

Лабораторна робота 2

Тема : Програмування лінійних алгоритмів

Мета роботи: набути практичних навиків створення програмних проєктів лінійної структури в Visual C++.

Хід роботи

Завдання до роботи

Створити проєкт для обчислення опору електричного кола, яке складається з двох поєднаних опорів, у два варіанти: 1) за паралельного з'єднання опорів та 2) за послідовного з'єднання опорів в електричному колі. Розв'язок. Нагадаємо, що формула для обчислення опору електричного

$r_1 \cdot r_2$ кола за паралельного з'єднання опорів має вигляд $R = \frac{r_1 \cdot r_2}{r_1 + r_2}$, а послідовного $r_1 + r_2$ з'єднання опорів – $R = r_1 + r_2$. Обчислення для двох випадків сформуємо в програмі у двох окремих командних кнопках на формі:

Лістинг 2.1 – Код програми

```
private: System::Void buttonParallel_Click(System::Object^ sender,
System::EventArgs^ e) {
if (textR1->Text == "" || textR2->Text == "") {
MessageBox::Show("Введіть обидва значення R1 і R2!", "Помилка",
MessageBoxButtons::OK, MessageBoxIcon::Error);
return; }
double r1 = Convert::ToDouble(textR1->Text); double
r2 = Convert::ToDouble(textR2->Text);
double R = (r1 * r2) / (r1 + r2);
labelResult->Text = "Результат: R (паралельно) = " + R.ToString("F3") + "
Ом"; }
private: System::Void buttonSeries_Click(System::Object^ sender,
System::EventArgs^ e) {
if (textR1->Text == "" || textR2->Text == "") {
MessageBox::Show("Введіть обидва значення R1 і R2!", "Помилка",
MessageBoxButtons::OK, MessageBoxIcon::Error);
return; }
double r1 = Convert::ToDouble(textR1->Text); double
r2 = Convert::ToDouble(textR2->Text); double R = r1
+ r2;
labelResult->Text = "Результат: R (послідовно) = " + R.ToString("F3") + "
Ом";
} };
```

1. Розглянути та виконати приклад реалізації візуальної форми
2. Створити програмний проект в середовищі Visual C++ та виконати індивідуальне завдання

5.	Тіло рухається за законом $S = t^3 - \sqrt{t}$. Обчислити швидкість тіла і відстань у момент часу t . Значення t ввести з екрана. (Функція швидкості є похідною від функції відстані)
----	--

