МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ПОЛТАВСЬКА ПОЛІТЕХНІКА

ІМЕНІ ЮРІЯ КОНДРАТЮКА»

Навчально науковий інститут інформаційних технологій і робототехніки

Кафедра комп’ютерних та інформаційних технологій і систем

**Лабораторна робота № 1**

з навчальної дисципліни

"ОБ’ЄКТНО ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ"

Варіант – 8

**Виконала:**

*студента 201-ТК*

*Зуєва Микита Сергійовича*

**Перевірив:**

*Демиденко Максим Ігорович*

Полтава 2023

**Індивідуальне завдання**

1. Обчислити значення виразу за формулою (усі змінні приймають дійсні значення): 𝑥 − 10sin 𝑥 + cos (𝑥 − 𝑦)

2. Обчислити периметр і площу прямокутного трикутника за довжинами а й b двох катетів.

3. Скласти лінійну програму, яка друкує значення true, якщо зазначене висловлення є дійсним, і false – якщо ні: “Сума двох довільних цифр заданого тризначного натурального числа N дорівнює третій цифрі”. 25

4. Написати програму знаходження суми найбільшого й найменшого із трьох чисел.

5. Визначити кількість M-значних натуральних чисел, у яких сума цифр, що знаходяться у непарних розрядах, рівна N (1 ≤ N ≤ 30, 0 < M < 5).

6. Дано послідовність дійсних чисел а1, а2,..., аn. Указати ті її елементи, які належать відрізку [с, d].

7. У записці слова зашифровані: кожне з них записане навпаки. Розшифрувати повідомлення

Код програми:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.ToolBar;

namespace Lab1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

//завдання 1

// клік на кнопці OK

private void btnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рідки у число виникає виняток.

try

{

double x = Convert.ToDouble(txtInputX.Text); //змінна Х

double y = Convert.ToDouble(txtInputY.Text); //змінна Y

double result = x - Math.Pow(10,Math.Sin(x))+Math.Cos(x-y); //змінна для збереження результату обчислення

OutLbl.Text = result.ToString(); //вивід результату в Outlbl

}

catch

{

// обробка виключення:

// перемістити курсор в поле редагування

txtInputX.Focus();

}

}

//натискання кнопки в полі редагування змінної X

private void txtInputX\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (txtInputX.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

btnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування змінної Y

private void txtInputY\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (txtInputY.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

btnOK.Focus();

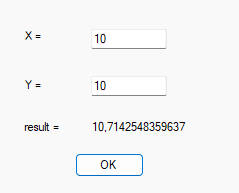
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 2

// клік на кнопці OK

private void tab2BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рідки у число виникає виняток.

try

{

double a = Convert.ToDouble(tab2InputA.Text); // катет а

double b = Convert.ToDouble(tab2InputB.Text); //катет b

double c = Math.Sqrt(Math.Pow(a, 2)+Math.Pow(b,2));// гіпотенуза c

double P = a + b + c; // знаходження периметра

tab2LblOutP.Text = P.ToString(); // вивід периметра

double S = (a \* b)/2; // знаходження площі

tab2LblOutS.Text = S.ToString(); // вивід площі

}

catch

{

// обробка виключення:

// перемістити курсор в поле редагування

tab2InputA.Focus();

}

}

//натискання кнопки в полі редагування катета a

private void tab2InputA\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab2InputA.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab2BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

////натискання кнопки в полі редагування катета b

private void tab2InputB\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab2InputB.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab2BtnOK.Focus();

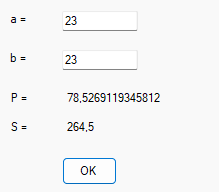
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 3

// клік на кнопці OK

private void tab3BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рядки у число виникає виняток.

try

{

int inputNumber = Convert.ToInt32(tab3InputNum.Text); //конвертація вводу

if (inputNumber < 100 || inputNumber > 999) // перевірка числа на відповідність умові

{

throw new ArgumentException();

}

int firstDigit = inputNumber / 100; // перша фира

int secondDigit = (inputNumber % 100) / 10; // друга цифра

int thirdDigit = inputNumber % 10; // третя цифра

// перевірка зазначеного висловлення

if (firstDigit + secondDigit == thirdDigit || secondDigit + thirdDigit == firstDigit || firstDigit + thirdDigit == secondDigit)

