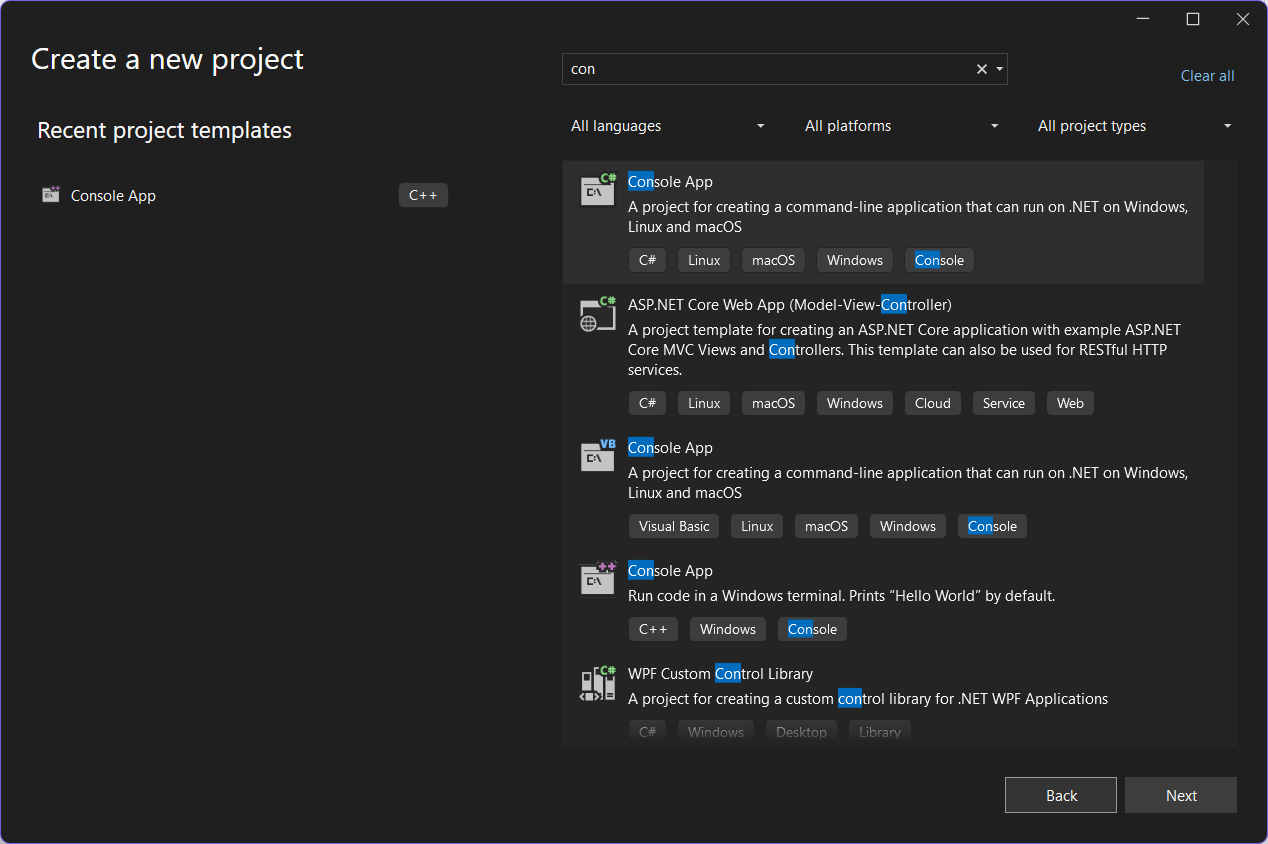
**LAB 2:**

**WINDOWS FORMS APPLICATION ĐƠN GIẢN**

# B. Nội dung

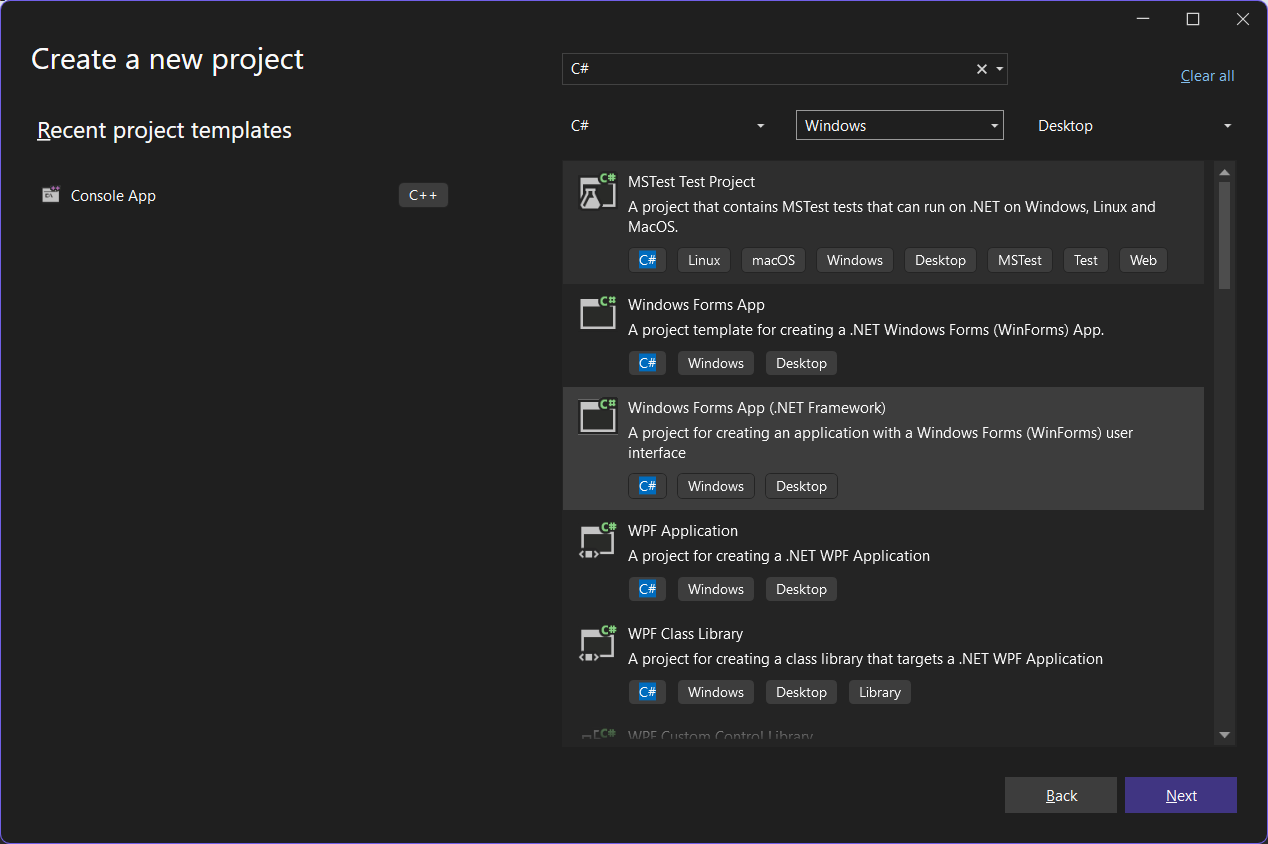
## 1. Tạo Project Windows Forms Application trên Microsoft visualstudio 2022

