1150080159

Nguyễn Thị Anh Thư

**LAB 5 (2)**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

# **Thực hành 1:**

using System;

using System.Globalization;

using System.Windows.Forms;

namespace WindowsFormsApp1\_LAB5.\_2\_

{

public partial class Form1 : Form

{

private readonly CultureInfo \_ci = CultureInfo.InvariantCulture;

public Form1()

{

InitializeComponent(); // do Designer sinh

if (txtKetQua != null) txtKetQua.ReadOnly = true;

// Nếu trong Designer CHƯA gán Click cho các nút,

// bạn có thể mở comment 6 dòng bên dưới:

// btnCong.Click += btnCong\_Click;

// btnTru.Click += btnTru\_Click;

// btnNhan.Click += btnNhan\_Click;

// btnChia.Click += btnChia\_Click;

// btnXoa.Click += btnXoa\_Click;

// btnThoat.Click += btnThoat\_Click;

}

// --- Helpers ---

private bool TryGetInputs(out double a, out double b)

{

// chấp nhận cả dấu phẩy hoặc chấm

string sa = txtA.Text.Trim().Replace(',', '.');

string sb = txtB.Text.Trim().Replace(',', '.');

bool okA = double.TryParse(sa, NumberStyles.Float, \_ci, out a);

bool okB = double.TryParse(sb, NumberStyles.Float, \_ci, out b);

if (!okA || !okB)

{

MessageBox.Show("Vui lòng nhập số thực hợp lệ cho a và b!",

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

return false;

}

return true;

}

// --- Event handlers (hãy gán Click của các nút về các hàm này) ---

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

{

if (TryGetInputs(out double a, out double b))

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

}

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

{

if (TryGetInputs(out double a, out double b))

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

}

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

{

if (TryGetInputs(out double a, out double b))

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

}

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

{

if (!TryGetInputs(out double a, out double b)) return;

if (b == 0)

{

MessageBox.Show("Mẫu số không được bằng 0. Nhập lại!",

"Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtB.SelectAll();

txtB.Focus();

return;

}

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

}

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

{

txtA.Clear();

txtB.Clear();

txtKetQua.Clear();

txtA.Focus();

}

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

{

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

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

Close();

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.

## **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:

using System;

using System.Windows.Forms;

namespace Apdung1

{

public partial class Form1 : Form // <-- partial + : Form

{

public Form1()

{

InitializeComponent(); // do Designer sinh

}

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

{

if (!TryGetInputs(out int a, out int b)) return;

if (!rdoUSCLN.Checked && !rdoBSCNN.Checked)

{

MessageBox.Show("Vui lòng chọn USCLN hoặc BSCNN.",

"Thiếu lựa chọn", MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

if (rdoUSCLN.Checked)

txtKetQua.Text = Gcd(a, b).ToString();

else

txtKetQua.Text = Lcm(a, b).ToString();

}

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

{

if (MessageBox.Show("Thoát chương trình?", "Xác nhận",

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

Close();

}

// ===== Helpers =====

private bool TryGetInputs(out int a, out int b)

{

bool okA = int.TryParse(txtA.Text.Trim(), out a);

bool okB = int.TryParse(txtB.Text.Trim(), out b);

if (!okA || !okB)

{

MessageBox.Show("Nhập hai số NGUYÊN hợp lệ (a, b).",

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

return false;

}

return true;

}

private int Gcd(int x, int y) // Euclid, cho phép số âm

{

x = Math.Abs(x); y = Math.Abs(y);

while (y != 0)

{

int r = x % y;

x = y; y = r;

}

return x; // nếu a=b=0 => 0

}

private long Lcm(int x, int y) // BCNN = |a\*b|/UCLN, nếu có 0 => 0

{

long ax = Math.Abs((long)x), ay = Math.Abs((long)y);

if (ax == 0 || ay == 0) return 0;

int g = Gcd(x, y);

return (ax / g) \* ay; // chia trước tránh tràn

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.

