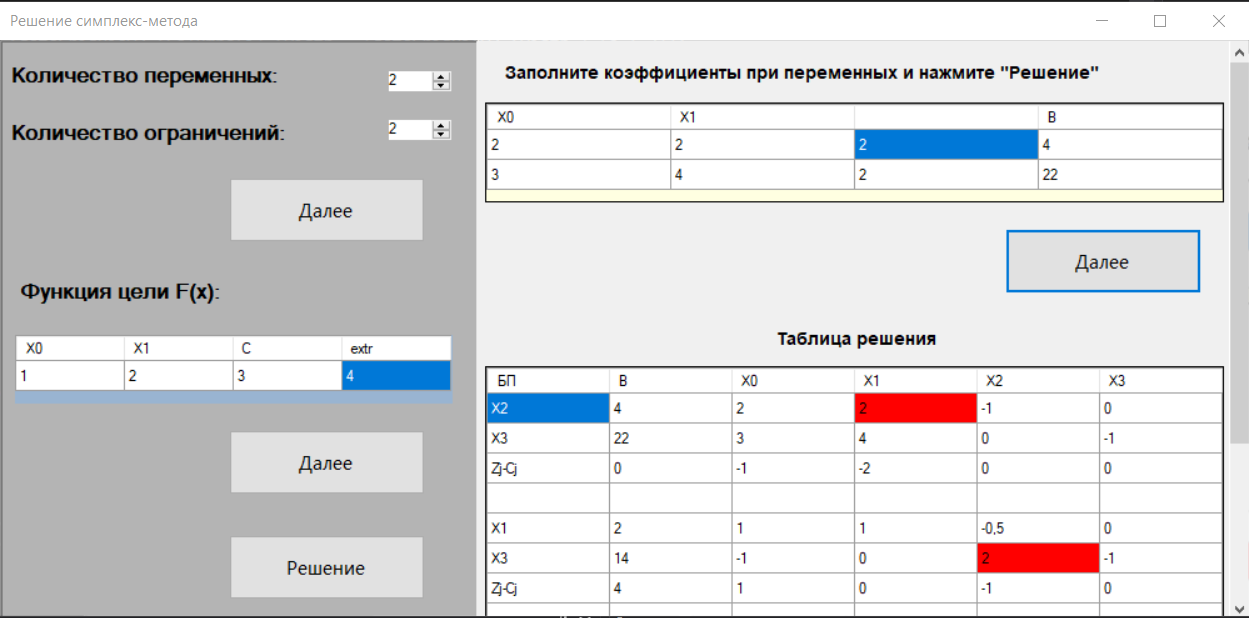


Рисунок 1 – Пример работы программы (1)

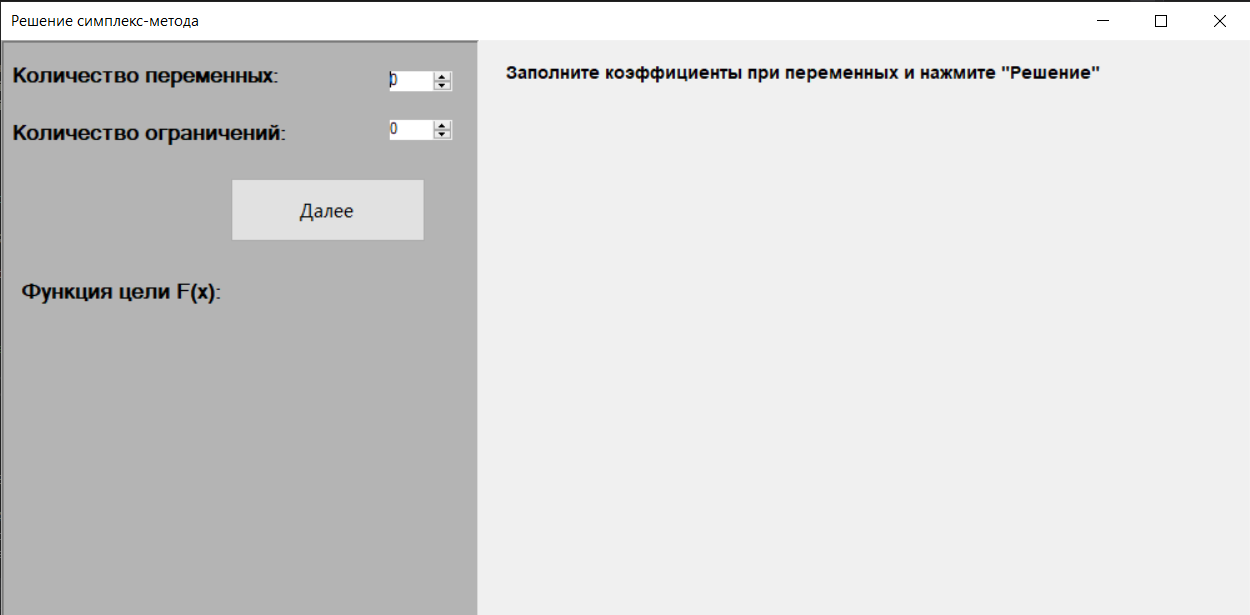


Рисунок 2 – Пример работы программы (2)