**Bước 1:** Khởi động chương trình visual studio 2022. Hình ảnh dưới đây là màn hình khởi động của ứng dụng:

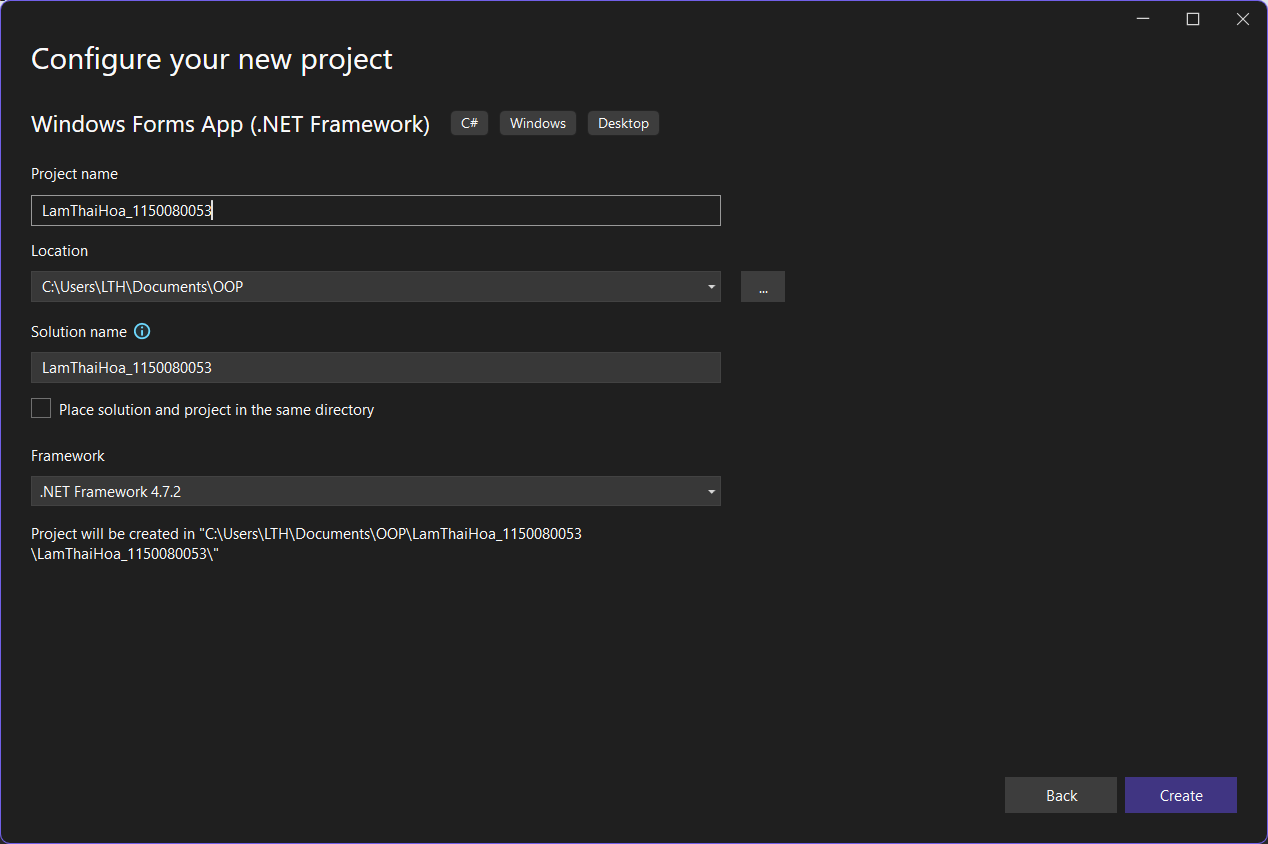


**Bước 2:** Chọn Create a new project trên màn hình khởi động. Trong màn hình tiếp theo, ở mục All languages → chọn C# trong danh sách ngôn ngữ lập trình.

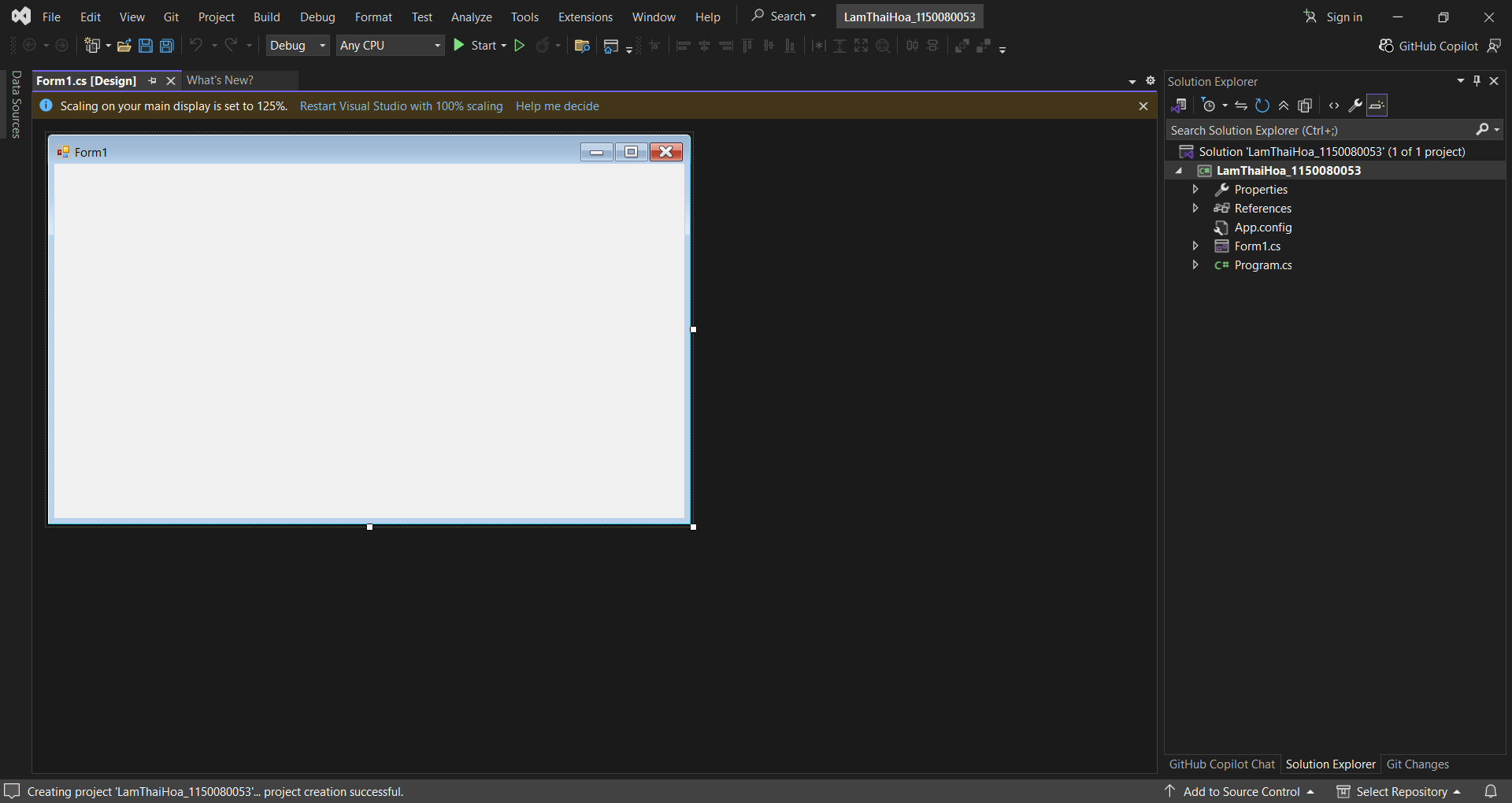
Mục All platforms → chọn Windows. Ở mục All project types → chọn Desktop.



**Bước 3:** Trong cửa sổ tiếp theo, bạn điền đầy đủ các thông tin về Project.

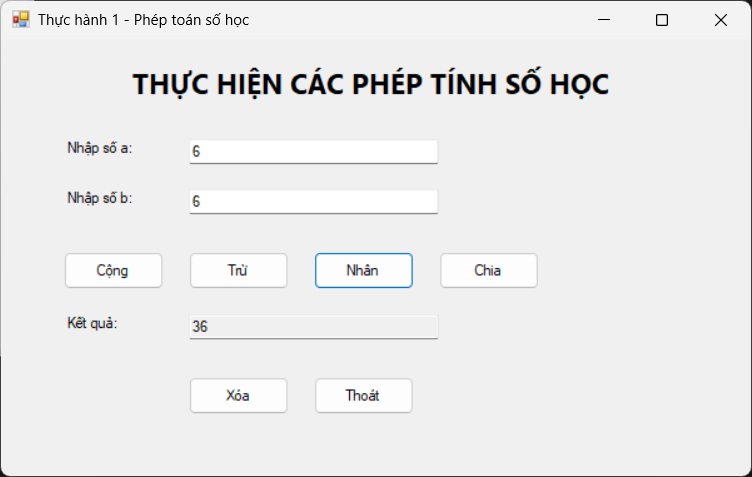


Đây là kết quả!