{

tab3LblOut.Text = "True";

}

else

{

tab3LblOut.Text = "False";

}

}

//обробка виключень

catch (ArgumentException)

{

//якщо введено не тризначне число

tab3LblOut.Text = "Число повинно містити три цифри!";

tab3InputNum.Focus();

}

catch

{

//якщо не введено нічого

tab3InputNum.Focus();

}

}

//натискання кнопки в полі редагування

private void tab3InputNum\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab3InputNum.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab3BtnOK.Focus();

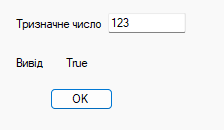
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 4

// клік на кнопці OK

private void tab4BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рядки у число виникає виняток.

try

{

int firstNum = Convert.ToInt32(tab4InputNum1.Text); //перше число

int secondNum = Convert.ToInt32(tab4InputNum2.Text); //друге число

int thirdNum = Convert.ToInt32(tab4InputNum3.Text); //третє число

int maxNum = Math.Max(firstNum, Math.Max(secondNum, thirdNum)); //пошук максимального числа

int minNum = Math.Min(firstNum, Math.Min(secondNum, thirdNum)); // пошук мінімального числа

int sum = maxNum + minNum; // знаходження суми

tab4LblOut.Text = sum.ToString(); // вивід результату

}

catch

{

// обробка виключення:

// перемістити курсор в поле редагування

tab4InputNum1.Focus();

}

}

//натискання кнопки в полі редагування tab4InputNum1

private void tab4InputNum1\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab4InputNum1.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab4BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab4InputNum2

private void tab4InputNum2\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab4InputNum2.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab4BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab4InputNum3

private void tab4InputNum3\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab4InputNum3.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab4BtnOK.Focus();

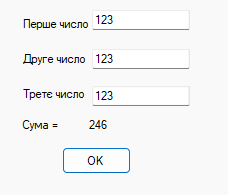
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 5

// клік на кнопці OK

private void tab5BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рядки у число виникає виняток.

try

{

int M = Convert.ToInt32(tab5InputM.Text);

int N = Convert.ToInt32(tab5InputN.Text);

if (M < 1 || M >= 5) // перевірка правильності введеного М

throw new ArgumentOutOfRangeException();

if (N < 1 || N > 30) // перевірка правильності введеного N

throw new ArgumentOutOfRangeException();

int count = 0; // лічильник для збереження к-сті M-значних натуральних чисел

int lowerBound = (int)Math.Pow(10, M - 1); //мінімальне M-значне число

int upperBound = (int)Math.Pow(10, M); //максимальне M-значне число

for (int num = lowerBound; num < upperBound; num++) //цикл у якому виконується обрахунок к-сті

{

int sum = 0; //змінна для знаходження суми, необхідної для знаходження к-сті

string numStr = num.ToString();

for (int i = 0; i < M; i++) //цикл знаходження не парних розрядів

{

if (i % 2 == 0) // Непарні розряди (1, 3, 5...)

{

sum += numStr[i] - '0';

}

}

if (sum == N) //умова при виконанні якої збільшується лічильник count

{

count++;

}

}

tab5LblOut.Text = count.ToString(); //вивід результату

}

// обробка виключень:

catch (ArgumentOutOfRangeException)

{

tab5LblOut.Text = "Введіть коректні дані";

tab5InputM.Focus();

}

catch

{

// перемістити курсор в поле редагування

tab5InputM.Focus();

}

}

//натискання кнопки в полі редагування tab5InputM

private void tab5InputM\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab5InputM.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab5BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab5InputN

private void tab5InputN\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab5InputN.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab5BtnOK.Focus();

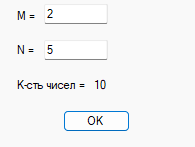
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 6

// клік на кнопці OK

private void tab6BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рядки у число виникає виняток.

try

{

double start = Convert.ToDouble(tab6InputStart.Text); // початок послідовності

double end = Convert.ToDouble(tab6InputEnd.Text); // кінець послідовності

double c = Convert.ToDouble(tab6InputC.Text); // початок відрізку

double d = Convert.ToDouble(tab6InputD.Text); // кінець відрізку

if (start > end) // створення виключення якщо початок послідовності більше ніж кінець

