Họ tên: Phạm Gia Bảo

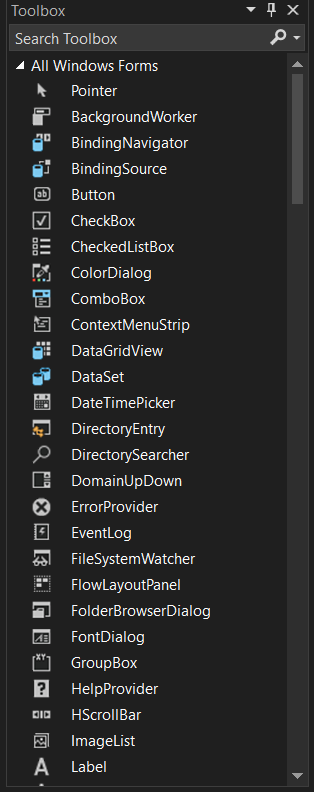
Lớp: 11\_ĐH\_CNPM1

MSSV: 1150080043

Bài làm

Thực hành 1:

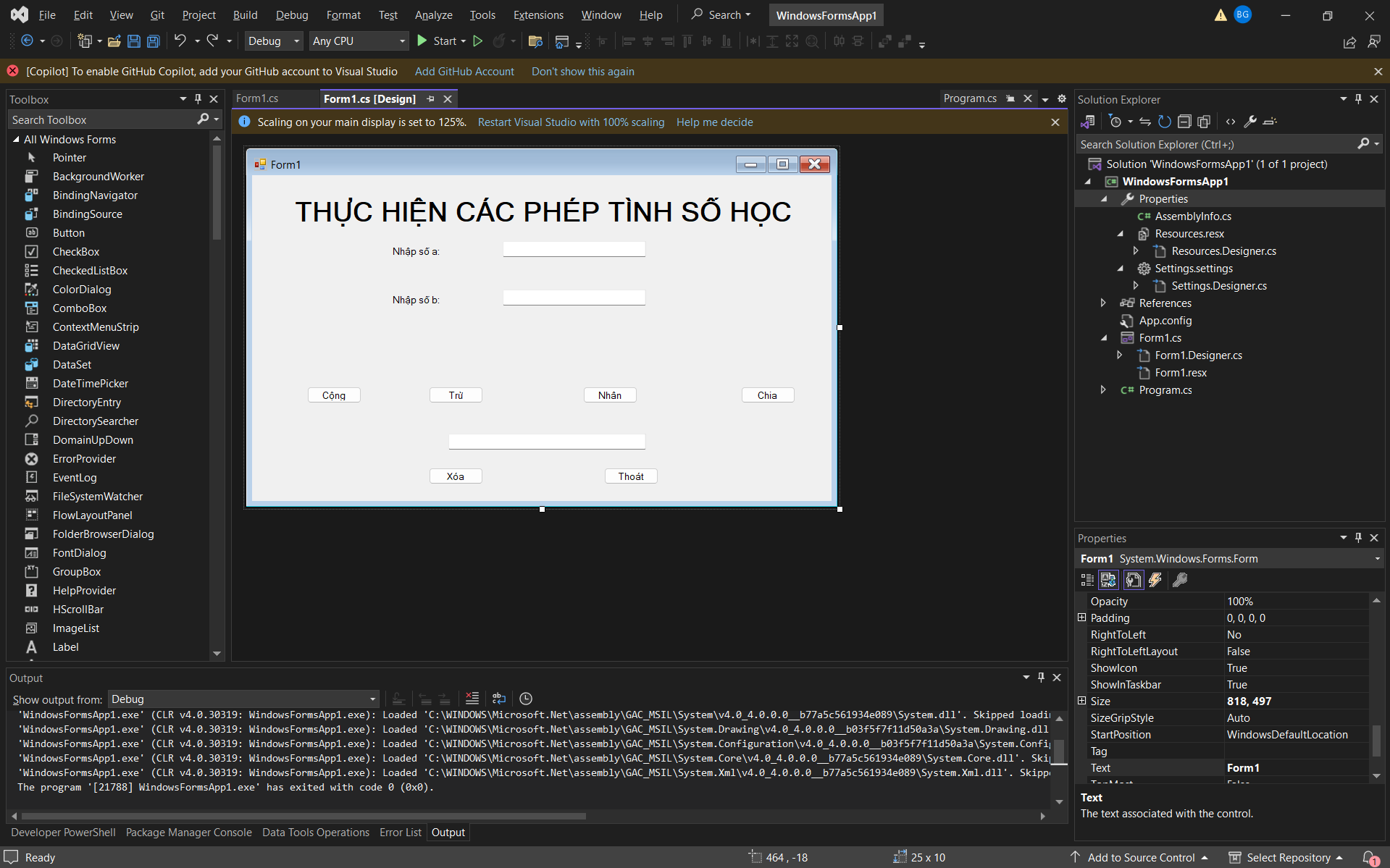
Bước 1: Nhấn vào toolbox hoặc nhấn ctrl + alt + x