# 2. Thực hành

## Thực hành 1: Thiết kế form



**Form.cs**

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;

namespace LamThaiHoa\_1150080053

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

try

{

double a = Convert.ToDouble(txtA.Text.Trim());

double b = Convert.ToDouble(txtB.Text.Trim());

txtKetQua.Text = (a + b).ToString();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi!",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

double a = Convert.ToDouble(txtA.Text.Trim());

double b = Convert.ToDouble(txtB.Text.Trim());

txtKetQua.Text = (a - b).ToString();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi!",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

double a = Convert.ToDouble(txtA.Text.Trim());

double b = Convert.ToDouble(txtB.Text.Trim());

txtKetQua.Text = (a \* b).ToString();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi!",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

double a = Convert.ToDouble(txtA.Text.Trim());

double b = Convert.ToDouble(txtB.Text.Trim());

if (b == 0)

{

MessageBox.Show("Mẫu số không được phép bằng 0. Vui lòng nhập lại!",

"Thông báo lỗi!", MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtB.Clear();

txtB.Focus();

return;

}

txtKetQua.Text = (a / b).ToString();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi!",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

txtA.Clear();

txtB.Clear();

txtKetQua.Clear();

txtA.Focus();

}

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

{

DialogResult result = MessageBox.Show("Bạn có thực sự muốn thoát không?",

"Xác nhận thoát", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

this.Close();

}

}

private void txtA\_TextChanged(object sender, EventArgs e)

{

}

}

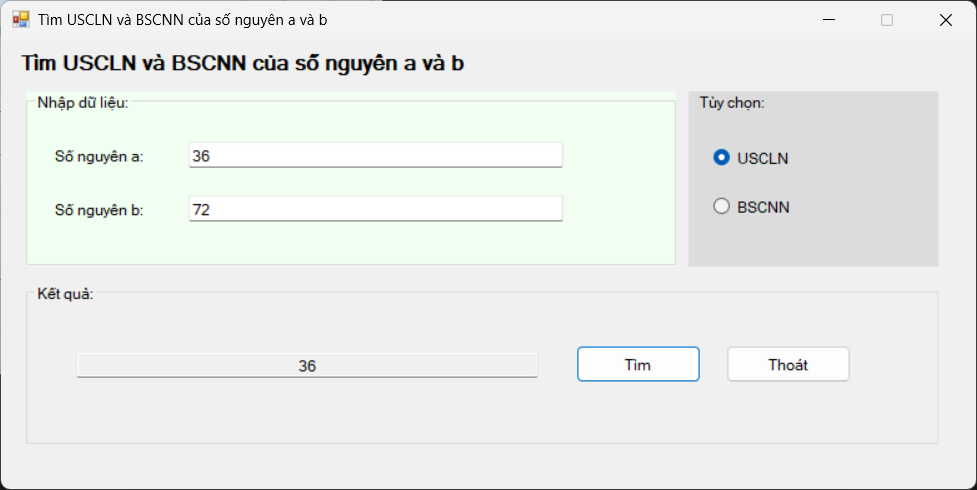
}

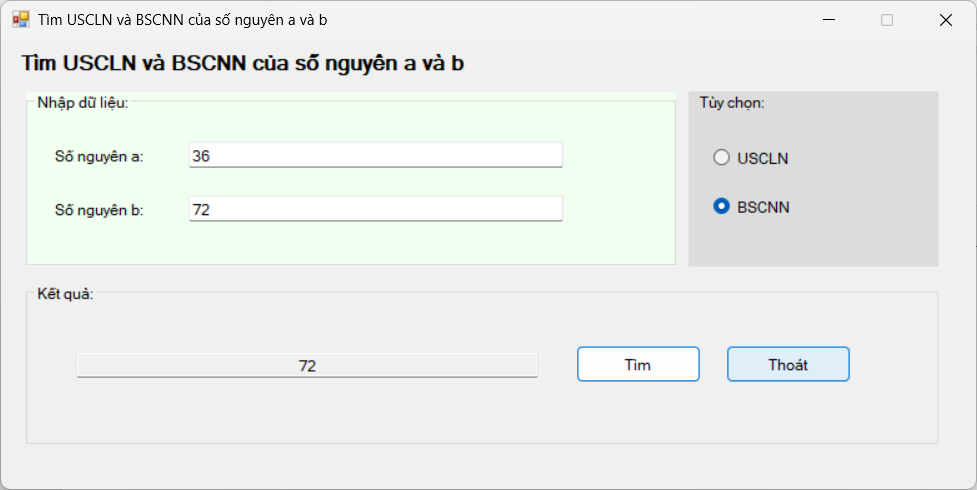
## Áp dụng 1: Thêm vào một project có tên là “ApDung1”, thực hiện tìm ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số nguyên a và b:

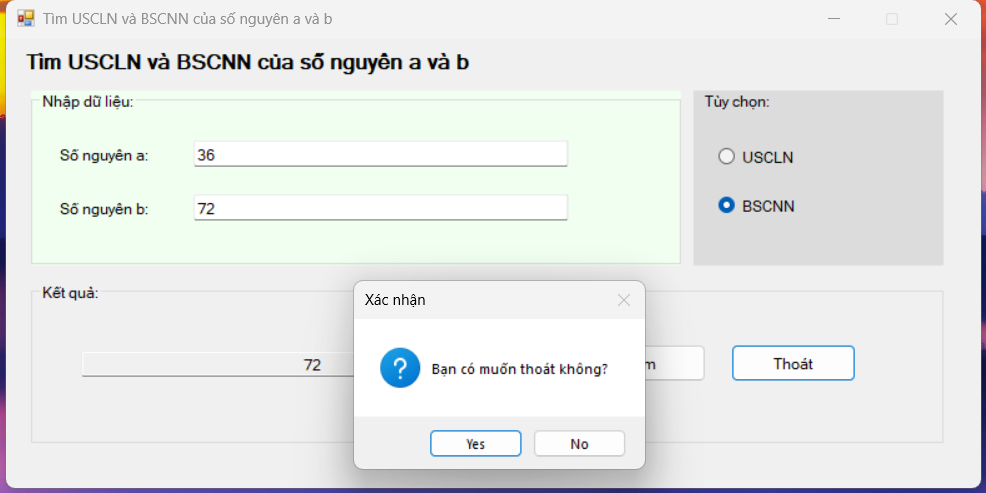
Yêu cầu:

− Viết 2 hàm để tính ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số a và b. Khi nhấn vào nút Tìm, nếu người dùng chọn tùy chọn tìm USCLN thì hiển thị ước số chung lớn nhất, ngược lại hiển thị bội số chung nhỏ nhất.

− Nút thoát để thoát khỏi chương trình.







