

Міністерство освіти і науки, молоді та спорту України
Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

з домашнє завдання №24.1 з дисципліни:

“Алгоритми та моделі обчислень”

Варіант: № 24.

Виконав:

ст. групи КІ-203

Ширий Богдан Ігорович

Перевірів:

ст. викладач кафедри ЕОМ

Козак Назар Богданович

ЗАВДАННЯ:

УМОВА:

Застосовуючи OpenCV скласти програму (C++), яка дозволяє виявити на фото обличчя людей. При цьому потрібно виявити не більше MAX_FACES_COUNT обличь.

ВИБІР ВАРІАНТУ:

$$(N_{\text{ж}} + N_{\text{г}} + 1) \% 10 + 1 = (24 + 3 + 1) \% 10 + 1 = 28 \% 10 + 1 = 9,$$

де: $N_{\text{ж}}$ – порядковий номер студента в групі, а $N_{\text{г}}$ – номер групи.

Отже, мій шуканий варіант – це максимум 9 облич, які можуть бути виявлені.

ВИКОНАННЯ:

Склав C++ десктопну програму та її роботу зобразив на рисунках від 1 до 7.

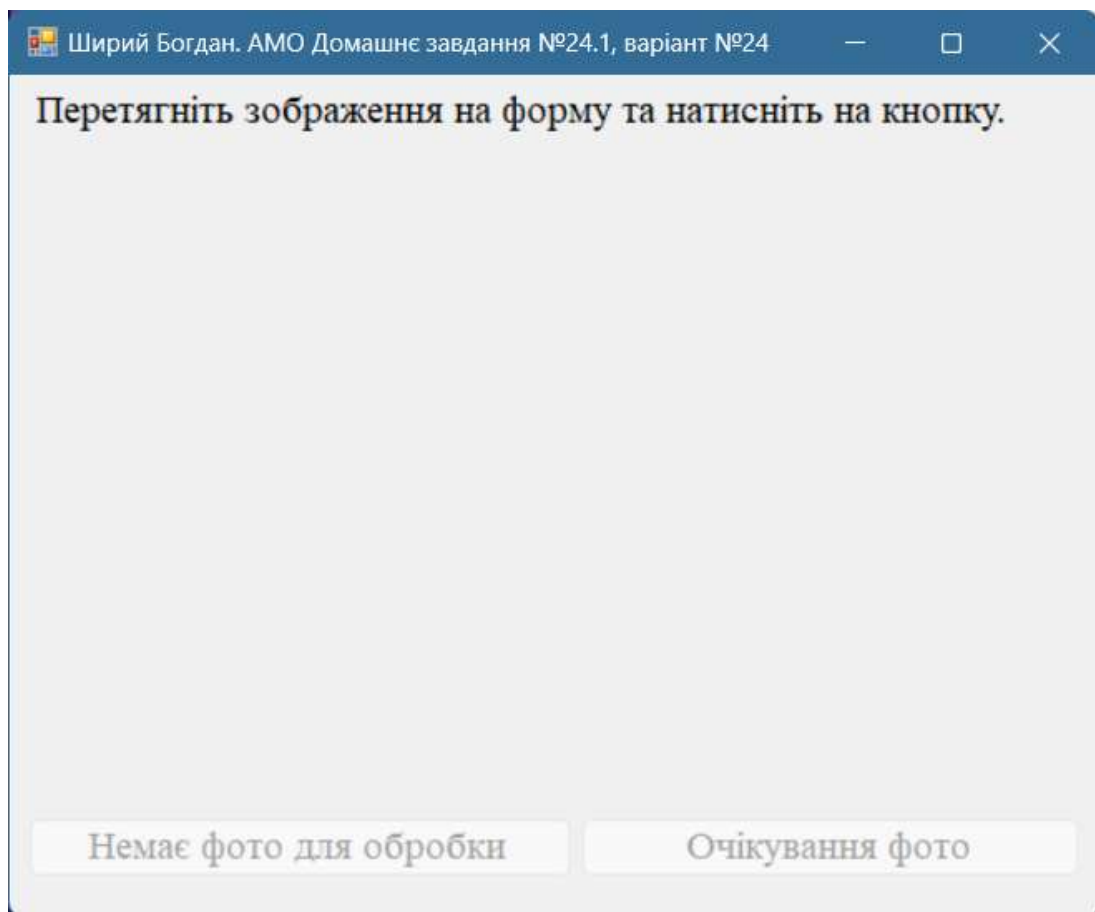


Рис. 1. Початковий вигляд програми.