Bước 2: Chọn các biến để nhập biến như sau

* TextBox1 cho phép nhập a: txtA
* TextBox2 cho phép nhập b: txtB
* TextBox3 cho phép hiển thị kết quả phép tính: txtKetQua
* Button1 phép toán cộng: btnCong
* Button2 phép toán trừ: btnTru
* Button3 phép toán nhân: btnNhan
* Button4 phép toán chia: btnChia
* Button5 phép toán xóa dữ liệu: btnXoa
* Button6 phép toán thoát chương trình: btnThoat
* Label1 cho phép đặt tên: Nhập số a:
* Label2 cho phép đặt tên: Nhập số b:
* Label3 cho phép đặt tên: Kết quả:
* Label4 cho phép đặt tên: THỰC HIỆN CÁC PHÉP TÍNH SỐ HỌC

Bước 3: Chỉnh các biến theo thiết kế form



Bước 4: Nhấn vào tab Form1.cs để nhập code như sau:

using System;

using System.Globalization;

using System.Windows.Forms;

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

btnCong.Click += btnCong\_Click;

btnChia.Click += btnChia\_Click;

btnThoat.Click += btnThoat\_Click;

btnTru.Click += btnTru\_Click;

btnNhan.Click += btnNhan\_Click;

btnXoa.Click += btnXoa\_Click;

}

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

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

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

{

a = 0; b = 0;

var style = NumberStyles.Float | NumberStyles.AllowThousands;

var culture = CultureInfo.CurrentCulture;

string sa = txtA.Text.Trim();

string sb = txtB.Text.Trim();

if (!double.TryParse(sa, style, culture, out a))

{

MessageBox.Show("Giá trị A không hợp lệ. Vui lòng nhập số.", "Thông báo lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtA.Focus();

txtA.SelectAll();

return false;

}

if (!double.TryParse(sb, style, culture, out b))

{

MessageBox.Show("Giá trị B không hợp lệ. Vui lòng nhập số.", "Thông báo lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtB.Focus();

txtB.SelectAll();

return false;

}

return true;

}

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

{

try

{

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

double tong = a + b;

txtKetQua.Text = tong.ToString(CultureInfo.CurrentCulture);

}

catch (Exception ex)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

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

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

}

catch (Exception ex)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

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

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

}

catch (Exception ex)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

try

{

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

if (b == 0)

{

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

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

txtB.Clear();

txtB.Focus();

return;

}

double kq = a / b;

txtKetQua.Text = kq.ToString(CultureInfo.CurrentCulture);

}

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)

{

var result = MessageBox.Show(

"Bạn có thực sự muốn thoát không?",

"Xác nhận thoát",

MessageBoxButtons.YesNo,

MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

this.Close();