**Form.cs**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Globalization;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace ApDung1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

this.AcceptButton = btnTim;

this.CancelButton = btnThoat;

rdoUSCLN.Checked = true;

}

private static long Gcd(long a, long b)

{

a = Math.Abs(a); b = Math.Abs(b);

while (b != 0) { long t = a % b; a = b; b = t; }

return a;

}

private static long Lcm(long a, long b)

{

if (a == 0 || b == 0) return 0;

return Math.Abs(a / Gcd(a, b) \* b);

}

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

{

try

{

long a = long.Parse(txtA.Text.Trim());

long b = long.Parse(txtB.Text.Trim());

long kq = rdoUSCLN.Checked ? Gcd(a, b) : Lcm(a, b);

txtKetQua.Text = kq.ToString();

}

catch

{

MessageBox.Show("Vui lòng nhập 2 số nguyên hợp lệ.",

"Lỗi nhập liệu", MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtKetQua.Clear();

}

}

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

{

if (MessageBox.Show("Bạn có muốn thoát không?",

"Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

Close();

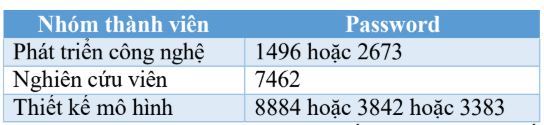
}

}

}

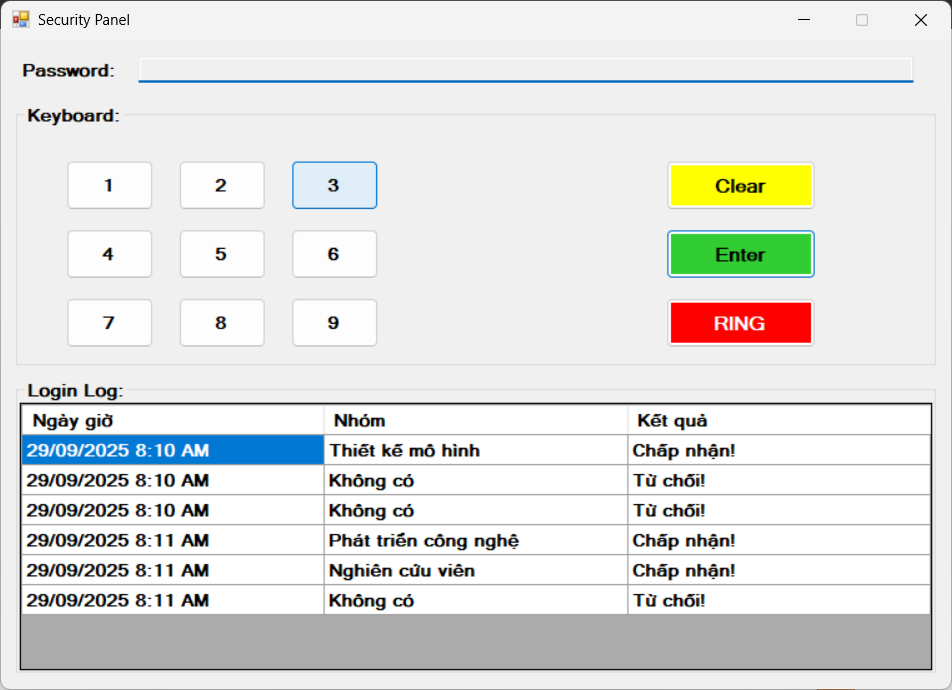
}

## Áp dụng 2: Xây dựng ứng dụng mô phỏng thiết bị Security Panel theo yêu cầu: Một phòng Lab muốn lắp thiết bị bảo vệ đặt được ở bên ngoài cửa ra vào có tên là Security. Chỉ cho phép những cá nhân nào nhập đúng password đã được quy định mới được vào phòng. Bảng dưới là password được cấp cho những nhóm thành viên:

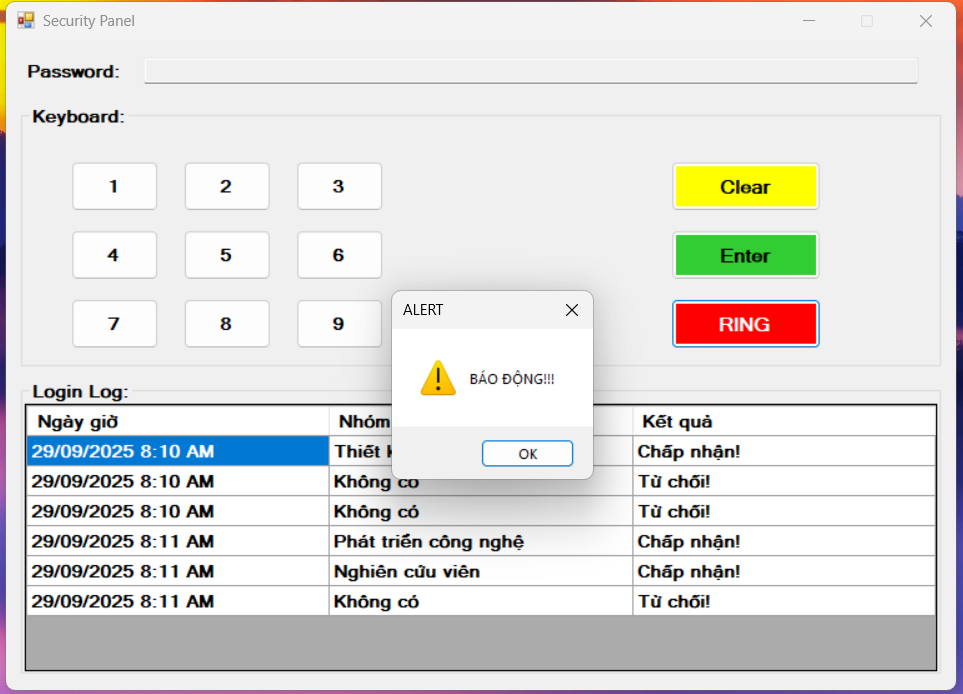


Mỗi một lần nhập password sẽ có 2 trạng thái: chấp nhận hoặc từ chối. Tất cả các thông tin truy cập sẽ được hiển thị trong khung thông tin phía dưới bàn phím số.

Nếu là chấp nhận thì ghi lại và hiển thị thông về về ngày, thời gian, tên nhóm và chấp nhận. Nếu là từ chối thì ghi lại và hiển thị thông về về ngày, thời gian, từ chối.



Khi nhấn Ring

**Form.cs**

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;

namespace ApDung2

{

public partial class Form1 : Form

{

private readonly Dictionary<string, string> passwordDict =

new Dictionary<string, string>

{

{ "1496", "Phát triển công nghệ" },

{ "2673", "Phát triển công nghệ" },

{ "7462", "Nghiên cứu viên" },

{ "8884", "Thiết kế mô hình" },

{ "3842", "Thiết kế mô hình" },

{ "3383", "Thiết kế mô hình" },

};

public Form1()

{

InitializeComponent();

this.AutoScaleMode = AutoScaleMode.None;

this.Font = new System.Drawing.Font("Microsoft Sans Serif", 9F);

this.AcceptButton = btnEnter;

}

private void Form1\_Load(object sender, EventArgs e)

{

dgvLog.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill;

dgvLog.RowHeadersVisible = false;

dgvLog.ReadOnly = true;

txtPassword.ReadOnly = true;

txtPassword.Clear();

}

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

{

var b = sender as Button;

if (b != null) txtPassword.Text += b.Text;

}

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

{

txtPassword.Clear();

txtPassword.Focus();

}

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

{

string input = (txtPassword.Text ?? "").Trim();

string group = "Không có";

string result = "Từ chối!";

string g;

if (passwordDict.TryGetValue(input, out g))

{

group = g;

result = "Chấp nhận!";

}

dgvLog.Rows.Add(DateTime.Now.ToString("g"), group, result);

txtPassword.Clear();

txtPassword.Focus();

}

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

{

MessageBox.Show("BÁO ĐỘNG!!!", "ALERT",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

## Thực hành 2: Tính tiền công dịch vụ tại phòng khám nha khoa.

Giả sử, tại một phòng khám nha khoa cần tính tiền các loại dịch vụ khác nhau.

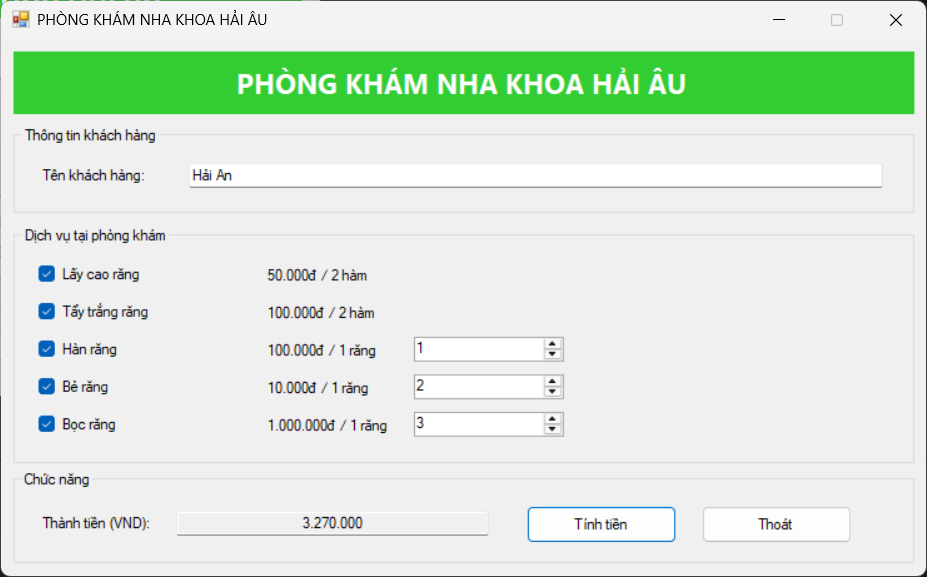
Một biểu mẫu thanh toán gồm:

− Tên khách hàng.