Нижче наводжу обробку двох фото з обличчями, щоб довести, що програма знаходить не більше 9 облич, які знаходяться кожен раз випадково. У цій програмі також можна переключатися між останніми початковим та обробленим фото.

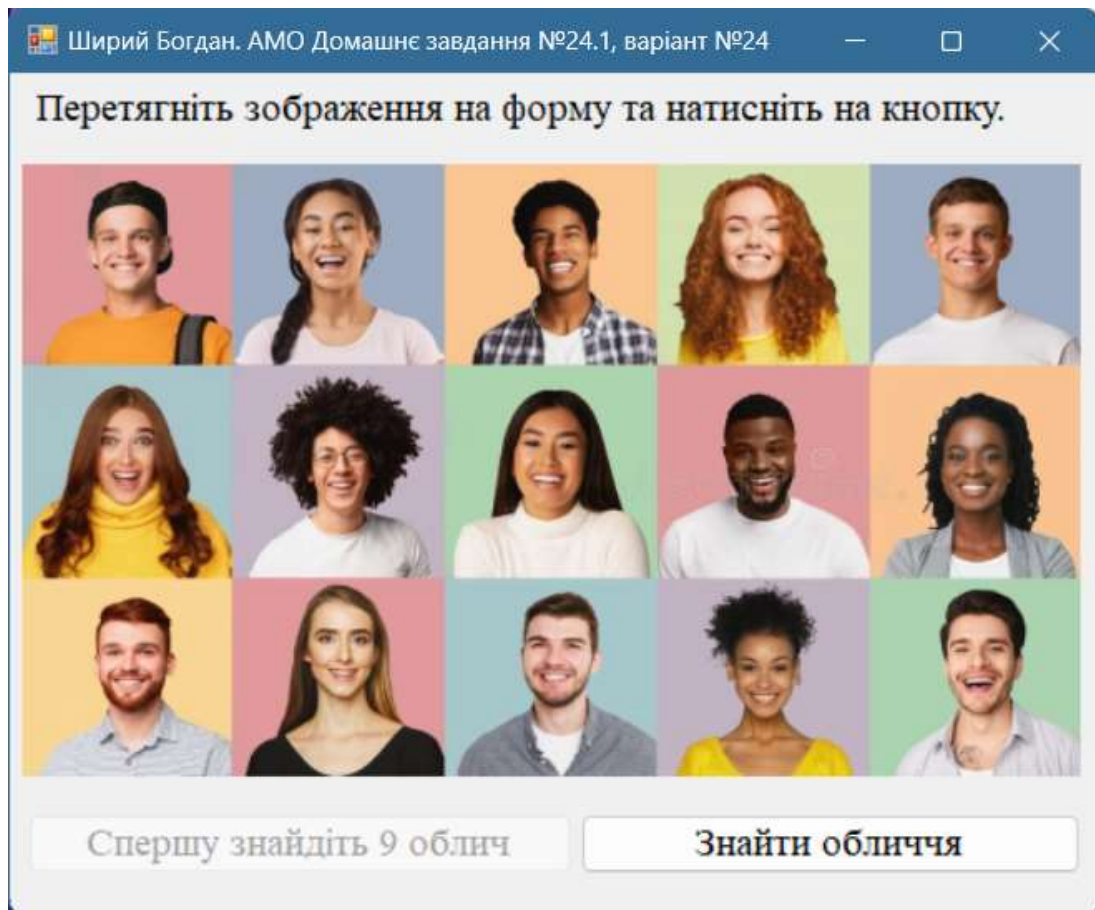


Рис. 2. Програма, після перетягування фото на форму.