1. Программный код

#pragma once

#include <cmath>

namespace Simplex {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

int a, b;

double func\_celi[10];

double c;

double table[10][20];

double dop\_table[10][20];

double table\_smesh[10][20];

double B[10];

double B\_smesh[10];

double dop\_B[10];

char znak[10];

double delta[20];

double delta\_smesh[20];

double delta\_b;

double delta\_b\_smesh;

int BP[10];

int row\_counter;

int max\_row;

int max\_cell;

double kek;

int smesh\_counter = 0;

bool ukazatel = false;

bool ukazatel2 = false;

/// <summary>

/// Сводка для MyForm

/// </summary>

public ref class MyForm : public System::Windows::Forms::Form

{

public:

MyForm(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Label^ label1;

protected:

private: System::Windows::Forms::NumericUpDown^ numericUpDown1;

private: System::Windows::Forms::NumericUpDown^ numericUpDown2;

private: System::Windows::Forms::Panel^ panel1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::DataGridView^ dataGridView1;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::DataGridView^ dataGridView2;

private: System::Windows::Forms::Button^ button3;

private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::Panel^ panel2;

private: System::Windows::Forms::Button^ button4;

private: System::Windows::Forms::Label^ label5;

private: System::Windows::Forms::DataGridView^ dataGridView3;

private: System::Windows::Forms::Button^ button5;

private: System::Windows::Forms::Label^ label6;

private: System::Windows::Forms::DataGridView^ dataGridView4;

protected:

protected:

protected:

protected:

protected:

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

///

///

void InitializeComponent(void)

{

this->label1 = (gcnew System::Windows::Forms::Label());

this->numericUpDown1 = (gcnew System::Windows::Forms::NumericUpDown());

this->numericUpDown2 = (gcnew System::Windows::Forms::NumericUpDown());

this->panel1 = (gcnew System::Windows::Forms::Panel());

this->button3 = (gcnew System::Windows::Forms::Button());

this->button2 = (gcnew System::Windows::Forms::Button());

this->label3 = (gcnew System::Windows::Forms::Label());

this->dataGridView1 = (gcnew System::Windows::Forms::DataGridView());

this->button1 = (gcnew System::Windows::Forms::Button());

this->label2 = (gcnew System::Windows::Forms::Label());

this->dataGridView2 = (gcnew System::Windows::Forms::DataGridView());

this->label4 = (gcnew System::Windows::Forms::Label());

this->panel2 = (gcnew System::Windows::Forms::Panel());

this->label6 = (gcnew System::Windows::Forms::Label());

this->button5 = (gcnew System::Windows::Forms::Button());

this->label5 = (gcnew System::Windows::Forms::Label());

this->dataGridView3 = (gcnew System::Windows::Forms::DataGridView());

this->button4 = (gcnew System::Windows::Forms::Button());

this->dataGridView4 = (gcnew System::Windows::Forms::DataGridView());

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown1))->BeginInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown2))->BeginInit();

this->panel1->SuspendLayout();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView1))->BeginInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView2))->BeginInit();

this->panel2->SuspendLayout();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView3))->BeginInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView4))->BeginInit();

this->SuspendLayout();

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(3, 16);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(274, 25);

this->label1->TabIndex = 0;

this->label1->Text = L"Количество переменных:";

this->label1->Click += gcnew System::EventHandler(this, &MyForm::label1\_Click);

//

// numericUpDown1

//

this->numericUpDown1->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->numericUpDown1->Location = System::Drawing::Point(309, 23);

this->numericUpDown1->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->numericUpDown1->Name = L"numericUpDown1";

this->numericUpDown1->Size = System::Drawing::Size(50, 18);

this->numericUpDown1->TabIndex = 2;

this->numericUpDown1->ValueChanged += gcnew System::EventHandler(this, &MyForm::numericUpDown1\_ValueChanged);

//

// numericUpDown2

//

this->numericUpDown2->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->numericUpDown2->Location = System::Drawing::Point(309, 62);

this->numericUpDown2->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->numericUpDown2->Name = L"numericUpDown2";

this->numericUpDown2->Size = System::Drawing::Size(50, 18);

this->numericUpDown2->TabIndex = 3;

this->numericUpDown2->ValueChanged += gcnew System::EventHandler(this, &MyForm::numericUpDown2\_ValueChanged);

//

// panel1

//

this->panel1->BackColor = System::Drawing::SystemColors::ActiveBorder;

this->panel1->BorderStyle = System::Windows::Forms::BorderStyle::Fixed3D;

this->panel1->Controls->Add(this->button3);

this->panel1->Controls->Add(this->button2);

this->panel1->Controls->Add(this->label3);

this->panel1->Controls->Add(this->dataGridView1);

this->panel1->Controls->Add(this->button1);

this->panel1->Controls->Add(this->label2);

this->panel1->Controls->Add(this->numericUpDown2);

this->panel1->Controls->Add(this->label1);

this->panel1->Controls->Add(this->numericUpDown1);

this->panel1->Location = System::Drawing::Point(0, 0);

this->panel1->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->panel1->Name = L"panel1";

this->panel1->Size = System::Drawing::Size(383, 464);

this->panel1->TabIndex = 4;

this->panel1->Paint += gcnew System::Windows::Forms::PaintEventHandler(this, &MyForm::panel1\_Paint);