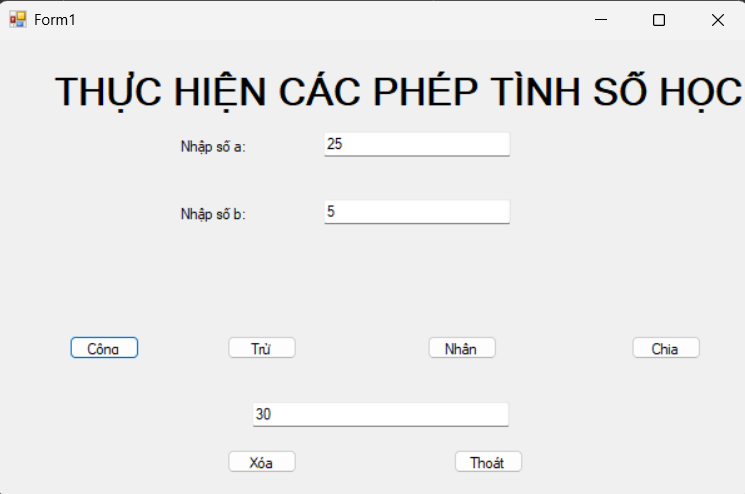
}

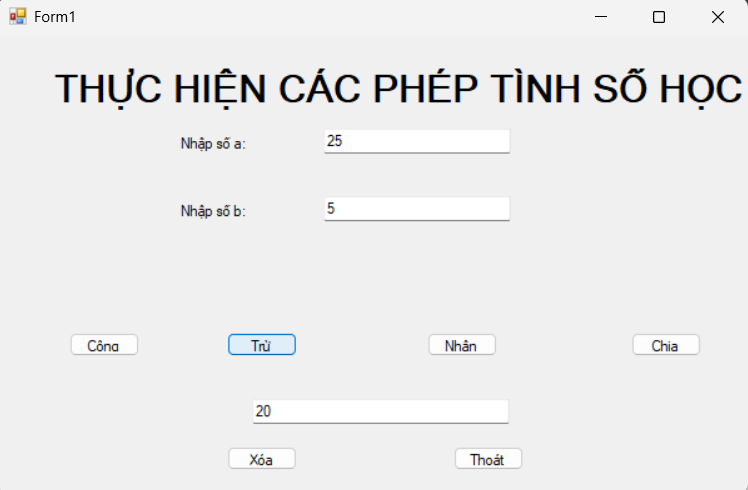
}

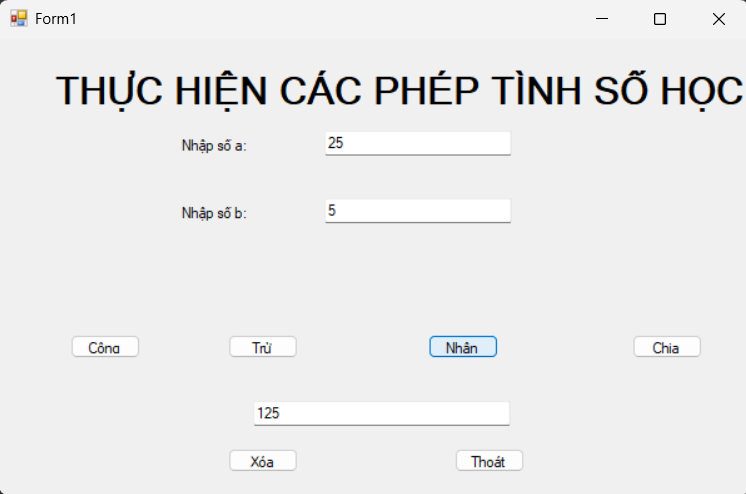
}

}

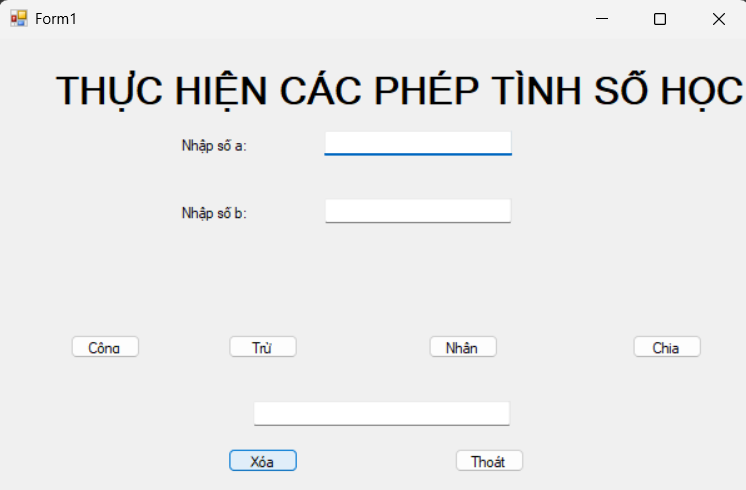
Kết quả:

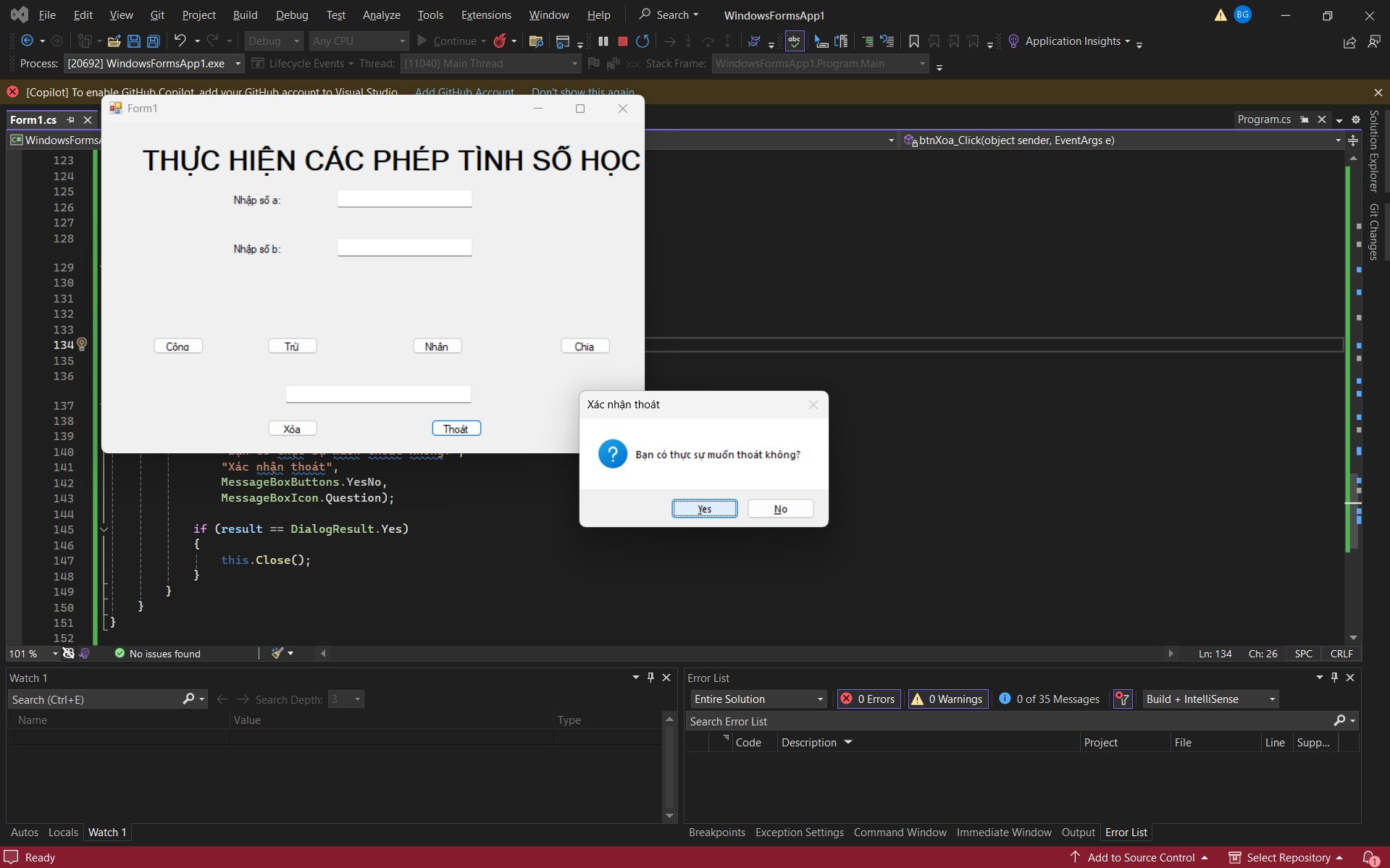