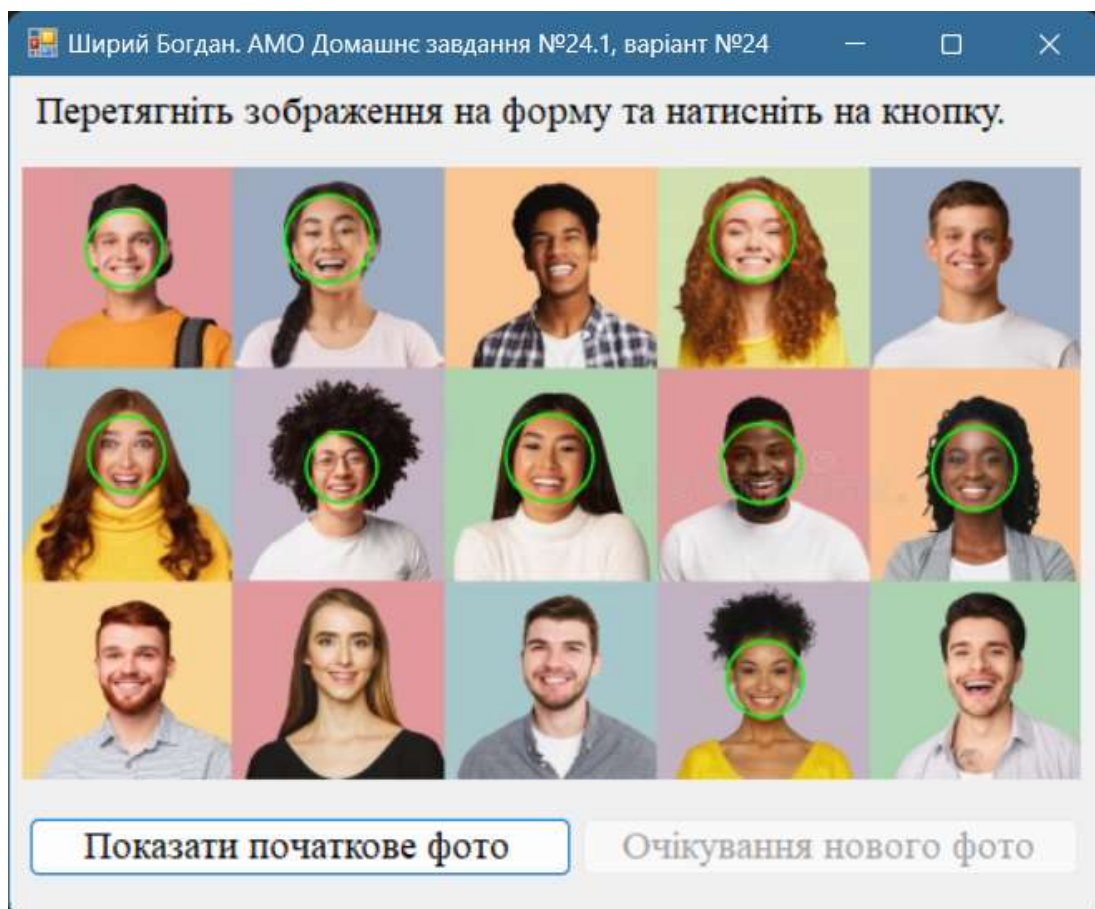


Рис. 3. Програма, після натиску на кнопку «Знайти обличчя».