//

// button3

//

this->button3->Font = (gcnew System::Drawing::Font(L"Microsoft YaHei", 10.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button3->Location = System::Drawing::Point(181, 394);

this->button3->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->button3->Name = L"button3";

this->button3->Size = System::Drawing::Size(157, 52);

this->button3->TabIndex = 9;

this->button3->Text = L"Решение";

this->button3->UseVisualStyleBackColor = true;

this->button3->Visible = false;

this->button3->Click += gcnew System::EventHandler(this, &MyForm::button3\_Click);

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Microsoft YaHei", 10.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button2->Location = System::Drawing::Point(181, 310);

this->button2->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(157, 52);

this->button2->TabIndex = 8;

this->button2->Text = L"Далее";

this->button2->UseVisualStyleBackColor = true;

this->button2->Visible = false;

this->button2->Click += gcnew System::EventHandler(this, &MyForm::button2\_Click);

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(10, 189);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(211, 25);

this->label3->TabIndex = 7;

this->label3->Text = L"Функция цели F(x):";

//

// dataGridView1

//

this->dataGridView1->AllowUserToAddRows = false;

this->dataGridView1->AllowUserToDeleteRows = false;

this->dataGridView1->BackgroundColor = System::Drawing::SystemColors::ActiveCaption;

this->dataGridView1->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->dataGridView1->ColumnHeadersHeightSizeMode = System::Windows::Forms::DataGridViewColumnHeadersHeightSizeMode::AutoSize;

this->dataGridView1->Location = System::Drawing::Point(10, 234);

this->dataGridView1->Name = L"dataGridView1";

this->dataGridView1->RowHeadersVisible = false;

this->dataGridView1->RowHeadersWidth = 51;

this->dataGridView1->RowTemplate->Height = 24;

this->dataGridView1->ScrollBars = System::Windows::Forms::ScrollBars::None;

this->dataGridView1->ShowCellErrors = false;

this->dataGridView1->ShowCellToolTips = false;

this->dataGridView1->ShowEditingIcon = false;

this->dataGridView1->ShowRowErrors = false;

this->dataGridView1->Size = System::Drawing::Size(350, 55);

this->dataGridView1->TabIndex = 6;

this->dataGridView1->Visible = false;

this->dataGridView1->CellContentClick += gcnew System::Windows::Forms::DataGridViewCellEventHandler(this, &MyForm::dataGridView1\_CellContentClick);

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Microsoft YaHei", 10.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->Location = System::Drawing::Point(181, 108);

this->button1->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(157, 52);

this->button1->TabIndex = 5;

this->button1->Text = L"Далее";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm::button1\_Click\_1);

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(3, 62);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(274, 25);

this->label2->TabIndex = 4;

this->label2->Text = L"Количество ограничений:";

this->label2->Click += gcnew System::EventHandler(this, &MyForm::label2\_Click);

//

// dataGridView2

//

this->dataGridView2->AllowUserToAddRows = false;

this->dataGridView2->AllowUserToDeleteRows = false;

this->dataGridView2->BackgroundColor = System::Drawing::SystemColors::Info;

this->dataGridView2->ColumnHeadersHeightSizeMode = System::Windows::Forms::DataGridViewColumnHeadersHeightSizeMode::AutoSize;

this->dataGridView2->Location = System::Drawing::Point(7, 49);

this->dataGridView2->Name = L"dataGridView2";

this->dataGridView2->RowHeadersVisible = false;

this->dataGridView2->RowHeadersWidth = 51;

this->dataGridView2->RowTemplate->Height = 24;

this->dataGridView2->ScrollBars = System::Windows::Forms::ScrollBars::None;

this->dataGridView2->ShowCellErrors = false;

this->dataGridView2->ShowCellToolTips = false;

this->dataGridView2->ShowEditingIcon = false;

this->dataGridView2->ShowRowErrors = false;

this->dataGridView2->Size = System::Drawing::Size(591, 30);

this->dataGridView2->TabIndex = 5;

this->dataGridView2->Visible = false;

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Microsoft Himalaya", 10, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(20, 17);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(560, 19);

this->label4->TabIndex = 9;

this->label4->Text = L"Заполните коэффициенты при переменных и нажмите \"Решение\"";

//

// panel2

//

this->panel2->AutoScroll = true;

this->panel2->Controls->Add(this->dataGridView4);

this->panel2->Controls->Add(this->label6);

this->panel2->Controls->Add(this->button5);

this->panel2->Controls->Add(this->label5);

this->panel2->Controls->Add(this->dataGridView3);

this->panel2->Controls->Add(this->button4);

this->panel2->Controls->Add(this->dataGridView2);

this->panel2->Controls->Add(this->label4);

this->panel2->Location = System::Drawing::Point(381, 1);

this->panel2->Name = L"panel2";

this->panel2->Size = System::Drawing::Size(619, 463);

this->panel2->TabIndex = 10;