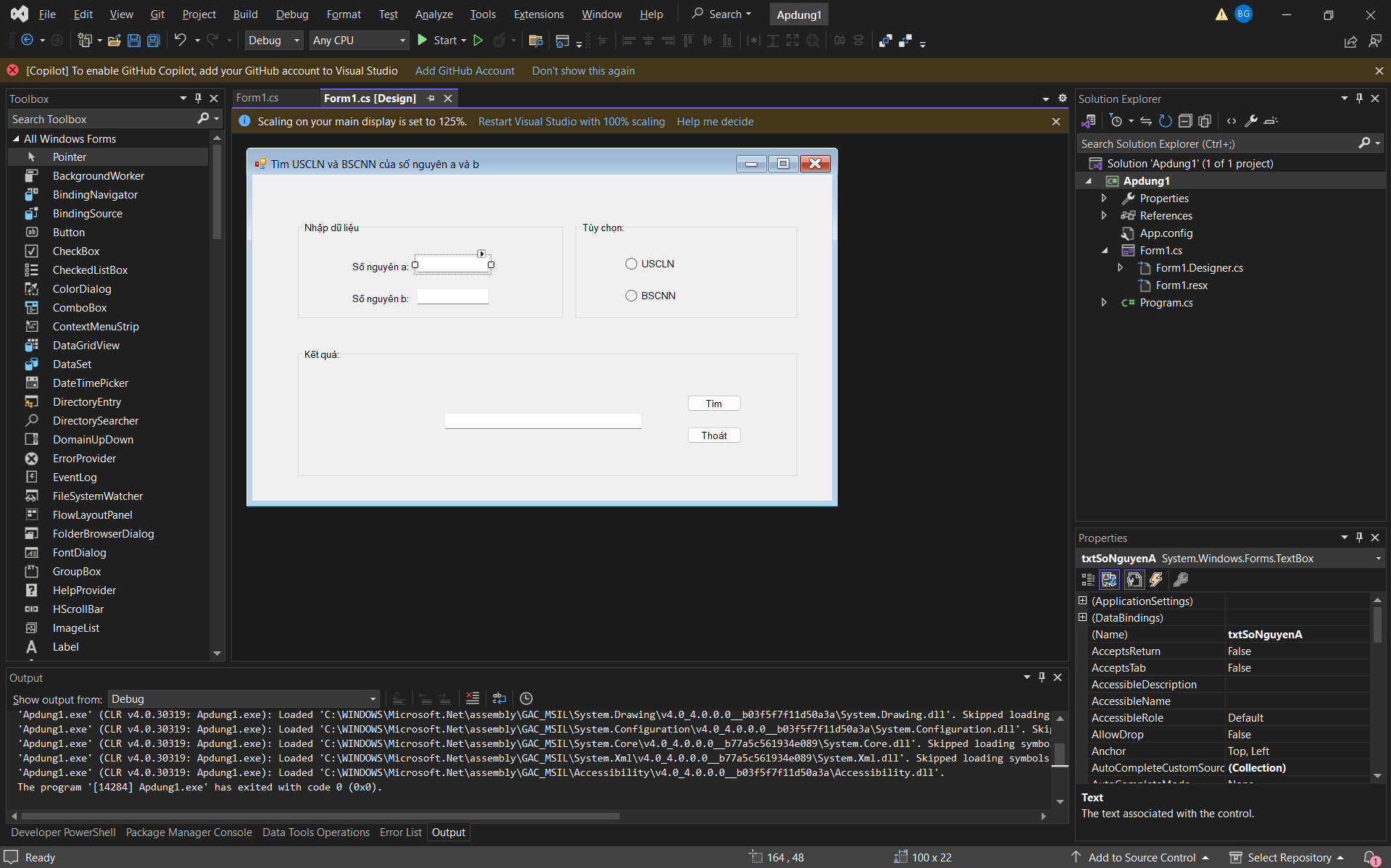


Áp dụng 1:

Bước 1: Nhấn toolbox và chọn để nhập tên như sau

* GroupBox1 cho phép nhập text: Nhập dữ liệu
* GroupBox2 cho phép nhập text: Tùy chọn:
* GroupBox3 cho phép nhập text: Kết quả:
* TextBox1 cho phép nhập name: txtSoNguyenA và text: Số nguyên a:
* TextBox2 cho phép nhập name: txtSoNguyenB và text: Số nguyên b:
* TextBox3 cho phép nhập name: txtKetQua
* RadioButton1 cho phép nhập name: radioButtonUSCLN và text: USCLN
* RadioButton2 cho phép nhập name: radioButtonBSCNN và text: BSCNN
* Button1 cho phép nhập name: btTim và text: Tìm
* Button2 cho phép nhập name: btThoat và text: Thoát

Bước 2: Chỉnh các biến theo thiết kế form



Bước 3: Nhấn vào tab Form1.cs để nhập code như sau:

using System;

using System.Globalization;

using System.Windows.Forms;

namespace Apdung1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

btTim.Click += btTim\_Click;

btThoat.Click += btThoat\_Click;

txtKetQua.ReadOnly = true;

}

private long USCLN(long a, long b)

{

a = Math.Abs(a);

b = Math.Abs(b);

if (a == 0) return b;

if (b == 0) return a;

while (b != 0)

{

long r = a % b;

a = b;

b = r;

}

return a;

}

private long BSCNN(long a, long b)

{

a = Math.Abs(a);

b = Math.Abs(b);

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

long gcd = USCLN(a, b);

return (a / gcd) \* b;

}

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

{

a = 0; b = 0;

if (!long.TryParse(txtSoNguyenA.Text.Trim(), out a))

{

MessageBox.Show("Số nguyên a không hợp lệ.", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtSoNguyenA.Focus(); txtSoNguyenA.SelectAll();

return false;

}

if (!long.TryParse(txtSoNguyenB.Text.Trim(), out b))

{

MessageBox.Show("Số nguyên b không hợp lệ.", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtSoNguyenB.Focus(); txtSoNguyenB.SelectAll();

return false;

}

return true;

}

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

{

try

{

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

if (radioButtonUSCLN.Checked)

{

long kq = USCLN(a, b);

txtKetQua.Text = $"USCLN({a}, {b}) = {kq}";

}

else

{

long kq = BSCNN(a, b);

txtKetQua.Text = $"BSCNN({a}, {b}) = {kq}";

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

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

"Xác nhận thoát",

MessageBoxButtons.YesNo,

MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

Close();

}

}

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

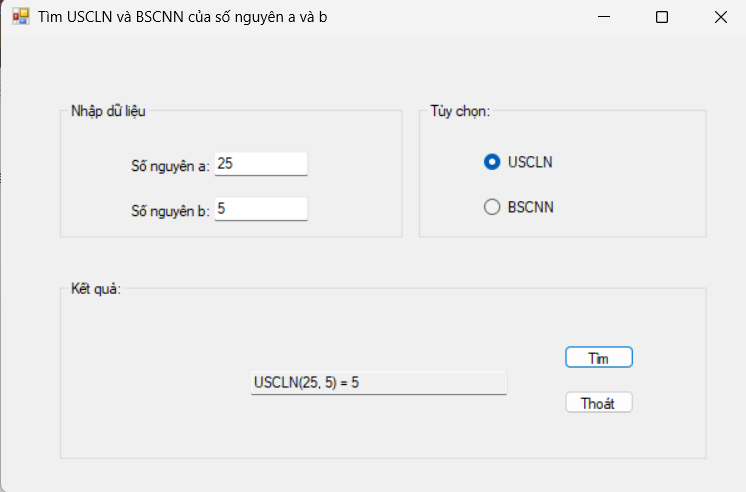
private void radioButton1\_CheckedChanged(object sender, EventArgs e) { }