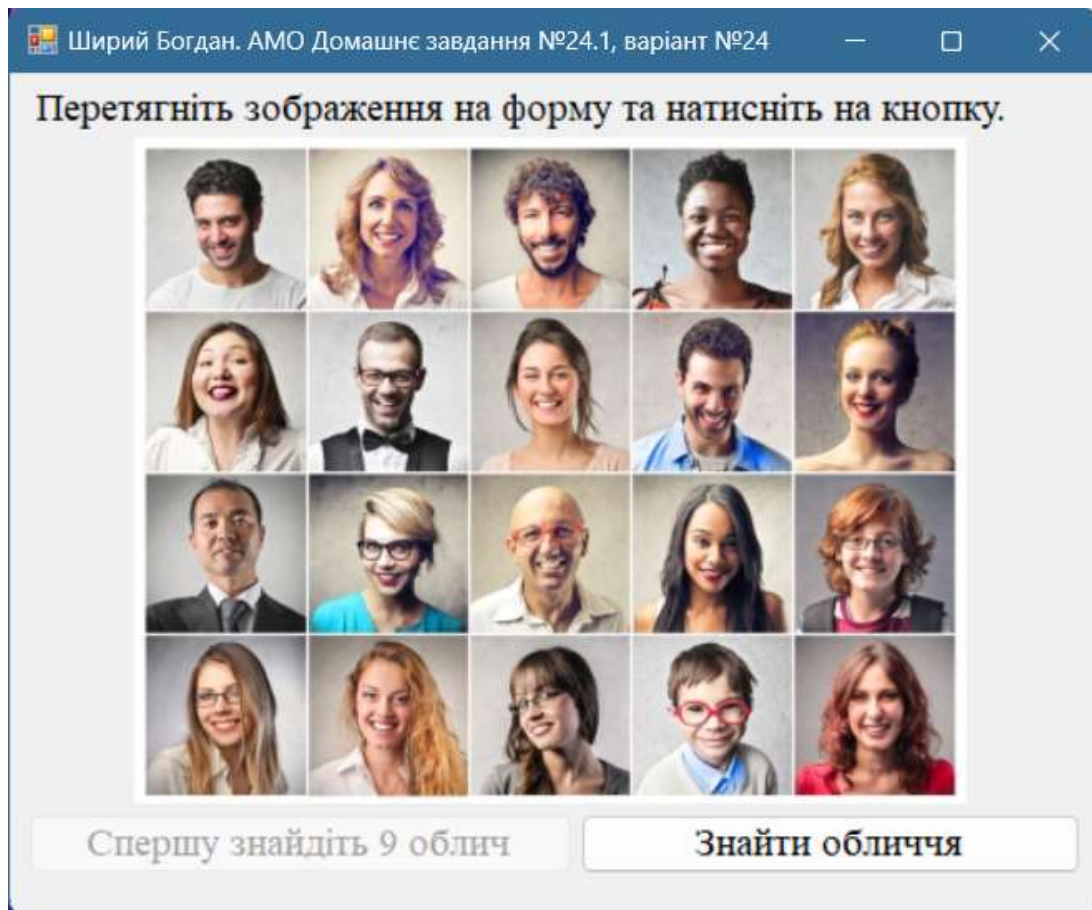


Рис. 4. Програма, після перетягування фото на форму.

