

Лабораторна робота 3

Тема: Створення Windows додатків в середовищі MS Visual C++ із використанням додаткових елементів вводу/виводу даних

Мета роботи: формування навичок використання різних елементів управління у програмах Windows Forms

Хід роботи

Завдання для роботи

Приклад1 Заповнення таблиці

Розглянемо приклад створення Windows – додатку, який дозволяє створити матрицю заданої розмірності, ввести елементи матриці, або згенерувати їх за допомогою генератора випадкових чисел та підрахувати суму діагональних елементів матриці та кількість парних елементів. Для роботи із випадковими числами слід приєднати <random.h

Код функції для обробки події - натискання кнопки button1 матиме наступний вигляд:

Лістинг 3.1 Код програми

```
#pragma endregion
private: System::Void
button1_Click(System::Object^
System::EventArgs^
this->textBox3->Text = L"";
this->textBox4->Text = L"";
int m, n; try {
m = Convert::ToInt32(this->textBox1->Text); n = Convert::ToInt32(this-
>textBox2->Text);
if (m <= 0 || n <= 0) { MessageBox::Show(L"Розміри мають бути > 0.",
L"Помилка", MessageBoxButtons::OK, MessageBoxIcon::Error); return;
} }
catch (FormatException
ption^ ex) {
MessageBox::Show(L"Заповніть форму даними", L"Помилка вводу даних",
MessageBoxButtons::OK, MessageBoxIcon::Exclamation);
return;
}
dataGridView1->Columns->Clear(); dataGridView1->ColumnCount = n;
dataGridView1->RowCount = m;
```

```
int sum = 0;
int kil = 0;
for (int i = 0; i < m; i++)
{
    for (int j = 0; j < n; j++)
    {
        int value = randGen->Next(100) - 50;
        dataGridView1->Rows[i]->Cells[j]->Value = Convert::ToString(value);
        if (i == j) { sum += value;
        }
        if (value % 2 == 0) { kil++;
        } }
    }
this->textBox3->Text = Convert::ToString(sum);
this->textBox4->Text = Convert::ToString(kil);
MessageBox::Show(L"Генерацію та обчислення за Прикладом виконано
успішно.", L"Виконано", MessageBoxButtons::OK,
MessageBoxIcon::Information);
} };
}
```

1. Завдання 1. Розглянути і опрацювати теоретичний матеріал, виконати приклади. Відповідні скріни вікон навести у звіті
2. Завдання 2. Створити проект Windows Form для розв'язання задачі згідно варіанту.

5.	Задано матриця D (4,4). Якщо максимальний елемент матриці стоїть на головній діагоналі, то всі елементи головної діагоналі зробити рівними максимальному
----	--