Рисунок 2.1 – Варіант індивідуального завдання

Лістинг 2.1 – Код програми

```
#pragma once
#include <cmath>

namespace Lab2 {

    using namespace System;
    using namespace System::ComponentModel;    using namespace
System::Collections;    using namespace System::Windows::Forms;    using
namespace System::Data;    using namespace System::Drawing;

    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
        }
    protected:
        ~Form1()    {
if (components)
    {
        delete components;
    }
    private:
System::Windows::Forms::Label^ label1;    private:
System::Windows::Forms::Label^ labelS;    private:
System::Windows::Forms::Label^ labelV;    private:
System::Windows::Forms::TextBox^ textBoxT;    private:
System::Windows::Forms::TextBox^ textBoxS;    private:
System::Windows::Forms::TextBox^ textBoxV;    private:
System::Windows::Forms::Button^ buttonCalc;    private:
System::Windows::Forms::Button^ buttonClear;    private:
System::Windows::Forms::Button^ buttonExit;    private:
System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code    void
InitializeComponent(void)
    {
```

```
this->label1 = (gcnew System::Windows::Forms::Label());
this->labelS = (gcnew System::Windows::Forms::Label());
this->labelV = (gcnew System::Windows::Forms::Label());
this->textBoxT = (gcnew System::Windows::Forms::TextBox());
this->textBoxS = (gcnew System::Windows::Forms::TextBox());
this->textBoxV = (gcnew System::Windows::Forms::TextBox());
this->buttonCalc = (gcnew System::Windows::Forms::Button());
this->buttonClear = (gcnew System::Windows::Forms::Button());
this->buttonExit = (gcnew System::Windows::Forms::Button());
this->SuspendLayout();
// // label1 //
this->label1->Location = System::Drawing::Point(20, 20);
this->label1->Name = L"label1";
this->label1->Size = System::Drawing::Size(200, 23);
this->label1->TabIndex = 0;
this->label1->Text = L"Введіть час t ( $t \geq 0$ ):";
// // labelS //
this->labelS->Location = System::Drawing::Point(20, 70);
this->labelS->Name = L"labelS";
this->labelS->Size = System::Drawing::Size(100, 23);
this->labelS->TabIndex = 1; this->labelS->Text = L"Відстань S:";
// // labelV //
this->labelV->Location = System::Drawing::Point(20, 110);
this->labelV->Name = L"labelV";
this->labelV->Size = System::Drawing::Size(120, 23);
this->labelV->TabIndex = 2;
this->labelV->Text = L"Швидкість V(t):";
// // textBoxT //
this->textBoxT->Location = System::Drawing::Point(200, 20);
this->textBoxT->Name = L"textBoxT";
this->textBoxT->Size = System::Drawing::Size(120, 22);
this->textBoxT->TabIndex = 3;
// // textBoxS //
this->textBoxS->Location = System::Drawing::Point(200, 70);
this->textBoxS->Name = L"textBoxS"; this->
>textBoxS->ReadOnly = true;
this->textBoxS->Size = System::Drawing::Size(120, 22);
this->textBoxS->TabIndex = 4;
// // textBoxV //
this->textBoxV->Location = System::Drawing::Point(200,
110);
this->textBoxV->Name = L"textBoxV"; this->
>textBoxV->ReadOnly = true;
this->textBoxV->Size = System::Drawing::Size(120, 22);
this->textBoxV->TabIndex = 5;
// // buttonCalc //
this->buttonCalc->Location = System::Drawing::Point(20,
160);
this->buttonCalc->Name = L"buttonCalc";
```

```
        this->buttonCalc->Size = System::Drawing::Size(90, 30);
this->buttonCalc->TabIndex = 6;                this->buttonCalc->Text =
L"Обчислити";                this->buttonCalc->UseVisualStyleBackColor =
true;                this->buttonCalc->Click += gcnew
System::EventHandler(this,
&Form1::buttonCalc_Click);
        //                // buttonClear                //
        this->buttonClear->Location = System::Drawing::Point(140,
160);                this->buttonClear->Name = L"buttonClear";
        this->buttonClear->Size = System::Drawing::Size(90, 30);
this->buttonClear->TabIndex = 7;                this->buttonClear->Text =
L"Очистити";                this->buttonClear->UseVisualStyleBackColor =
true;

        this->buttonClear->Click += gcnew
System::EventHandler(this,
&Form1::buttonClear_Click);
        //                // buttonExit                //
        this->buttonExit->Location = System::Drawing::Point(260,
160);                this->buttonExit->Name = L"buttonExit";
        this->buttonExit->Size = System::Drawing::Size(60, 30);
this->buttonExit->TabIndex = 8;                this->buttonExit->Text =
L"Вихід";

        this->buttonExit->UseVisualStyleBackColor = true;
        this->buttonExit->Click += gcnew System::EventHandler(this,
&Form1::buttonExit_Click);
        //                // Form1                //
        this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(360, 220);
this->Controls->Add(this->label1);                this->Controls-
>Add(this->labelS);                this->Controls->Add(this->labelV);
this->Controls->Add(this->textBoxT);                this->Controls-
>Add(this->textBoxS);                this->Controls->Add(this->textBoxV);
this->Controls->Add(this->buttonCalc);                this->Controls-
>Add(this->buttonClear);                this->Controls->Add(this-
>buttonExit);                this->Name = L"Form1";
        this->Text = L"Pyx тіла:  $S = t^3 - \sqrt{t}$ ";
this->ResumeLayout(false);                this->PerformLayout();
    }
#pragma endregion
```

```
private: System::Void buttonCalc_Click(System::Object^ sender,
System::EventArgs^ e)
{
    try
    {
        double t = System::Convert::ToDouble(textBoxT->Text);

        if (t < 0.0)
        {
            MessageBox::Show("Час t має бути невід'ємним ( $t \geq 0$ ).",
```

```
"Помилка введення",
MessageBoxButtons::OK, MessageBoxIcon::Warning);
return;
}
double S = pow(t, 3) - sqrt(t);
double Vt;
if (t == 0.0) {
    MessageBox::Show("Швидкість в точці t = 0 не визначена
(ділення на нуль).
Виведено тільки S(0).", "Увага", MessageBoxButtons::OK,
MessageBoxIcon::Information);
Vt = System::Double::NaN;
}
else
{
    Vt = 3.0 * pow(t, 2) - 1.0 / (2.0 * sqrt(t));
}
textBoxS->Text = System::String::Format("{0:F4}", S);
if (System::Double::IsNaN(Vt))
    textBoxV->Text
= "не визначено";
else
    textBoxV->Text = System::String::Format("{0:F4}", Vt);
}
catch (FormatException^)
{
    MessageBox::Show("Неправильний формат введення. Введіть число
(наприклад 2.5).",
"Помилка", MessageBoxButtons::OK, MessageBoxIcon::Error);
}
catch (Exception^ ex)
{
    MessageBox::Show("Сталася помилка: " + ex->Message, "Помилка",
MessageBoxButtons::OK, MessageBoxIcon::Error);
}
}
private: System::Void buttonClear_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBoxT->Clear();
    textBoxS->Clear();
    textBoxV->Clear();
}
private: System::Void buttonExit_Click(System::Object^ sender,
System::EventArgs^ e)
{
    this->Close();
}
};
}
```

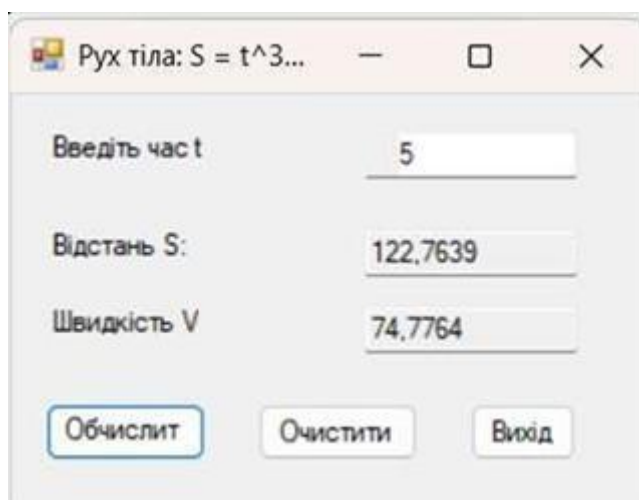


Рисунок 2.2 – Результат роботи програми

Висновок: На цій лабораторній роботі я набула практичних навиків створення програмних проєктів лінійної структури в Visual C++.