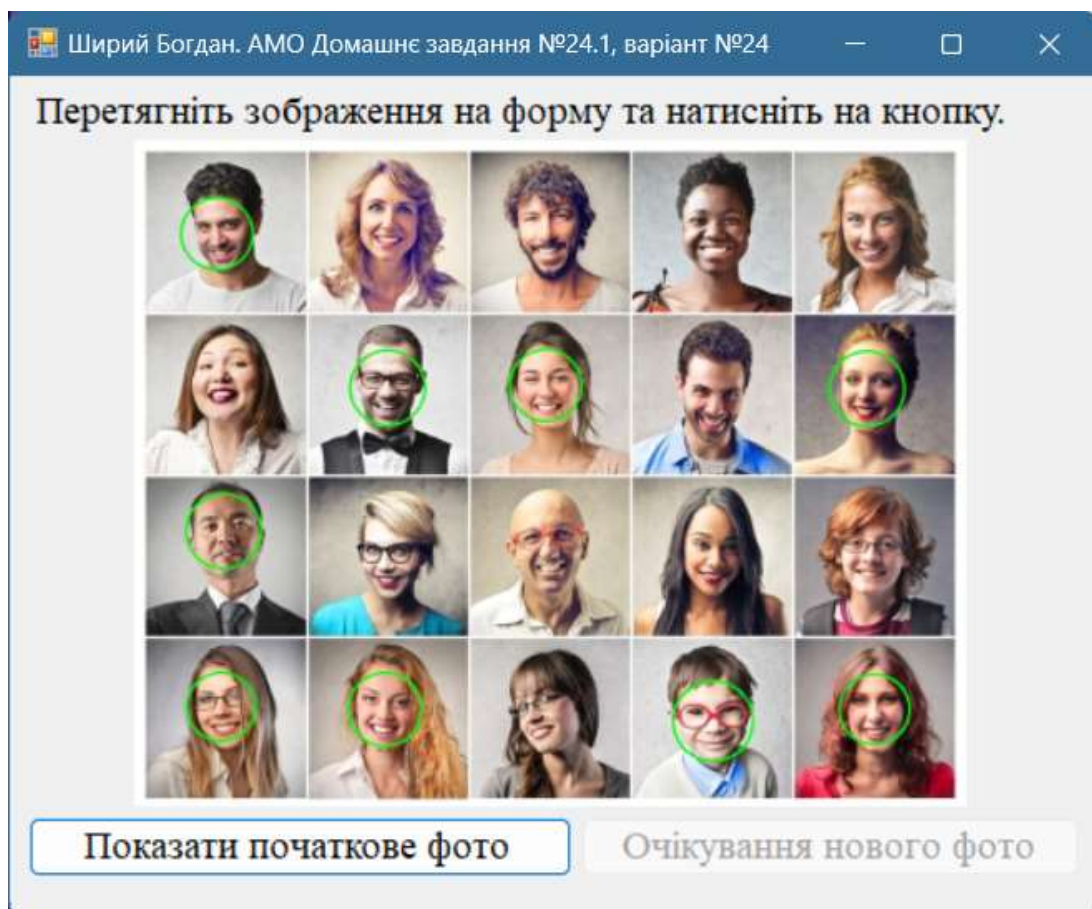


Рис. 5. Програма, після натиску на кнопку «Знайти обличчя».



Рис. 6. Показ початкового фото.