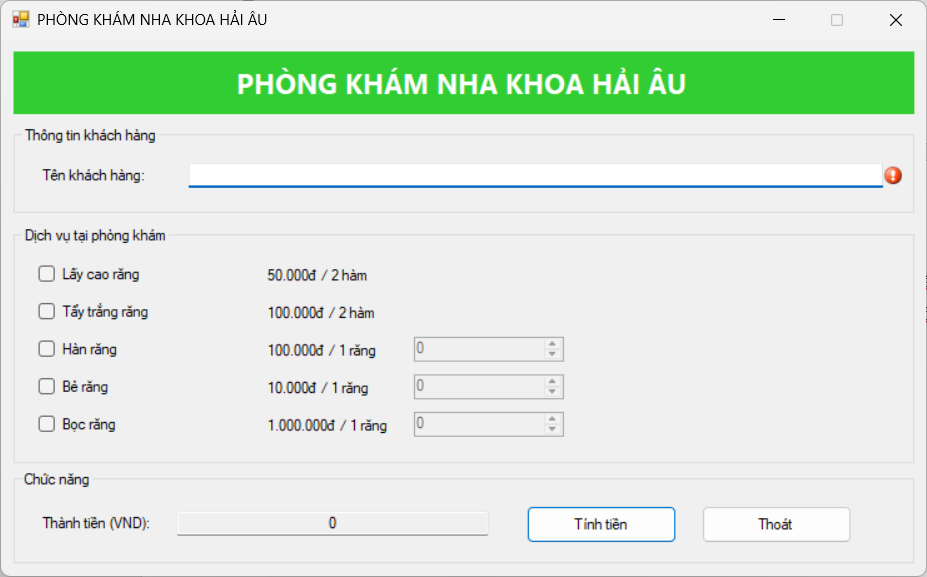
− Các loại dịch vụ như: lấy cao răng (50.000đ), tẩy trắng răng (100.000đ), hàn răng (100.000đ), bẻ răng (10.000đ), bọc răng (1.000.000đ).

− Hàn răng, bẻ răng và bọc răng được tính theo chiếc răng. Giá tiền trên được tính theo 1 chiếc răng.

− Biểu mẫu tính tiền không được để trống tên khách hàng. Có cảnh báo nếu trống.



Cảnh báo nếu trống



**Form.cs**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Globalization;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace LamThaiHoa\_1150080053\_TH2

{

public partial class Form1 : Form

{

private const int GIA\_LAY\_CAO\_RANG = 50\_000;

private const int GIA\_TAY\_TRANG = 100\_000; private const int GIA\_HAN\_RANG = 100\_000;

private const int GIA\_BE\_RANG = 10\_000;

private const int GIA\_BOC\_RANG = 1\_000\_000;

public Form1()

{

InitializeComponent();

this.AcceptButton = btnTinhTien;

this.CancelButton = btnThoat;

nupHanRang.Enabled = nupBeRang.Enabled = nupBocRang.Enabled = false;

txtThanhTien.Text = "0";

}

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

{

if (!ValidateChildren(ValidationConstraints.Enabled)) return;

long thanhTien = 0;

if (chkLayCaoRang.Checked) thanhTien += GIA\_LAY\_CAO\_RANG;

if (chkTayTrangRang.Checked) thanhTien += GIA\_TAY\_TRANG;

if (chkHanRang.Checked) thanhTien += (long)nupHanRang.Value \* GIA\_HAN\_RANG;

if (chkBeRang.Checked) thanhTien += (long)nupBeRang.Value \* GIA\_BE\_RANG;

if (chkBocRang.Checked) thanhTien += (long)nupBocRang.Value \* GIA\_BOC\_RANG;

txtThanhTien.Text = thanhTien.ToString("#,0", CultureInfo.GetCultureInfo("vi-VN"));

}

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

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Xác nhận",

MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

Close();

}

}

private void txtTenKH\_Validating(object sender, CancelEventArgs e)

{

if (string.IsNullOrWhiteSpace(txtTenKH.Text))

{

e.Cancel = true;

err.SetError(txtTenKH, "Tên khách hàng không được để trống!");

}

else

{

e.Cancel = false;

err.SetError(txtTenKH, null);

}

}

private void chkHanRang\_CheckedChanged(object sender, EventArgs e)

{

nupHanRang.Enabled = chkHanRang.Checked;

if (!chkHanRang.Checked) nupHanRang.Value = 0;

}

private void chkBeRang\_CheckedChanged(object sender, EventArgs e)

{

nupBeRang.Enabled = chkBeRang.Checked;

if (!chkBeRang.Checked) nupBeRang.Value = 0;

}

private void chkBocRang\_CheckedChanged(object sender, EventArgs e)

{

nupBocRang.Enabled = chkBocRang.Checked;

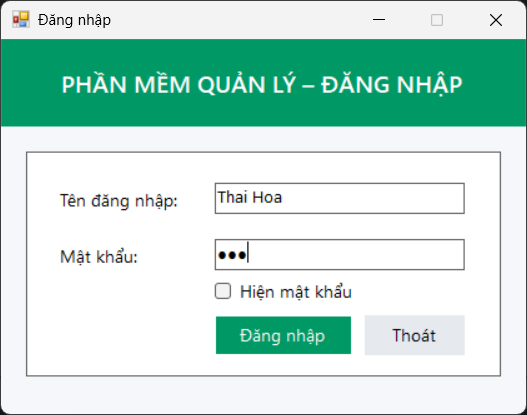
if (!chkBocRang.Checked) nupBocRang.Value = 0;