## **Áp dụng 2:** Xây dựng ứng dụng mô phỏng thiết bị Security Panel theo yêu cầu:

using System;

using System.Collections.Generic;

using System.Media;

using System.Windows.Forms;

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

namespace Apdung2

{

public partial class Form1 : Form

{

// map password -> tên nhóm

private readonly Dictionary<string, string> \_passToGroup = 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();

}

/\* ====== EVENTS ====== \*/

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

{

var btn = (Button)sender;

// giới hạn 4 số cho pass (theo ví dụ)

if (txtPassword.TextLength < 4)

txtPassword.Text += btn.Text;

}

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

{

txtPassword.Clear();

txtPassword.Focus();

}

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

{

string pass = txtPassword.Text.Trim();

string group = ResolveGroup(pass);

string result = group == null ? "Từ chối!" : "Chấp nhận!";

if (group == null) group = "Không có";

// log

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

// feedback nhẹ

if (result == "Chấp nhận!")

SystemSounds.Asterisk.Play();

else

SystemSounds.Hand.Play();

txtPassword.Clear();

txtPassword.Focus();

}

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

{

SystemSounds.Beep.Play();

MessageBox.Show("Chuông báo kêu!", "RING", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

/\* ====== HELPERS ====== \*/

private string ResolveGroup(string pass)

{

if (string.IsNullOrWhiteSpace(pass)) return null;

return \_passToGroup.TryGetValue(pass, out var grp) ? grp : null;

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.

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

using System;

using System.ComponentModel;

using System.Globalization;

using System.Windows.Forms;

namespace ThucHanh2

{

public partial class Form1 : Form

{

// Đơn giá

private const int GIA\_LAY\_CAO\_RANG = 50\_000; // /2 hàm (trọn gói)

private const int GIA\_TAY\_TRANG = 100\_000; // /2 hàm (trọn gói)

private const int GIA\_HAN\_RANG = 100\_000; // /1 răng

private const int GIA\_BE\_RANG = 10\_000; // /1 răng

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

private readonly CultureInfo \_vi = new CultureInfo("vi-VN");

public Form1()

{

InitializeComponent();

// gán sự kiện cho nút (phòng khi Designer chưa gán)

btnTinhTien.Click += btnTinhTien\_Click;

btnThoat.Click += btnThoat\_Click;

// bật/tắt input số lượng theo checkbox

chkHanRang.CheckedChanged += (\_, \_\_) => numHanRang.Enabled = chkHanRang.Checked;

chkBeRang.CheckedChanged += (\_, \_\_) => numBeRang.Enabled = chkBeRang.Checked;

chkBocRang.CheckedChanged += (\_, \_\_) => numBocRang.Enabled = chkBocRang.Checked;

// set trạng thái ban đầu theo tình trạng tick hiện tại (đúng hơn là ép false)

numHanRang.Enabled = chkHanRang.Checked;

numBeRang.Enabled = chkBeRang.Checked;

numBocRang.Enabled = chkBocRang.Checked;

// validate tên KH

txtTenKH.Validating += txtTenKH\_Validating;

// Enter để tính nhanh

this.AcceptButton = btnTinhTien;

}

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

{

// kiểm tra tên KH (và các control có Validate nếu có)

if (!ValidateChildren(ValidationConstraints.Enabled))

return;

long thanhTien = 0;

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

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

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

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

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

txtThanhTien.Text = thanhTien.ToString("N0", \_vi) + " đ";

}

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

{

if (MessageBox.Show("Thoát chương trình?", "Xác nhận",

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

Close();

}

// === Validation: tên khách hàng không được trống ===

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

{

if (string.IsNullOrWhiteSpace(txtTenKH.Text))

{

e.Cancel = true; // chặn rời control

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

}

else

{

e.Cancel = false;

err.SetError(txtTenKH, null);

}

}

}

}

**A screenshot of a computer

AI-generated content may be incorrect.**

## **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.

using System;

using System.ComponentModel;

using System.Windows.Forms;

namespace Apdung3 // <--- đổi cho khớp project

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

// gán sự kiện (phòng khi Designer chưa gán)

btnLogin.Click += btnLogin\_Click;

btnCancel.Click += (\_, \_\_) => this.Close();

// Enter = Đăng nhập

this.AcceptButton = btnLogin;

// checkbox hiện/ẩn mật khẩu

chkShowPass.CheckedChanged += (\_, \_\_) =>

txtPassword.UseSystemPasswordChar = !chkShowPass.Checked;