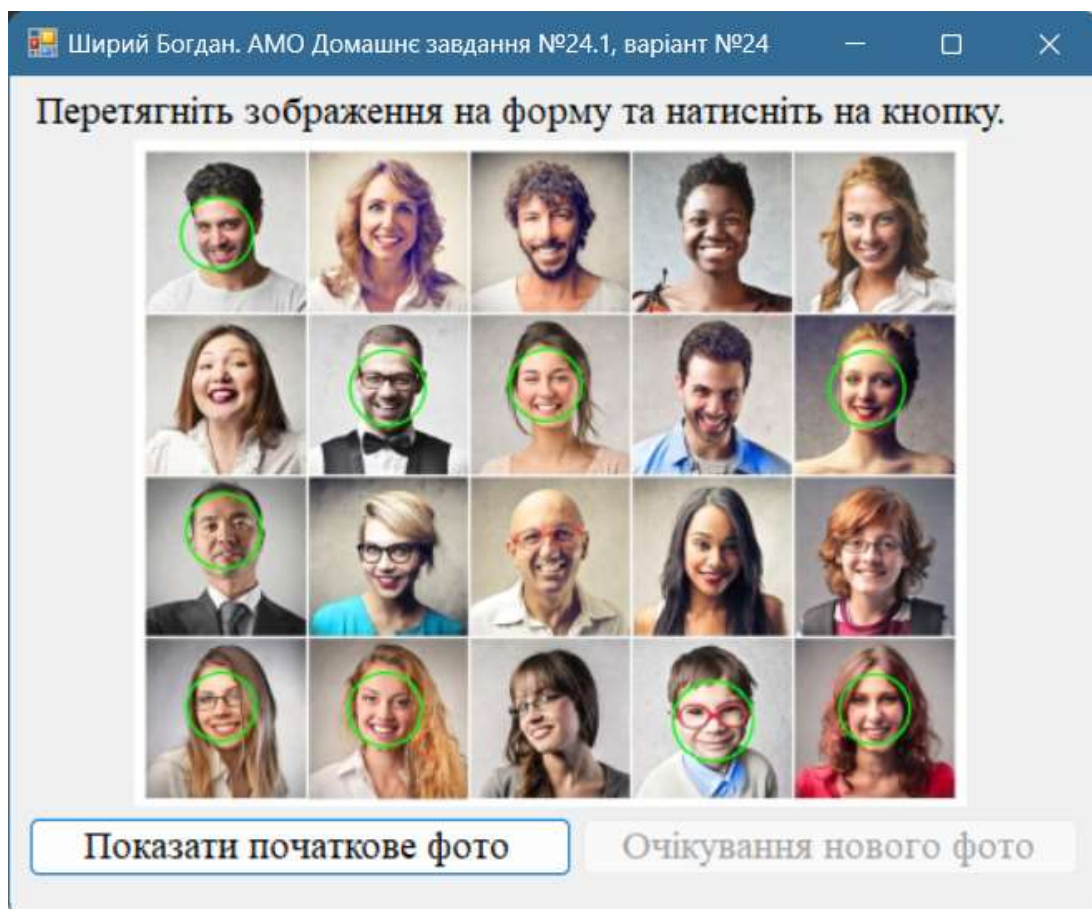


Рис. 7. Показ обробленого слова.

Відповідно, у лістингах навів код кожного файлу:

- ShyryiForm.cpp - лістинг 1,
- ShyryiForm.h - лістинг 2.

Лістинг 1. Код файлу MainForm.cpp.

```
#include "ShyryiForm.h"

using namespace System;
using namespace System::Windows::Forms;

[STAThreadAttribute]

int main() {
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
    acmShyryiHW24_1::ShyryiForm form;
    Application::Run(% form);
}
```

Лістинг 2. Код заголовкового файлу MainForm.h.

```
#pragma once
#include <opencv2/opencv.hpp>
#include <opencv2/core.hpp>

namespace acmShyryiHW24_1 {

    using namespace System;
    using namespace System::Windows::Forms;
    using namespace System::Drawing;
    using namespace System::IO;
    /// <summary>
    /// Summary for ShyryiForm
    /// </summary>
    public ref class ShyryiForm : public System::Windows::Forms::Form
    {
        String^ imagePath;
        bool IsInputImage = true, first = true, isChanged = false;

        // Завантажує фото та відображає його у PictureBox
        void LoadAndDisplayImage(String^ imagePath)
        {
            try
            {
                // Завантажити фото з файлу
                pictureBox->Image = Image::FromFile(imagePath);
            }
            catch (Exception^ ex)
            {
                MessageBox::Show("Помилка завантаження фото: " + ex->Message);
            }
        }

        // Обробник події перетягування файлів на форму
        void ShyryiForm_DragEnter(Object^ sender, DragEventArgs^ e)
        {
            // Перевірка, чи є перетягнуті файли зображення
            if (e->Data->GetDataPresent(DataFormats::FileDrop))
            {
                array<String^>^ files = (array<String^>^ e->Data
                    ->GetData(DataFormats::FileDrop));
                if (files != nullptr && files->Length > 0)
                {
                    String^ fileExtension = Path::GetExtension(files[0]);
                    if (fileExtension->Equals(".jpg",
                        StringComparison::InvariantCultureIgnoreCase) ||
                        fileExtension->Equals(".jpeg",
                        StringComparison::InvariantCultureIgnoreCase) ||
                        fileExtension->Equals(".png",
                        StringComparison::InvariantCultureIgnoreCase))
                    {