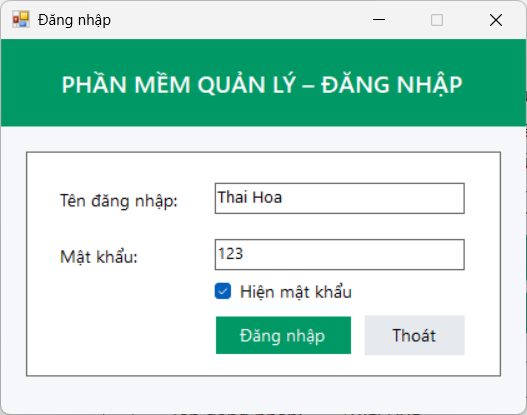
}

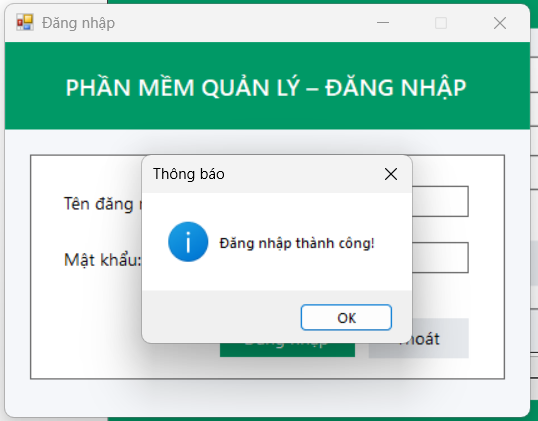
}

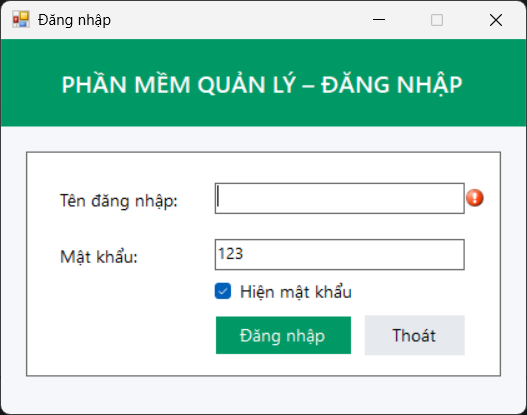
}

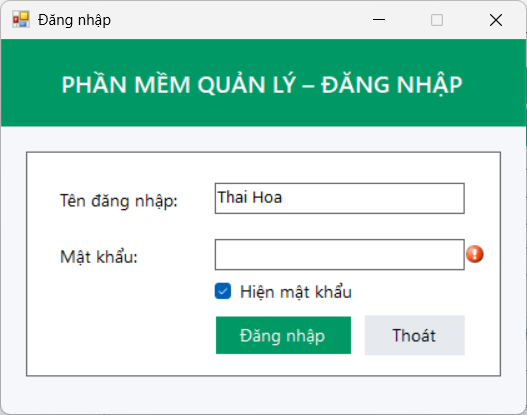
## Áp dụng 3: Thiết kế và lập trình form Đăng nhập của một ứng dụng. Thông tin cần đăng nhập gồm username và password. Yêu cầu người dùng không được để trống trường thông tin nào. Nếu để trống trường thông tin nào thì cảnh báo người dùng để bổ sung.











**Form.cs**

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;

namespace ApDung3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

this.AcceptButton = btnLogin;

this.CancelButton = btnExit;

}

private void chkShow\_CheckedChanged(object sender, EventArgs e)

{

txtPass.UseSystemPasswordChar = !chkShow.Checked;

}

private void txtUser\_Validating(object sender, CancelEventArgs e)

{

if (string.IsNullOrWhiteSpace(txtUser.Text))

{ e.Cancel = true; err.SetError(txtUser, "Không được để trống tên đăng nhập"); }

else

{ e.Cancel = false; err.SetError(txtUser, null); }

}

private void txtPass\_Validating(object sender, CancelEventArgs e)

{

if (string.IsNullOrWhiteSpace(txtPass.Text))

{ e.Cancel = true; err.SetError(txtPass, "Không được để trống mật khẩu"); }

else

{ e.Cancel = false; err.SetError(txtPass, null); }

}

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

{

if (!ValidateChildren(ValidationConstraints.Enabled)) return;

MessageBox.Show("Đăng nhập thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnThoat\_Click(object sender, EventArgs e) => Close();

}

}

## Thực hành 3: Thiết kế form frmXuLySo với control ListBox

**Yêu cầu:**

− Chỉ cho nhập số vào textbox Nhập số nguyên

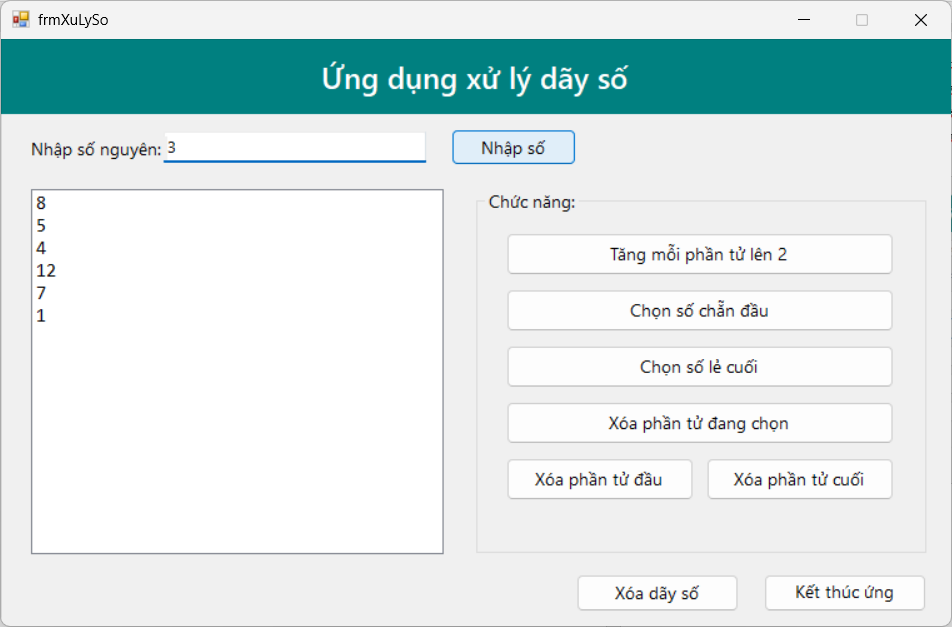
− Khi nhập dữ liệu vào textbox Nhập số nguyên và nhấn Enter (hoặc click

vào nút Nhập số) thì số mới nhập này được thêm vào Listbox đồng thời dữ

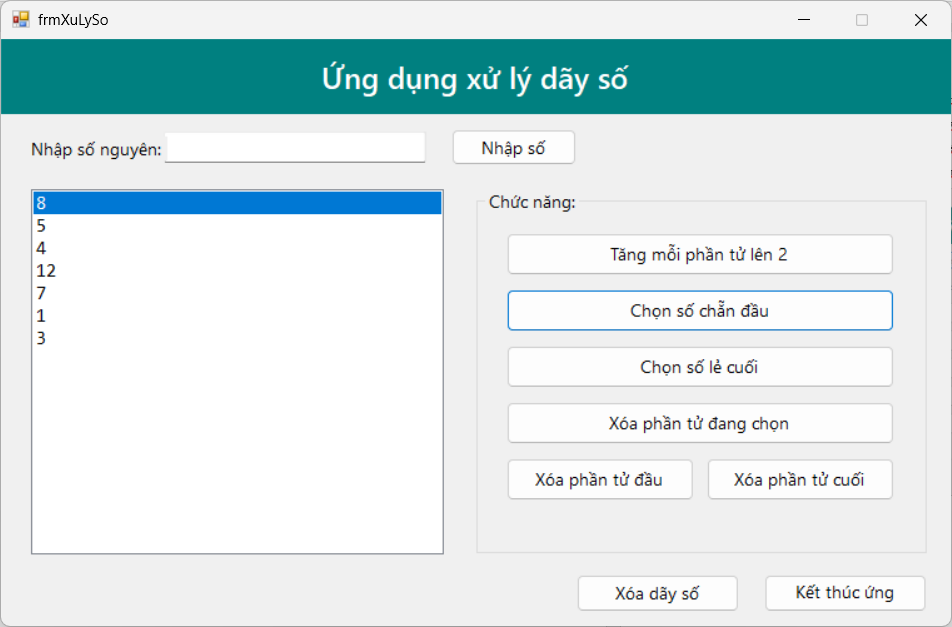
liệu trong textbox bị xóa và focus được chuyển về textbox

− Khi nhấn vào các phím chức năng, yêu cầu sẽ được thực hiện trên listbox.