//

// label6

//

this->label6->AutoSize = true;

this->label6->Font = (gcnew System::Drawing::Font(L"Microsoft Himalaya", 10, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label6->Location = System::Drawing::Point(237, 311);

this->label6->Name = L"label6";

this->label6->Size = System::Drawing::Size(169, 19);

this->label6->TabIndex = 14;

this->label6->Text = L"Таблица отсечений";

this->label6->Visible = false;

//

// button5

//

this->button5->Font = (gcnew System::Drawing::Font(L"Microsoft YaHei", 10.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button5->Location = System::Drawing::Point(404, 385);

this->button5->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->button5->Name = L"button5";

this->button5->Size = System::Drawing::Size(176, 52);

this->button5->TabIndex = 13;

this->button5->Text = L"Метод отсечений";

this->button5->UseVisualStyleBackColor = true;

this->button5->Visible = false;

this->button5->Click += gcnew System::EventHandler(this, &MyForm::button5\_Click);

//

// label5

//

this->label5->AutoSize = true;

this->label5->Font = (gcnew System::Drawing::Font(L"Microsoft Himalaya", 10, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label5->Location = System::Drawing::Point(237, 235);

this->label5->Name = L"label5";

this->label5->Size = System::Drawing::Size(156, 19);

this->label5->TabIndex = 12;

this->label5->Text = L"Таблица решения";

this->label5->Visible = false;

this->label5->Click += gcnew System::EventHandler(this, &MyForm::label5\_Click);

//

// dataGridView3

//

this->dataGridView3->AllowUserToAddRows = false;

this->dataGridView3->AllowUserToDeleteRows = false;

this->dataGridView3->BackgroundColor = System::Drawing::SystemColors::Info;

this->dataGridView3->ColumnHeadersHeightSizeMode = System::Windows::Forms::DataGridViewColumnHeadersHeightSizeMode::AutoSize;

this->dataGridView3->Location = System::Drawing::Point(7, 260);

this->dataGridView3->Name = L"dataGridView3";

this->dataGridView3->RowHeadersVisible = false;

this->dataGridView3->RowHeadersWidth = 51;

this->dataGridView3->RowTemplate->Height = 24;

this->dataGridView3->ScrollBars = System::Windows::Forms::ScrollBars::None;

this->dataGridView3->Size = System::Drawing::Size(591, 30);

this->dataGridView3->TabIndex = 11;

this->dataGridView3->Visible = false;

//

// button4

//

this->button4->Font = (gcnew System::Drawing::Font(L"Microsoft YaHei", 10.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button4->Location = System::Drawing::Point(423, 109);

this->button4->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->button4->Name = L"button4";

this->button4->Size = System::Drawing::Size(157, 52);

this->button4->TabIndex = 10;

this->button4->Text = L"Далее";

this->button4->UseVisualStyleBackColor = true;

this->button4->Visible = false;

this->button4->Click += gcnew System::EventHandler(this, &MyForm::button4\_Click);

//

// dataGridView4

//

this->dataGridView4->AllowUserToAddRows = false;

this->dataGridView4->AllowUserToDeleteRows = false;

this->dataGridView4->BackgroundColor = System::Drawing::SystemColors::Info;

this->dataGridView4->ColumnHeadersHeightSizeMode = System::Windows::Forms::DataGridViewColumnHeadersHeightSizeMode::AutoSize;

this->dataGridView4->Location = System::Drawing::Point(7, 333);

this->dataGridView4->Name = L"dataGridView4";

this->dataGridView4->RowHeadersVisible = false;

this->dataGridView4->RowHeadersWidth = 51;

this->dataGridView4->RowTemplate->Height = 24;

this->dataGridView4->ScrollBars = System::Windows::Forms::ScrollBars::None;

this->dataGridView4->Size = System::Drawing::Size(591, 30);

this->dataGridView4->TabIndex = 15;

this->dataGridView4->Visible = false;

//

// MyForm

//

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Inherit;

this->BackgroundImageLayout = System::Windows::Forms::ImageLayout::Center;

this->ClientSize = System::Drawing::Size(999, 461);

this->Controls->Add(this->panel2);

this->Controls->Add(this->panel1);

this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedDialog;

this->Margin = System::Windows::Forms::Padding(3, 4, 3, 4);

this->Name = L"MyForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"Решение симплекс-метода";

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown1))->EndInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown2))->EndInit();

this->panel1->ResumeLayout(false);

this->panel1->PerformLayout();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView1))->EndInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView2))->EndInit();

this->panel2->ResumeLayout(false);

this->panel2->PerformLayout();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView3))->EndInit();

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->dataGridView4))->EndInit();

this->ResumeLayout(false);