```



```

        {
            e->Effect = DragDropEffects::Copy;
        }
    }
}

// Обробник події перетягування та відпускання файлів на форму
void ShyryiForm_DragDrop(Object^ sender, DragEventArgs^ e)
{
    // Отримання списку перетягуваних файлів
    array<String^>^ files = (array<String^>^e->Data
        ->GetData(DataFormats::FileDrop);

    // Завантаження та відображення першого зображення зі списку файлів
    if (files != nullptr && files->Length > 0)
    {
        String^ fileExtension = Path::GetExtension(files[0]);
        if (fileExtension->Equals(".jpg",
            StringComparison::InvariantCultureIgnoreCase) ||
            fileExtension->Equals(".jpeg",
            StringComparison::InvariantCultureIgnoreCase) ||
            fileExtension->Equals(".png",
            StringComparison::InvariantCultureIgnoreCase))
        {
            imagePath = files[0];
            File::Delete("inputImage.jpg");
            File::Copy(imagePath, "inputImage.jpg");
            LoadAndDisplayImage(imagePath);

            IsInputImage = true; isChanged = true;
            this->FindFacesButton->Text = L"Знайти обличчя";
            this->changeViewButton->Text =
                L"Спершу знайдіть 9 обличч";
            this->changeViewButton->Enabled = false;
            this->FindFacesButton->Enabled = true;
        }
    }
}

public:
    ShyryiForm(void)
    {
        InitializeComponent();
        //
        //TODO: Add the constructor code here
        //
        pictureBox->SizeMode = PictureBoxSizeMode::Zoom;

        // Підключити обробники подій перетягування файлів на форму
        AllowDrop = true;
        DragEnter += gcnew DragEventHandler(this,
            &ShyryiForm::ShyryiForm_DragEnter);
        DragDrop += gcnew DragEventHandler(this,
            &ShyryiForm::ShyryiForm_DragDrop);

        this->changeViewButton->Enabled = false;
        this->FindFacesButton->Enabled = false;
    }

protected:
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    ~ShyryiForm()
    {
        if (components)
        {
            delete components;
        }
    }

private:
    System::Windows::Forms::PictureBox^ pictureBox;
private:
    System::Windows::Forms::Label^ InfoLabel;
private:
    System::Windows::Forms::Button^ FindFacesButton;
private:
    System::Windows::Forms::Button^ changeViewButton;

protected:

private:
    /// <summary>
    /// Required designer variable.

```

```

/// </summary>
System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
void InitializeComponent(void)
{
    this->pictureBox =
        (gcnew System::Windows::Forms::PictureBox());
    this->InfoLabel =
        (gcnew System::Windows::Forms::Label());
    this->FindFacesButton =
        (gcnew System::Windows::Forms::Button());
    this->changeViewButton =
        (gcnew System::Windows::Forms::Button());
    (cli::safe_cast<System::ComponentModel
        ::ISupportInitialize^(this->pictureBox))->BeginInit();
    this->SuspendLayout();
    //
    // pictureBox
    //
    this->pictureBox->Location = System::Drawing::Point(8, 38);
    this->pictureBox->Name = L"pictureBox";
    this->pictureBox->Size = System::Drawing::Size(680, 395);
    this->pictureBox->TabIndex = 0;
    this->pictureBox->TabStop = false;
    //
    // InfoLabel
    //
    this->InfoLabel->AutoSize = true;
    this->InfoLabel->Font = (gcnew System::Drawing
        ::Font(L"Times New Roman", 13.8F, System::Drawing
        ::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
    this->InfoLabel->Location = System::Drawing::Point(12, 9);
    this->InfoLabel->Name = L"InfoLabel";
    this->InfoLabel->Size = System::Drawing::Size(578, 26);
    this->InfoLabel->TabIndex = 1;
    this->InfoLabel->Text =
        L"Перетягніть зображення на форму та натисніть на кнопку.";
    //
    // FindFacesButton
    //
    this->FindFacesButton->Font = (gcnew
        System::Drawing::Font(L"Times New Roman", 13.8F,
        System::Drawing::FontStyle::Regular, System::Drawing
        ::GraphicsUnit::Point, static_cast<System::Byte>(204)));
    this->FindFacesButton->Location = System::Drawing::Point(367, 439);
    this->FindFacesButton->Name = L"FindFacesButton";
    this->FindFacesButton->Size = System::Drawing::Size(321, 36);
    this->FindFacesButton->TabIndex = 2;
    this->FindFacesButton->Text = L"Очікування фото";
    this->FindFacesButton->UseVisualStyleBackColor = true;
    this->FindFacesButton->Click += gcnew System
        ::EventHandler(this, &ShyryiForm::FindFacesButton_Click);
    //
    // changeViewButton
    //
    this->changeViewButton->Font = (gcnew System::Drawing
        ::Font(L"Times New Roman", 13.8F, System::Drawing
        ::FontStyle::Regular, System::Drawing
        ::GraphicsUnit::Point, static_cast<System::Byte>(204)));
    this->changeViewButton->Location = System::Drawing::Point(12, 439);
    this->changeViewButton->Name = L"changeViewButton";
    this->changeViewButton->Size = System::Drawing::Size(349, 36);
    this->changeViewButton->TabIndex = 3;
    this->changeViewButton->Text = L"Немає фото для обробки";
    this->changeViewButton->UseVisualStyleBackColor = true;
    this->changeViewButton->Click += gcnew System
        ::EventHandler(this, &ShyryiForm::changeViewButton_Click);
    //
    // ShyryiForm
    //
    this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);
    this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
    this->ClientSize = System::Drawing::Size(700, 500);
    this->Controls->Add(this->changeViewButton);
    this->Controls->Add(this->FindFacesButton);
}