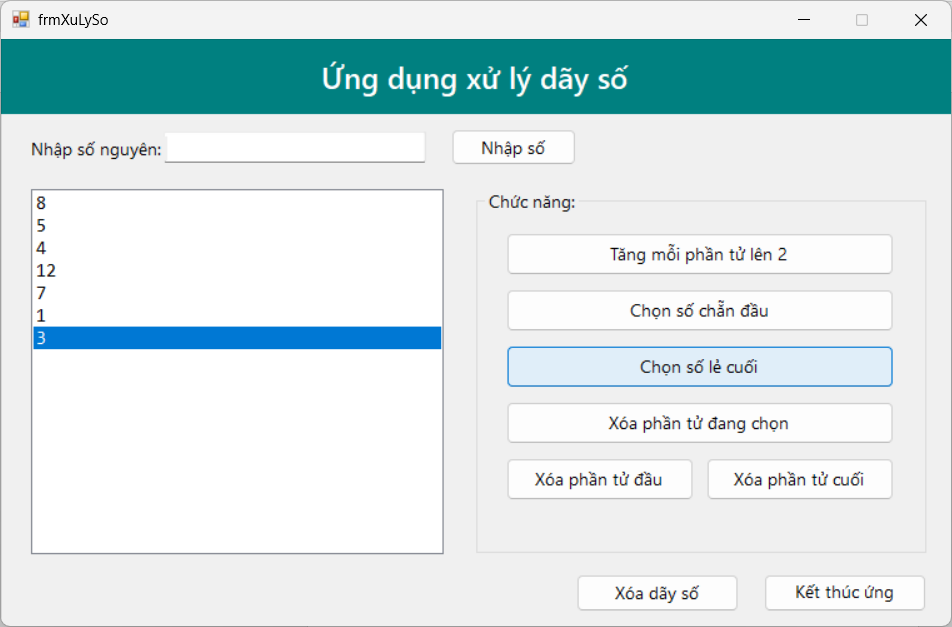
**Nhập số vào textbox**

****

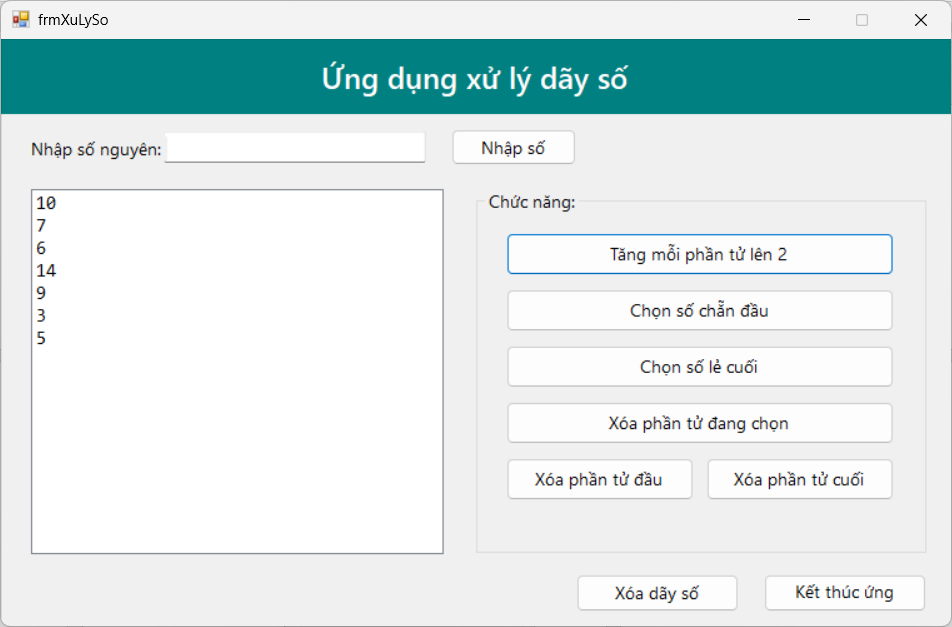
**Chọn số chẵn đầu**

****

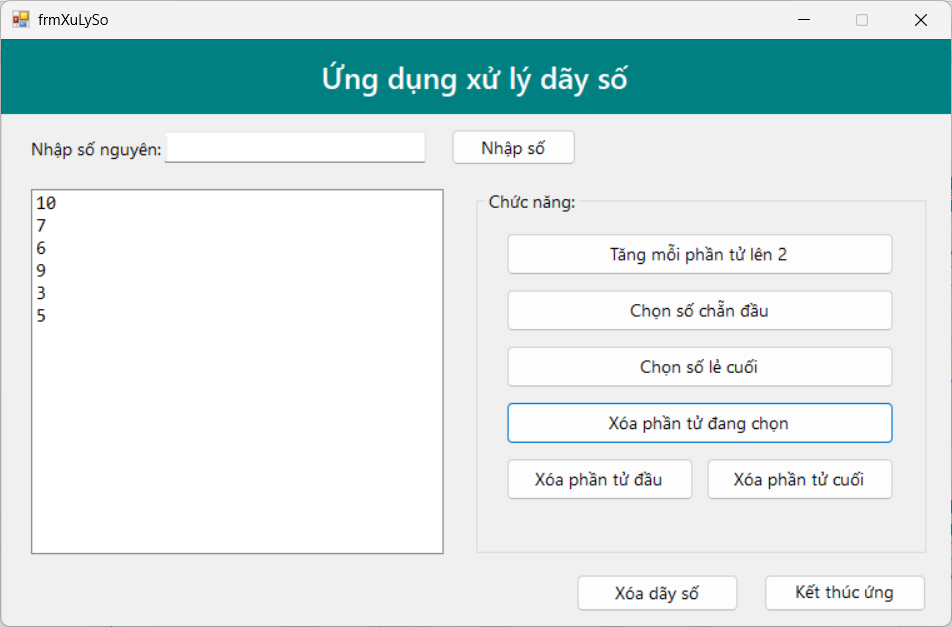
**Chọn số lẻ cuối**

****

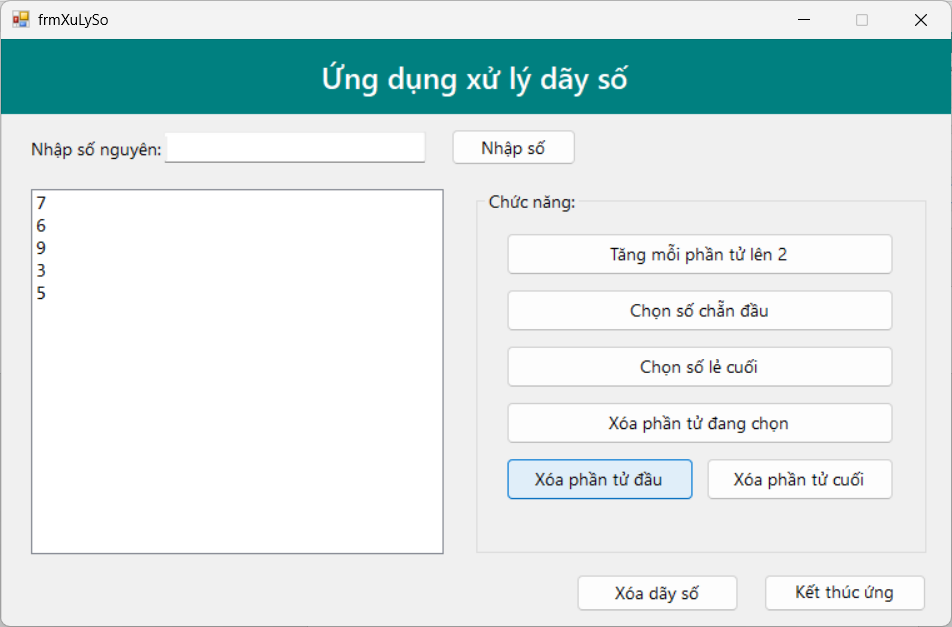
**Tăng mỗi phần tử lên 2 đơn vị**

****

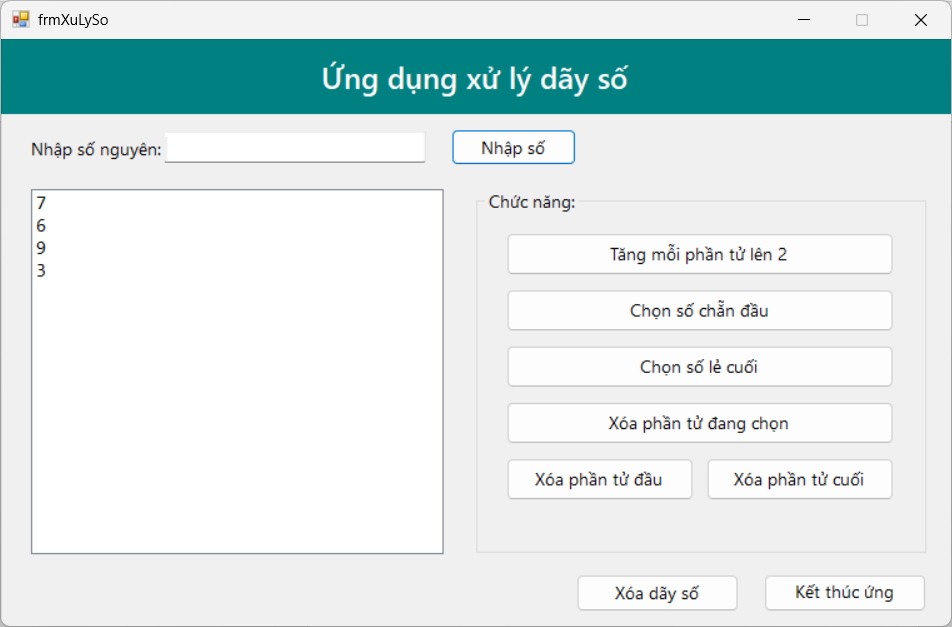
**Xóa phần tử 14**

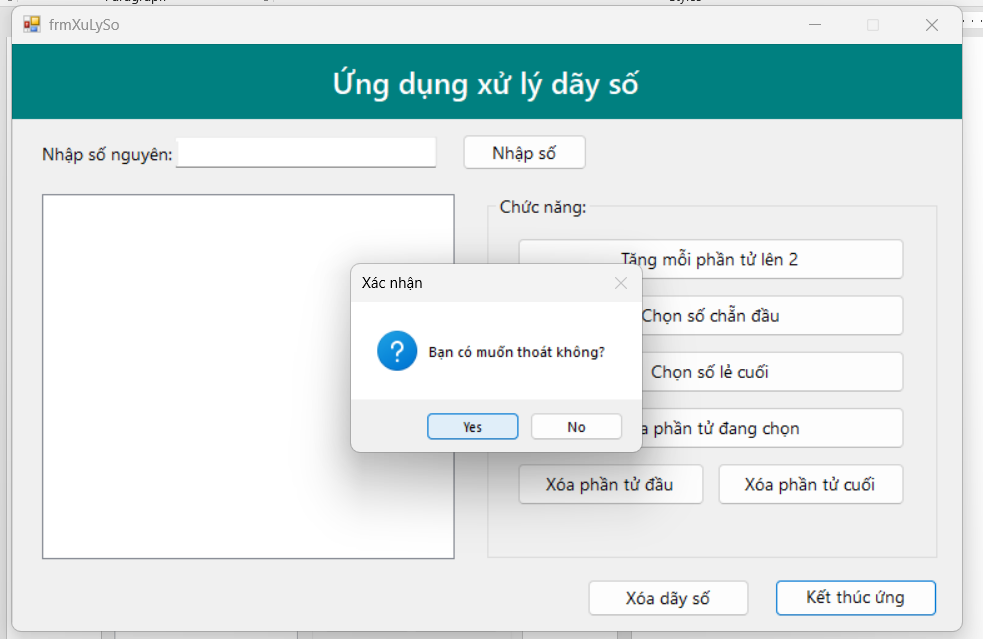
****

**Xóa phần tử đầu**

****

**Xóa phần tử cuối**

****

****

**Form.cs**

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;

namespace LamThaiHoa\_1150080053\_TH3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

lsbDaySo.Items.Clear();

}

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

{

int a;

string num = txtNhap.Text.Trim();

if (int.TryParse(num, out a))

{

lsbDaySo.Items.Add(a);

txtNhap.Clear();

txtNhap.Focus();

}

else

{

MessageBox.Show("Không phải số nguyên! Hãy nhập lại!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtNhap.Clear();

txtNhap.Focus();

}

}

private void btnTang2\_Click(object sender, EventArgs e)

{

if (lsbDaySo.Items.Count == 0)

{

MessageBox.Show("Dãy số đang trống. Vui lòng nhập dữ liệu!",

"Thông báo!", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtNhap.Focus();

}

else

{

for (int i = 0; i < lsbDaySo.Items.Count; i++)

{

lsbDaySo.Items[i] = (int)lsbDaySo.Items[i] + 2;

}

}

}

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

{

if (lsbDaySo.Items.Count == 0)

{

MessageBox.Show("Dãy số đang trống!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

for (int i = 0; i < lsbDaySo.Items.Count; i++)

{

if ((int)lsbDaySo.Items[i] % 2 == 0)