}

#pragma endregion

void tablica(int xx,int bb) { //xx-количество добавочных столбцов bb смещение таблицы

if (ukazatel == false) {

this->dataGridView4->Columns->Add("", "БП");

this->dataGridView4->Columns[0]->Width = (dataGridView4->Width / (a + b + b + 2));

this->dataGridView4->Columns->Add("", "B");

this->dataGridView4->Columns[1]->Width = (dataGridView4->Width / (a + b + b + 2));

ukazatel = true;

}

for (int i = 2; i < (a + b + 2 + xx); i++) { //Заполнение основной таблицы

/\* if (i == 0) {

this->dataGridView4->Columns->Add("", "БП");

this->dataGridView4->Columns[i]->Width = (dataGridView4->Width / (a + b+xx + 2));

}

else if (i == 1) {

this->dataGridView4->Columns->Add("", "B");

this->dataGridView4->Columns[i]->Width = (dataGridView4->Width / (a + b+xx + 2));

} \*/

//else {

this->dataGridView4->Columns->Add("", "X" + (i - 2));

this->dataGridView4->Columns[i]->Width = (dataGridView4->Width / (a + b+b + 2));

//}

}

for (int i = 0; i < b+xx+5; i++) { //Добавление строк в таблицу +1

this->dataGridView4->Rows->Add("", "");

this->dataGridView4->Height = (dataGridView4->Height + 25);

}

BP[b-1+xx] = b+ a + xx-1;

for (int i = 0; i < b+xx+1; i++) { //Заполняем строчку X

dataGridView4[0, i+bb]->Value = "X" + (BP[i]);

if (i == b + xx) {

dataGridView4[0, i+bb]->Value = "F ( x )";

}

}

for (int i = 0; i <= b + xx-1; i++) { //заполнение столбца дополнительного 0 и одной 1

table\_smesh[i][b+a+xx-1] = 0;

if (i == b + xx-1 ) {

table\_smesh[i][b+a+xx-1] = 1;

}

}

for (int i = 0; i < b+a + xx - 1; i++) { //заполнение дополнительной строки ТУТ КОРОЧЕ НАДО НЕ ПОПОРЯДКУ ДЕЛАТЬ А ИСКАТЬ СТРОЧКУ С ДРОБНЫМИ ЧИСЛАМИ

if (ukazatel2 == false) {

table\_smesh[b + xx - 1][i] = (-1) \* modf(table[xx - 1][i], &kek);

}

else {

if (table\_smesh[xx - 1][i]<0) //&& table\_smesh[xx - 1][i]>-1)

table\_smesh[b + xx - 1][i] = 1- ((-1) \*( modf(table\_smesh[xx - 1][i], &kek) + 1)); else

table\_smesh[b + xx - 1][i] = 1 - ((-1) \* modf(table\_smesh[xx - 1][i], &kek));

}

}

if (ukazatel2 == false){

B\_smesh[b + xx - 1] = (-1) \* modf(B[xx - 1], &kek); }//заполнение нового B

else

{

B\_smesh[b + xx - 1] = (-1) \* modf(B\_smesh[xx - 1], &kek);

}

ukazatel2 = true;

for (int i = 0; i < b+xx; i++) { //Заполняем строку B

dataGridView4[1, i+bb]->Value = B\_smesh[i];

}

for (int i = 0; i < b+xx; i++) { //Заполнение основной таблицы

for (int j = 0; j < a + b+xx; j++) {

dataGridView4[j + 2, i+bb]->Value = table\_smesh[i][j];

}

}

for (int i = 0; i < b+a + xx; i++) { //Заполняем строку строку дельта

dataGridView4[i+2, b+xx+bb]->Value = delta\_smesh[i];

}

dataGridView4[1, b + xx+bb]->Value = delta\_b\_smesh;

}

void otsechenie(int xx, int bb) {

double max = abs(table[0][0]);//abs //поиск наибольшего элемента отрицательного по модулю

max\_row = 0;

max\_cell = 0;

for (int i = 0; i < b + xx;i++) {

for (int j = 0; j < a+b + xx; j++) {

if (table\_smesh[i][j] < 0) {

max = table\_smesh[i][j];

max\_row = i;

}

}

}

double min;// = abs(1.0 \* B[0] / table[0][max\_cell]);

int aa = 0;

bool proverka = false;

do {

if (table\_smesh[max\_row][aa] < 0) {

min = (delta\_smesh[aa] / table\_smesh[max\_row][aa]);

proverka = true;

max\_cell = aa;

}

else aa = aa + 1;

} while (proverka == false && aa < b+a+xx); //Тут нужно окно об ошибке

//Поиск ведщего элемента тут еше надо на минимальный проверить из-за минуса

for (int i = 0; i < b + a + xx; i++) {

if ((delta\_smesh[i] / table\_smesh[max\_row][i]) < min && (table\_smesh[max\_row][i]) < 0) {

min = delta\_smesh[i] / table\_smesh[max\_row][i];

max\_cell = i;