Рисунок 3.1

Лістинг 3.2 – Код програми

```
#pragma once
#include <ctime>
#include <cstdlib>

namespace Lab3 {

    using namespace System;
    using namespace System::Windows::Forms;
    using namespace System::Drawing;
```

```
public ref class Form1 : public Form
{
public:
    Form1(void)
    {
        InitializeComponent();
    }

protected:
    ~Form1() { if (components) delete components; }

private:
    System::ComponentModel::Container^ components;
    DataGridView^ dataGridView1;
    Button^ buttonGenerate;
    Button^ buttonProcess;
    Button^ buttonExit;
    Random^ rnd;

#pragma region Windows Form Designer generated code
void InitializeComponent(void)
{
    this->dataGridView1 = gcnew DataGridView();
    this->buttonGenerate = gcnew Button();
    this->buttonProcess = gcnew Button();
    this->buttonExit = gcnew Button();
    this->SuspendLayout();

    // dataGridView1
    this->dataGridView1->Location = Point(20, 20);
    this->dataGridView1->Size = System::Drawing::Size(260,
180);

    this->dataGridView1->ColumnCount = 4;
    this->dataGridView1->RowCount = 4;
    for (int i = 0; i < 4; i++)
    {
        this->dataGridView1->Columns[i]->Width = 60;
        this->dataGridView1->Rows[i]->Height = 35;
    }

    // buttonGenerate
    this->buttonGenerate->Text = L"Згенерувати";
    this->buttonGenerate->Location = Point(300, 40);
    this->buttonGenerate->Click += gcnew EventHandler(this,
&Form1::buttonGenerate_Click);

    // buttonProcess
    this->buttonProcess->Text = L"Опрацювати";
```

```
        this->buttonProcess->Location = Point(300, 100);
        this->buttonProcess->Click += gcnew EventHandler(this,
&Form1::buttonProcess_Click);

        // buttonExit
        this->buttonExit->Text = L"Вихід";
        this->buttonExit->Location = Point(300, 160);
        this->buttonExit->Click += gcnew EventHandler(this,
&Form1::buttonExit_Click);

        // Form1
        this->ClientSize = System::Drawing::Size(420, 240);
        this->Controls->Add(this->dataGridView1);
        this->Controls->Add(this->buttonGenerate);
        this->Controls->Add(this->buttonProcess);
        this->Controls->Add(this->buttonExit);
        this->Text = L"Лабораторна %3 - Варіант 5";
        this->ResumeLayout(false);
    }
#pragma endregion

private:
    array<double, 2>^ D = gcnew array<double, 2>(4, 4);

    System::Void buttonGenerate_Click(System::Object^ sender,
System::EventArgs^ e)
    {
        Random^ rnd = gcnew Random();
        for (int i = 0; i < 4; i++)
        {
            for (int j = 0; j < 4; j++)
            {
                D[i, j] = rnd->Next(-10, 11);
                dataGridView1->Rows[i]->Cells[j]->Value = D[i, j];
            }
        }
    }

    System::Void buttonProcess_Click(System::Object^ sender,
System::EventArgs^ e)
    {
        double maxVal = D[0, 0];
        int i_max = 0, j_max = 0;

        for (int i = 0; i < 4; i++)
        {
            for (int j = 0; j < 4; j++)
            {
                if (D[i, j] > maxVal)
```

```
        {
            maxVal = D[i, j];
            i_max = i;
            j_max = j;
        }
    }

    if (i_max == j_max)
    {
        for (int i = 0; i < 4; i++)
        {
            D[i, i] = maxVal;
        }
        MessageBox::Show(L"Максимальний елемент на головній  
діагоналі. \n Діагональ змінено!", L"Результат", MessageBoxButtons::OK,  
MessageBoxIcon::Information);
    }

    for (int i = 0; i < 4; i++)
    {
        for (int j = 0; j < 4; j++)
        {
            dataGridView1->Rows[i]->Cells[j]->Value = D[i, j];
        }
    }
}

System::Void buttonExit_Click(System::Object^ sender,  
System::EventArgs^ e)
{
    Close();
}

};

}
```

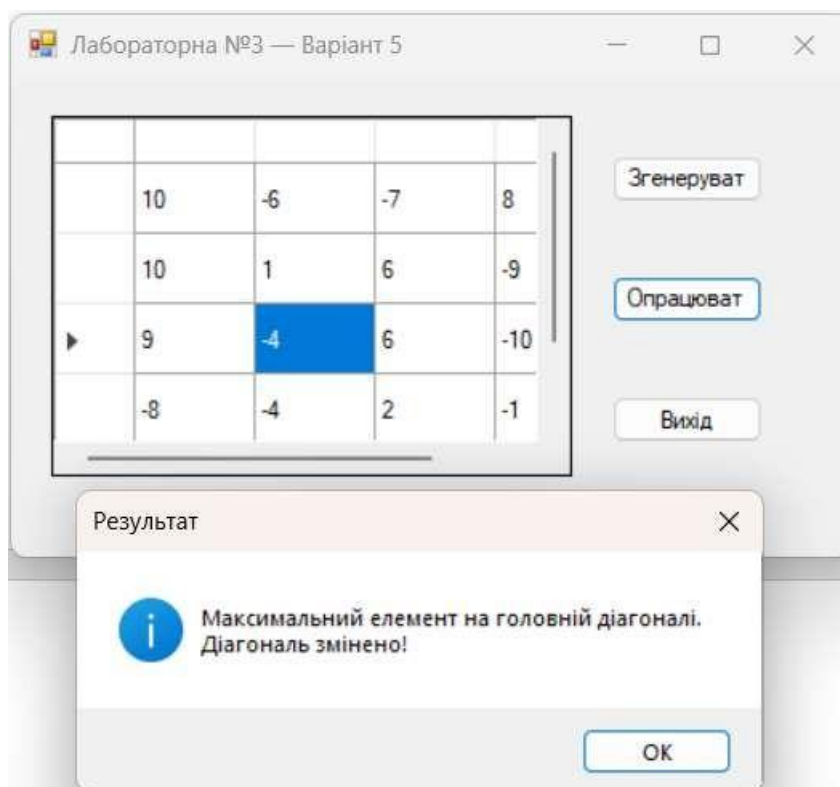


Рисунок 3.6 – Вікно результатів програми

Висновок: На цій лабораторній роботі я сформувала навички використання різних елементів управління у програмах Windows Forms