{

lsbDaySo.SelectedIndex = i;

break;

}

}

}

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

{

if (lsbDaySo.Items.Count == 0)

{

MessageBox.Show("Dãy số đang trống!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

for (int i = lsbDaySo.Items.Count - 1; i >= 0; i--)

{

if ((int)lsbDaySo.Items[i] % 2 != 0)

{

lsbDaySo.SelectedIndex = i;

break;

}

}

}

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

{

if (lsbDaySo.Items.Count == 0)

{

MessageBox.Show("Dãy số đang trống!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (lsbDaySo.SelectedIndex == -1)

{

MessageBox.Show("Bạn chưa chọn số cần xóa!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

while (lsbDaySo.SelectedIndex != -1)

{

lsbDaySo.Items.RemoveAt(lsbDaySo.SelectedIndex);

}

}

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

{

if (lsbDaySo.Items.Count > 0)

lsbDaySo.Items.RemoveAt(0);

}

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

{

if (lsbDaySo.Items.Count > 0)

lsbDaySo.Items.RemoveAt(lsbDaySo.Items.Count - 1);

}

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

{

lsbDaySo.Items.Clear();

}

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

{

DialogResult result = MessageBox.Show("Bạn có muốn thoát không?", "Xác nhận",

MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

Close();

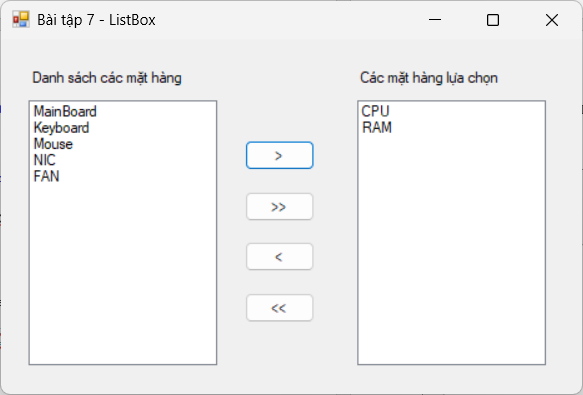
}

}

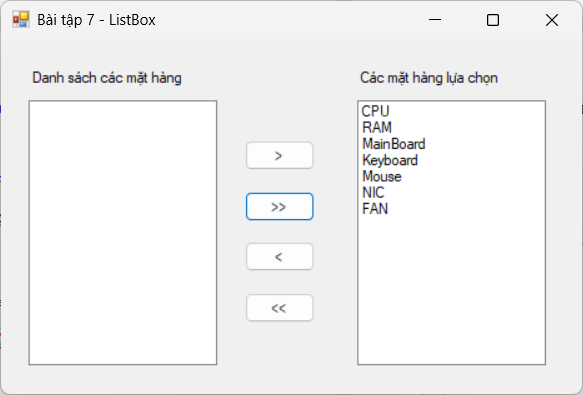
}

## Áp dụng 4: làm bài tập 7 tại Link.

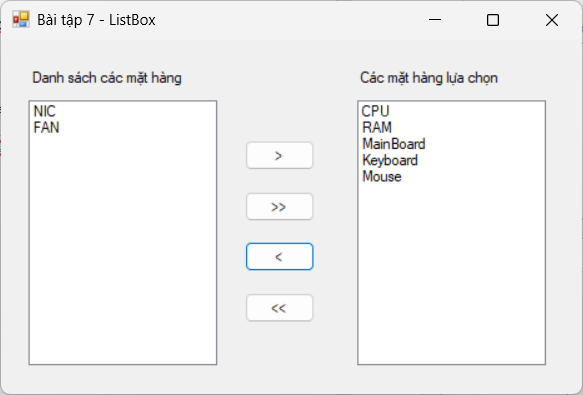
Chuyển CPU và RAM qua



Chuyển tất cả qua



Chuyển lại NIC và FAN



Chuyển lại tất cả