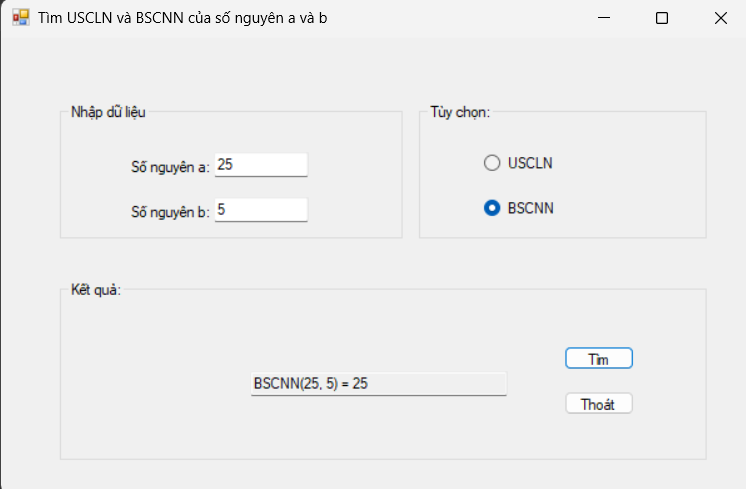
}

}

Kết quả:

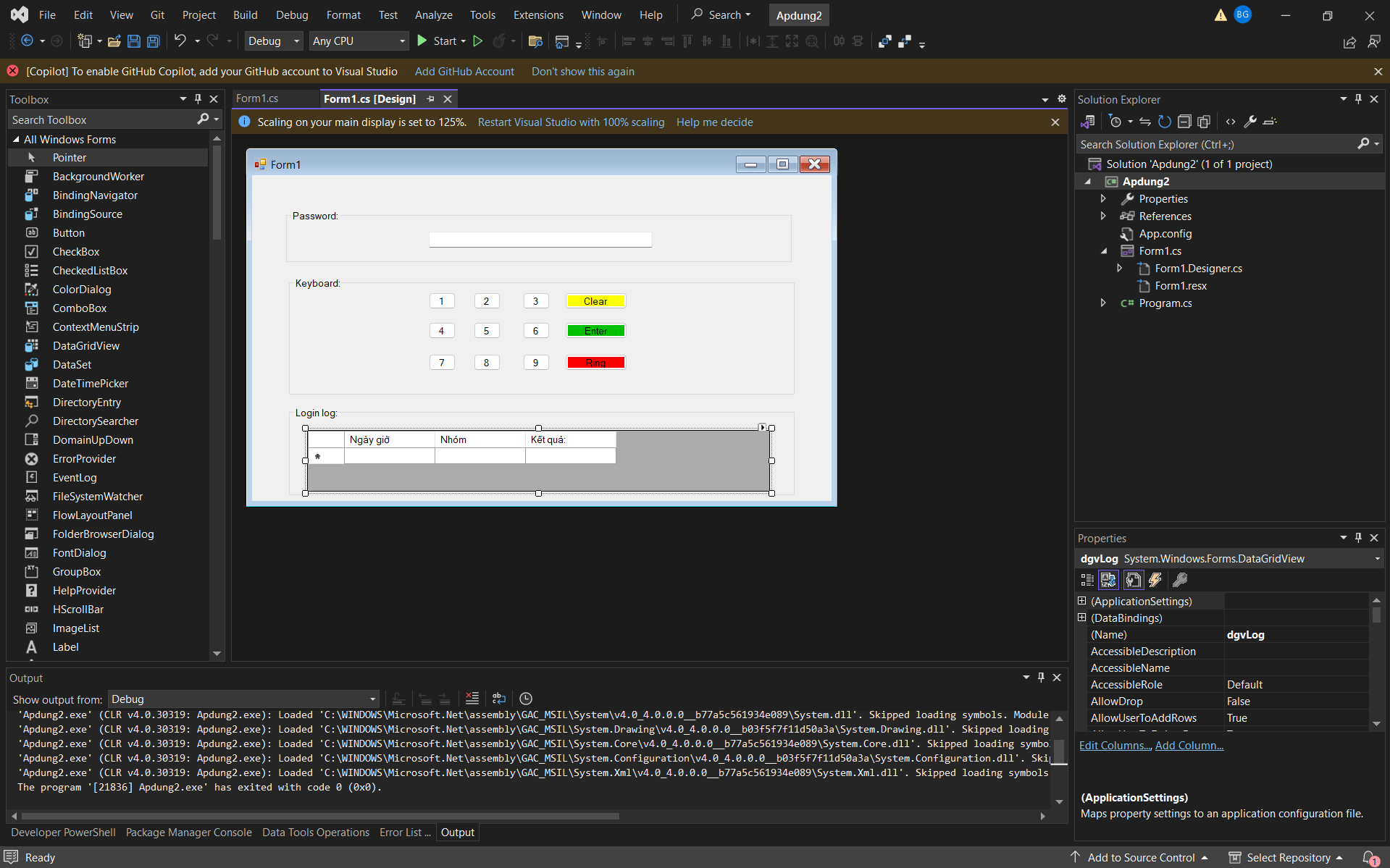
* Nhập số nguyên a và số nguyên b sau đó tích vào tùy chọn (USCLN hoặc BSCNN) và nhấn tìm để xem kết quả