{

throw new ArgumentException();

}

// створення послідовності

int startInt = (int)Math.Floor(start);

int endInt = (int)Math.Floor(end);

double[] arr = Enumerable.Range(startInt, endInt - startInt + 1).Select(x => (double)x).ToArray();

var result = arr.Where(num => num >= c && num <= d); //знаходження результату

tab6LblOut.Text = result.Any() ? string.Join(", ", result) : "Немає чисел у вказаному діапазоні"; //вивід

}

// обробка виключень:

catch (ArgumentException)

{

tab6LblOut.Text = "Початкове значення більше за кінцеве";

tab6InputStart.Focus();

}

catch

{

tab6InputStart.Focus();

}

}

//натискання кнопки в полі редагування tab6InputStart

private void tab6InputStart\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab6InputStart.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab6BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab6InputEnd

private void tab6InputEnd\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab6InputEnd.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab6BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab6InputC

private void tab6InputC\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab6InputC.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab6BtnOK.Focus();

return;

}

// інші символи заборонені

e.Handled = true;

}

//натискання кнопки в полі редагування tab6InputD

private void tab6InputD\_KeyPress(object sender, KeyPressEventArgs e)

{

// Правильними символами вважаються цифри,

// Кома, <Enter> і <Backspace>.

// Будемо вважати правильним символом

// також крапку, замінимо її комою.

// Інші символи заборонені.

// Щоб заборонений символ не відображався

// у полі редагування, привласним

// значення true властивості Handled параметра e

if ((e.KeyChar >= '0') && (e.KeyChar <= '9'))

{

// цифра

return;

}

if (e.KeyChar == '.')

{

// крапку замінемо комою

e.KeyChar = ',';

}

if (e.KeyChar == ',')

{

if (tab6InputD.Text.IndexOf(',') != -1)

{

// кома вже є в полі редагування

e.Handled = true;

}

return;

}

if (Char.IsControl(e.KeyChar))

{

// <Enter>, <Backspace>, <Esc>

if (e.KeyChar == (char)Keys.Enter)

// натиснута клавіша <Enter>

// встановити курсор на кнопку OK

tab6BtnOK.Focus();

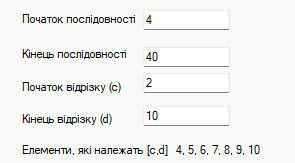
return;

}

// інші символи заборонені

e.Handled = true;

}



//завдання 7

// клік на кнопці OK

private void tab7BtnOK\_Click(object sender, EventArgs e)

{

// Якщо у полі редагування немає даних,

// то при спробі перетворити порожні

// рядки у число виникає виняток.

try

{

string encryptedMessage = tab7InputMessage.Text; // вхідне зашифроване повідомлення повідомлення

if (string.IsNullOrEmpty(encryptedMessage)) // створення виключення якщо немає вхідного повідомлення

throw new ArgumentException();

string[] words = encryptedMessage.Split(' '); // створення масиву слів зашифрованого повідомлення

for (int i = 0; i < words.Length; i++) // цикл для розшифрування повідомлення

{

char[] charArray = words[i].ToCharArray();

Array.Reverse(charArray);

words[i] = new string(charArray);

}

string decryptedMessage = string.Join(" ", words); // розшифроване повідомлення

tab7LblOut.Text = decryptedMessage; // вивід розшифрованого повідомлення

}

// обробка виключень

catch (ArgumentException)

{

tab7LblOut.Text = "Повідомлення не може бути порожнім.";

tab7InputMessage.Focus();

}

}

//натискання кнопки в полі редагування tab7InputMessage

private void tab7InputMessage\_KeyPress(object sender, KeyPressEventArgs e)

{

// при натисканні <Enter> встановити курсор на кнопку OK

if (Char.IsControl(e.KeyChar))

{

if (e.KeyChar == (char)Keys.Enter)

tab7BtnOK.Focus();

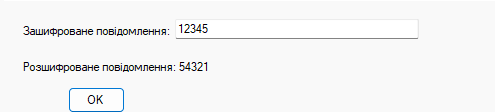
return;

}

}

}

}



Посилання на файли проєкту:

https://github.com/Nikita-Zuev/OOP