// mặc định ẩn mật khẩu

txtPassword.UseSystemPasswordChar = true;

// bật validate

txtUser.Validating += txt\_Validating\_NotEmpty;

txtPassword.Validating += txt\_Validating\_NotEmpty;

}

private void txt\_Validating\_NotEmpty(object sender, CancelEventArgs e)

{

var tb = (TextBox)sender;

if (string.IsNullOrWhiteSpace(tb.Text))

{

e.Cancel = true;

err.SetError(tb, "Không được để trống!");

}

else

{

e.Cancel = false;

err.SetError(tb, null);

}

}

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

{

// kiểm tra 2 ô đều hợp lệ

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

// TODO: ở đây mày có thể check tài khoản thực tế

// ví dụ: if (txtUser.Text == "admin" && txtPassword.Text == "123")

// { DialogResult = DialogResult.OK; Close(); } else { MessageBox.Show(...); }

MessageBox.Show(

"Đăng nhập thành công!",

"Thông báo",

MessageBoxButtons.OK,

MessageBoxIcon.Information

);

DialogResult = DialogResult.OK;

Close();

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.

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

using System;

using System.Windows.Forms;

namespace ThucHanh3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

lsbDaySo.Items.Clear();

}

// Nhập số vào ListBox

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

{

int a;

string num = txtInput.Text.Trim();

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

{

lsbDaySo.Items.Add(a);

txtInput.Clear();

txtInput.Focus();

}

else

{

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

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtInput.Clear();

txtInput.Focus();

}

}

// Tăng mỗi phần tử lên 2

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

{

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

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

}

// Chọn số chẵn đầu tiên

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

{

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

{

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

{

lsbDaySo.SelectedIndex = i;

break;

}

}

}

// Chọn số lẻ cuối cùng

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

{

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

{

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

{

lsbDaySo.SelectedIndex = i;

break;

}

}

}

// Xóa phần tử đang chọn

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

{

if (lsbDaySo.SelectedIndex != -1)

lsbDaySo.Items.RemoveAt(lsbDaySo.SelectedIndex);

else

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

}

// Xóa phần tử đầu

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

{

if (lsbDaySo.Items.Count > 0)

lsbDaySo.Items.RemoveAt(0);

}

// Xóa phần tử cuối

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

{

if (lsbDaySo.Items.Count > 0)

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

}

// Xóa dãy số

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

{

lsbDaySo.Items.Clear();

}

// Thoát ứng dụng

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

{

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

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

Close();

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.

## **Áp dụng 4:** làm bài tập 7

using System;

using System.Collections.Generic;

using System.Linq;

using System.Windows.Forms;

namespace Apdung4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

// seed dữ liệu bên trái

lstMatHang.Items.AddRange(new object[]

{

"CPU", "MainBoard", "RAM", "Keyboard", "Mouse", "NIC", "FAN", "SSD", "HDD", "VGA"

});

// gán sự kiện nút

btnRight.Click += (\_, \_\_) => MoveSelected(lstMatHang, lstDaChon);

btnRightAll.Click += (\_, \_\_) => MoveAll(lstMatHang, lstDaChon);

btnLeft.Click += (\_, \_\_) => MoveSelected(lstDaChon, lstMatHang);

btnLeftAll.Click += (\_, \_\_) => MoveAll(lstDaChon, lstMatHang);

// double-click để chuyển nhanh

lstMatHang.DoubleClick += (\_, \_\_) => MoveSelected(lstMatHang, lstDaChon);

lstDaChon.DoubleClick += (\_, \_\_) => MoveSelected(lstDaChon, lstMatHang);

}

private void MoveSelected(ListBox from, ListBox to)

{

if (from.SelectedItems.Count == 0) return;

// copy danh sách chọn (tránh sửa collection khi đang duyệt)

List<object> picked = from.SelectedItems.Cast<object>().ToList();

foreach (var item in picked)

{

to.Items.Add(item);

from.Items.Remove(item);

}

}

private void MoveAll(ListBox from, ListBox to)

{

if (from.Items.Count == 0) return;

to.Items.AddRange(from.Items.Cast<object>().ToArray());

from.Items.Clear();

}

}

}

A screenshot of a computer

AI-generated content may be incorrect.