using System;

using System.Drawing;

using System.Text;

class Steganography

{

public static void Encode(string inputImagePath, string outputImagePath, string message)

{

Bitmap bmp = new Bitmap(inputImagePath);

byte[] messageBytes = Encoding.UTF8.GetBytes(message + '\0'); // إضافة null لإنهاء النص

int msgIndex = 0;

for (int y = 0; y < bmp.Height; y++)

{

for (int x = 0; x < bmp.Width; x++)

{

if (msgIndex >= messageBytes.Length) break;

Color pixel = bmp.GetPixel(x, y);

byte r = (byte)((pixel.R & 0xFE) | ((messageBytes[msgIndex] >> 7) & 1));

byte g = (byte)((pixel.G & 0xFE) | ((messageBytes[msgIndex] >> 6) & 1));

byte b = (byte)((pixel.B & 0xFE) | ((messageBytes[msgIndex] >> 5) & 1));

bmp.SetPixel(x, y, Color.FromArgb(r, g, b));

messageBytes[msgIndex] <<= 3; // تحريك البيانات

msgIndex++;

}

}

bmp.Save(outputImagePath);

Console.WriteLine("تم حفظ الصورة مع النص المخفي!");

}

public static string Decode(string imagePath)

{

Bitmap bmp = new Bitmap(imagePath);

StringBuilder extractedMessage = new StringBuilder();

byte character = 0;

int bitIndex = 0;

for (int y = 0; y < bmp.Height; y++)

{

for (int x = 0; x < bmp.Width; x++)

{

Color pixel = bmp.GetPixel(x, y);

character |= (byte)((pixel.R & 1) << (7 - bitIndex));

character |= (byte)((pixel.G & 1) << (6 - bitIndex));

character |= (byte)((pixel.B & 1) << (5 - bitIndex));

bitIndex += 3;

if (bitIndex >= 8)

{

if (character == '\0') return extractedMessage.ToString();

extractedMessage.Append((char)character);

character = 0;

bitIndex = 0;

}

}

}

return extractedMessage.ToString();

}

static void Main()

{

string inputImage = "input.png"; // الصورة الأصلية

string outputImage = "output.png"; // الصورة مع النص المخفي

string secretMessage = "Hello, this is hidden text!";

Encode(inputImage, outputImage, secretMessage);

string extractedMessage = Decode(outputImage);

Console.WriteLine("النص المستخرج: " + extractedMessage);

}

}