}

}

//dataGridView3[0, 1]->Value = max\_row;

dataGridView4->Rows[max\_row+bb-6]->Cells[max\_cell+2]->Style->BackColor = BackColor.Red; //тут еще роукаунтер

for (int i = 0; i < b + xx + 3; i++) { //Добавление строк в таблицу

this->dataGridView4->Rows->Add("", "");

this->dataGridView4->Height = (dataGridView4->Height + 25);

}

BP[max\_row] = max\_cell;

for (int i = 0; i < b + xx + 1; i++) { //Заполняем строчку X

dataGridView4[0, i + bb]->Value = "X" + (BP[i]);

if (i == b + xx) {

dataGridView4[0, i + bb]->Value = "F ( x )";

}

}

for (int i = 0; i < b+xx; i++) { //Рассчёт новго B

if (i == max\_row) {

dop\_B[i] = B\_smesh[i] / table\_smesh[max\_row][max\_cell];

}

else

if (i < max\_row) {

dop\_B[i] = B\_smesh[i] - B\_smesh[max\_row] \* table\_smesh[i][max\_cell] / table\_smesh[max\_row][max\_cell];

}

else

if (i > max\_row) {

dop\_B[i] = B\_smesh[i] - B\_smesh[max\_row] \* table\_smesh[i][max\_cell] / table\_smesh[max\_row][max\_cell];

}

}

for (int i = 0; i < b+xx; i++) {

B\_smesh[i] = dop\_B[i];

}

for (int i = 0; i < b+xx; i++) { //запись В в таблицу

dataGridView4[1, i+bb]->Value = B\_smesh[i];

}

double max\_x, max\_y; //Рассчёт всей новой таблицы

double maximum = table\_smesh[max\_row][max\_cell];

for (int i = 0; i < b+xx; i++) {

for (int j = 0; j < (a + b+xx); j++) {

if (i == max\_row) {

dop\_table[i][j] = table\_smesh[i][j] / maximum;

}

else

if (j == max\_cell) {

dop\_table[i][max\_cell] = 0;

}

else {

max\_x = table\_smesh[max\_row][j] \* 1.0;

max\_y = table\_smesh[i][max\_cell] \* 1.0;

dop\_table[i][j] = table\_smesh[i][j] - (max\_x \* max\_y) / maximum;

}

}

}

for (int i = 0; i < b+xx; i++) { //Перезапись из доп. массива в основной

for (int j = 0; j < (a + b+xx); j++) {

table\_smesh[i][j] = dop\_table[i][j];

}

}

for (int i = 0; i < b+xx; i++) { //Вывод на экран новой таблицы

for (int j = 0; j < a + b+xx; j++) {

dataGridView4[j + 2, i+bb]->Value = table\_smesh[i][j];

}

}

delta\_b\_smesh = (delta\_b\_smesh - ((delta\_smesh[max\_cell] \* B\_smesh[b+xx-1]) / table\_smesh[max\_row][max\_cell])); //рассчёт дельта В

dataGridView4[1, b + xx+bb]->Value = delta\_b\_smesh;

for (int i = 0; i < 20; i++) { //рассчёт дельта х

delta\_smesh[i] = 0;

};

for (int i = 0; i < a + b+xx; i++) {

for (int j = 0; j < b+xx; j++) {

delta\_smesh[i] = table\_smesh[j][i] \* func\_celi[BP[j]] + delta\_smesh[i];

}

delta\_smesh[i] = (delta\_smesh[i] - func\_celi[i])\* (-1);

dataGridView4[i + 2, b + xx+bb]->Value = delta\_smesh[i];

}

}

void delta\_B() //Высчет дельта В

{

delta\_b = 0;

for (int i = 0; i < b; i++) {

delta\_b = B[i] \* func\_celi[BP[i]] + delta\_b;

}

this->dataGridView3->Rows->Add("", "");

this->dataGridView3->Height = (dataGridView3->Height + 25);

dataGridView3[0, b+row\_counter]->Value = "Zj-Cj";

dataGridView3[1, b+row\_counter]->Value = delta\_b;

};

void delta\_X()

{

for (int i = 0; i < 20; i++) {

delta[i] = 0;

};

for (int i = 0; i < a + b; i++) {

for (int j = 0; j < b; j++) {

delta[i] = table[j][i] \* func\_celi[BP[j]] + delta[i];

}

delta[i] = delta[i] - func\_celi[i];

dataGridView3[i + 2, b+row\_counter]->Value = delta[i];

}

}

void osnova() {

for (int i = 0; i < b+1; i++) { //Добавление строк в таблицу

this->dataGridView3->Rows->Add("", "");

this->dataGridView3->Height = (dataGridView3->Height + 25);

}

double max = abs(table[0][0]);//abs

max\_row = 0;

max\_cell = 0;

for (int i = 0; i < b; i++) { //Поиск наибольшего элемента в таблице

for (int j = 0; j < a+b; j++) {

if (abs(table[i][j]) > max) { //abs

//max\_row = i;

max = table[i][j];

max\_cell = j;

}

}

}

double min;// = abs(1.0 \* B[0] / table[0][max\_cell]);

int aa = 0;

bool proverka = false;

do {

if (table[aa][max\_cell] > 0) {

min = (1.0 \* B[aa] / table[aa][max\_cell]);

proverka = true;

max\_row = aa;