```



```

        this->Controls->Add(this->InfoLabel);
        this->Controls->Add(this->pictureBox);
        this->Name = L"ShyryiForm";
        this->Text = L"Ширій Богдан. АМО Домашнє завдання %24.1, варіант %24";
        (cli::safe_cast<System::ComponentModel
            :ISupportInitialize^(this->pictureBox))->EndInit();
        this->ResumeLayout(false);
        this->PerformLayout();
    }
#pragma endregion
private: System::Void FindFacesButton_Click
    (System::Object^ sender, System::EventArgs^ e) {

    // Завантаження фото
    cv::Mat image = cv::imread("inputImage.jpg");

    // Переведення у відтінки сірого (для покращення роботи алгоритму)
    cv::Mat gray;
    cv::cvtColor(image, gray, cv::COLOR_BGR2GRAY);

    // Завантаження каскадного класифікатора обличчя
    cv::CascadeClassifier faceCascade;
    // Завантажте шлях до вашого файлу XML каскадного класифікатора обличчя
    faceCascade.load("haarcascade_frontalface_alt.xml");

    // Виявлення обличчя
    std::vector<cv::Rect> faces;
    faceCascade.detectMultiScale(gray, faces, 1.1, 3, 0, cv::Size(30, 30));

    // Межі областей обличчя
    int maxFaces = 9; // Максимальна кількість облич, яку ви хочете виявити
    int numFaces = std::min(static_cast<int>(faces.size()), maxFaces);

    // Позначення облич колами або овалами
    for (int i = 0; i < numFaces; ++i)
    {
        cv::Point center(faces[i].x + faces[i].width / 2,
            faces[i].y + faces[i].height / 2);
        cv::Size axes(faces[i].width / 2, faces[i].height / 2);
        cv::ellipse(image, center, axes, 0, 0, 360,
            cv::Scalar(0, 255, 0), 2);
    }

    // Збереження результату у файл .jpg
    cv::imwrite("outputImage.jpg", image);
    cv::imwrite("outputImage2.jpg", image);

    // Вивід результату
    first ?
        LoadAndDisplayImage("outputImage.jpg") :
        LoadAndDisplayImage("outputImage2.jpg");
    first = !first;
    IsInputImage = false;
    this->changeViewButton->Text = L"Показати початкове фото";
    this->changeViewButton->Enabled = true;
    this->FindFacesButton->Text = L"Очікування нового фото";
    this->FindFacesButton->Enabled = false;
}
private: System::Void changeViewButton_Click
    (System::Object^ sender, System::EventArgs^ e) {
    if (IsInputImage)
    {
        IsInputImage = false;
        !first ?
            LoadAndDisplayImage("outputImage.jpg") :
            LoadAndDisplayImage("outputImage2.jpg");
        this->changeViewButton->Text = L"Показати початкове фото";
    }
    else
    {
        IsInputImage = true;
        LoadAndDisplayImage(imagePath);
        this->changeViewButton->Text = L"Показати оброблене фото";
    }
}
};
};
};

```