}

else aa = aa + 1;

} while (proverka == false && aa<b); //Тут нужно окно об ошибке

//Поиск ведщего элемента тут еше надо на минимальный проверить из-за минуса

for (int i = 0; i < b; i++) {

if ((B[i]/table[i][max\_cell]) < min && (table[i][max\_cell])>0) {

min = B[i]/table[i][max\_cell];

max\_row = i;

}

}

//dataGridView3[0, 1]->Value = max\_row;

dataGridView3->Rows[max\_row+row\_counter]->Cells[max\_cell+2]->Style->BackColor = BackColor.Red;

BP[max\_row] = max\_cell;

row\_counter = row\_counter + b + 2;

for (int i = row\_counter; i < row\_counter+b; i++) { //Заполняем строчку X

dataGridView3[0, i]->Value = "X" + (BP[i-row\_counter]);

}

for (int i = 0; i < b; i++) { //Рассчёт новго B

if (i == max\_row) {

dop\_B[i] = B[i] / table[max\_row][max\_cell];

} else

if (i < max\_row) {

dop\_B[i] = B[i] - B[max\_row] \* table[i][max\_cell] / table[max\_row][max\_cell];

} else

if (i > max\_row) {

dop\_B[i] = B[i] - B[max\_row] \* table[i][max\_cell] / table[max\_row][max\_cell];

}

}

for (int i = 0; i < b; i++) {

B[i] = dop\_B[i];

}

for (int i = row\_counter; i < row\_counter+b; i++) { //запись В в таблицу

dataGridView3[1, i]->Value = B[i-row\_counter];

}

double max\_x, max\_y; //Рассчёт всей новой таблицы

double maximum = table[max\_row][max\_cell];

for (int i = 0; i < b; i++){

for (int j = 0; j < (a + b); j++) {

if (i == max\_row) {

dop\_table[i][j] = table[i][j] / maximum;

}

else

if (j == max\_cell) {

dop\_table[i][max\_cell] = 0;

}

else {

max\_x = table[max\_row][j] \* 1.0;

max\_y = table[i][max\_cell] \* 1.0;

dop\_table[i][j] = table[i][j] - (max\_x \* max\_y) / maximum;

}

}

}

for (int i = 0; i < b; i++) { //Перезапись из доп. массива в основной

for (int j = 0; j < (a + b); j++) {

table[i][j] = dop\_table[i][j];

}

}

for (int i = row\_counter; i < row\_counter+b; i++) { //Вывод на экран новой таблицы

for (int j = 0; j < a + b; j++) {

dataGridView3[j + 2, i]->Value = table[i-row\_counter][j];

}

}

}

private: System::Void button1\_Click\_1(System::Object^ sender, System::EventArgs^ e) {

dataGridView1->Visible = true;

a = System::Convert::ToInt32(numericUpDown1->Text);

b = System::Convert::ToInt32(numericUpDown2->Text);

for (int i = 0; i < a+2; i++) {

if (i == a) {

this->dataGridView1->Columns->Add("", "C");

this->dataGridView1->Columns[i]->Width = (dataGridView1->Width / (a+2));

}

else

if (i == a + 1) {

this->dataGridView1->Columns->Add("", "extr");

this->dataGridView1->Columns[i]->Width = (dataGridView1->Width / (a+2));

}

else {

this->dataGridView1->Columns->Add("", "X" + i);

this->dataGridView1->Columns[i]->Width = (dataGridView1->Width / (a + 2));

}

}

this->dataGridView1->Rows->Add("", "");

button2->Visible = true;

}

private: System::Void label1\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void label2\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void numericUpDown1\_ValueChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void numericUpDown2\_ValueChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void panel1\_Paint(System::Object^ sender, System::Windows::Forms::PaintEventArgs^ e) {

}

private: System::Void dataGridView2\_CellContentClick(System::Object^ sender, System::Windows::Forms::DataGridViewCellEventArgs^ e) {

}

private: System::Void dataGridView1\_CellContentClick(System::Object^ sender, System::Windows::Forms::DataGridViewCellEventArgs^ e) {

}

private: System::Void button2\_Click(System::Object^ sender, System::EventArgs^ e) {

dataGridView2->Visible = true;

for (int i = 0; i < a + 2; i++) {

if (i == a) {

this->dataGridView2->Columns->Add("", "");

this->dataGridView2->Columns[i]->Width = (dataGridView2->Width / (a + 2));

}

else

if (i == a + 1) {

this->dataGridView2->Columns->Add("", "B");

this->dataGridView2->Columns[i]->Width = (dataGridView2->Width / (a + 2));

}

else {

this->dataGridView2->Columns->Add("", "X" + i);

this->dataGridView2->Columns[i]->Width = (dataGridView2->Width / (a + 2));

}

}

for (int i = 0; i < b; i++) {

this->dataGridView2->Rows->Add("", "");

this->dataGridView2->Height = (dataGridView2->Height + 25);

}

for (int i = 0; i < a+b; i++) { //Запись из функции в одномерный массив

if (i < a)

func\_celi[i] = Convert::ToDouble(dataGridView1->Rows[0]->Cells[i]->Value);

else

func\_celi[i] = 0;

}

c = Convert::ToDouble(dataGridView1->Rows[0]->Cells[a]->Value); // Значение С

button4->Top = b\*25+100;

//dataGridView2[0, 0]->Value = c;

button3->Visible = true;

button4->Visible = true;

}

private: System::Void button3\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void button4\_Click(System::Object^ sender, System::EventArgs^ e) {

label5->Top = button4->Top + 80;

dataGridView3->Top = label5->Top + 30;

dataGridView3->Visible = true;

label5->Visible = true;

for (int i = 0; i < b; i++) { //Заполнение массива числами из таблицы

for (int j = 0; j < a; j++) {

table[i][j] = Convert::ToDouble(dataGridView2->Rows[i]->Cells[j]->Value);

}

}

for (int i = 0; i < b; i++) { //Заполнение массива числами В

B[i] = Convert::ToDouble(dataGridView2->Rows[i]->Cells[a+1]->Value);

}

for (int i = 0; i < b; i++) { //Заполнение массива знаком больще меньше

znak[i] = Convert::ToChar(dataGridView2->Rows[i]->Cells[a]->Value);

}

for (int i = 0; i < (a+b + 2); i++) { //Заполнение основной таблицы

if (i == 0) {

this->dataGridView3->Columns->Add("", "БП");

this->dataGridView3->Columns[i]->Width = (dataGridView3->Width / (a+b + 2));

}

else if (i == 1) {

this->dataGridView3->Columns->Add("", "B");

this->dataGridView3->Columns[i]->Width = (dataGridView3->Width / (a+b + 2));

}

else {

this->dataGridView3->Columns->Add("", "X" + (i - 2));

this->dataGridView3->Columns[i]->Width = (dataGridView3->Width / (a+b + 2));

}

}

for (int i = 0; i < b; i++) { //Обнуление массива доп. чисел

for (int j = a; j < a+b; j++) {

table[i][j] = 0;

}

}

int j = a;

for (int i = 0; i < b; i++) { //Заполнеине доп чисел единицами

if (znak[i] == '<')

table[i][j] = 1;

else

table[i][j] = (-1);

j = j + 1;

}

for (int i = 0; i < b; i++) { //Добавление строк в таблицу

this->dataGridView3->Rows->Add("", "");

this->dataGridView3->Height = (dataGridView3->Height + 25);

}

for (int i = 0; i < b; i++) { //Заполняем строчку X

dataGridView3[0, i]->Value = "X"+(a+i);

}

for (int i = 0; i < b; i++) { //Заполняем строку B

dataGridView3[1, i]->Value = B[i];

}

for (int i = 0; i < b; i++) { //Заполнение основной таблицы

for (int j = 0; j < a+b; j++) {

dataGridView3[j+2,i]->Value = table[i][j];

}

}

for (int i = 0; i < b; i++) { //Заполнение коэффициентов BP

BP[i] = a + i;

}

delta\_B(); //Подсчет и вывод дельта В

delta\_X(); //Подсчёт и вывод дельта иксов

int gg = 0;

for (int i = 0; i < a; i++) {

gg = 0;

for (int j = 0; j < a + b; j++) {

if (delta[j] < 0)

gg = 1;

}

if (gg == 1) {

osnova();

delta\_B(); //Подсчет и вывод дельта В

delta\_X(); //Подсчёт и вывод дельта иксов

}

}

button5->Top = (dataGridView3->Location.Y + dataGridView3->Rows->Count \* 25)+50;

button5->Visible = true;

label6->Top = (dataGridView3->Location.Y + dataGridView3->Rows->Count \* 25) + 100;

label6->Visible = true;

int smesh = (dataGridView3->Location.Y + dataGridView3->Rows->Count \* 25) + 50; // Относительное смещение таблицы

dataGridView4->Top = smesh +100;

}

private: System::Void label5\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void button5\_Click(System::Object^ sender, System::EventArgs^ e) {

dataGridView4->Visible = true;

for (int i = 0; i < b; i++) { //Перезапись из доп. массива в основной

for (int j = 0; j < (a + b); j++) {

table\_smesh[i][j] = table[i][j];

}

}

for (int i = 0; i < b; i++) {

B\_smesh[i] = B[i];

}

for (int i = 0; i < a+b; i++) {

delta\_smesh[i] = -delta[i];

}

delta\_b\_smesh = -delta\_b;

int xx = 1;

int bb = b + 4 + xx - 1;

tablica(xx,0);

otsechenie(xx,b+4+xx-1);

tablica(2, bb+5);

otsechenie(2,bb+bb+5);

// tablica(3, bb + 5);

//otsechenie(2, bb + bb + 5);

}

};

1. Вывод

В ходе выполнения лабораторной работы были получены навыки в решении системы уравнений симплекс-методом. Была написана программа, позволяющая решить любую систему уравнений симплекс-методом.