Áp dụng 2:

Bước 1: Chỉnh các biến theo thiết kế form



Bước 2: Nhấn vào tab Form1.cs để nhập code như sau:

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

{

private readonly Dictionary<string, HashSet<string>> \_passwords =

new Dictionary<string, HashSet<string>>(StringComparer.OrdinalIgnoreCase)

{

["Phát triển công nghệ"] = new HashSet<string> { "1496", "2673" },

["Nghiên cứu viên"] = new HashSet<string> { "7462" },

["Thiết kế mô hình"] = new HashSet<string> { "8884", "3842", "3383" },

};

public Form1()

{

InitializeComponent();

btn1.Click += DigitButton\_Click;

btn2.Click += DigitButton\_Click;

btn3.Click += DigitButton\_Click;

btn4.Click += DigitButton\_Click;

btn5.Click += DigitButton\_Click;

btn6.Click += DigitButton\_Click;

btn7.Click += DigitButton\_Click;

btn8.Click += DigitButton\_Click;

btn9.Click += DigitButton\_Click;

btnClear.Click += btnClear\_Click;

btnEnter.Click += btnEnter\_Click;

btnRing.Click += btnRing\_Click;

SetupGrid();

txtPassword.ReadOnly = true;

}

private void SetupGrid()

{

dgvLog.AllowUserToAddRows = false;

dgvLog.AllowUserToDeleteRows = false;

dgvLog.ReadOnly = true;

dgvLog.SelectionMode = DataGridViewSelectionMode.FullRowSelect;

dgvLog.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill;

dgvLog.Columns.Clear();

dgvLog.Columns.Add(new DataGridViewTextBoxColumn

{

Name = "Ngaygio",

HeaderText = "Ngày giờ"

});

dgvLog.Columns.Add(new DataGridViewTextBoxColumn

{

Name = "Nhom",

HeaderText = "Nhóm"

});

dgvLog.Columns.Add(new DataGridViewTextBoxColumn

{

Name = "ketqua",

HeaderText = "Kết quả"

});

}

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

{

var btn = sender as Button;

if (btn == null) return;

if (txtPassword.Text.Length >= 4) return;

txtPassword.Text += btn.Text;

}

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

{

txtPassword.Clear();

txtPassword.Focus();

}

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

{

var pwd = txtPassword.Text.Trim();

if (pwd.Length == 0)

{

SystemSounds.Exclamation.Play();

MessageBox.Show("Chưa nhập mật khẩu!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

var (ok, group) = ValidatePassword(pwd);

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

ok ? group : "Không có",

ok ? "Chấp nhận" : "Từ chối");

if (ok)

{

SystemSounds.Asterisk.Play();

MessageBox.Show($"Chào mừng nhóm: {group}", "Được phép vào",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

{

SystemSounds.Hand.Play();

MessageBox.Show("Sai mật khẩu! Từ chối truy cập.", "Từ chối",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

txtPassword.Clear();

txtPassword.Focus();

}

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

{

SystemSounds.Beep.Play();

}

private (bool ok, string group) ValidatePassword(string pwd)

{

foreach (var kv in \_passwords)

{

if (kv.Value.Contains(pwd))

return (true, kv.Key);

}

return (false, string.Empty);

}

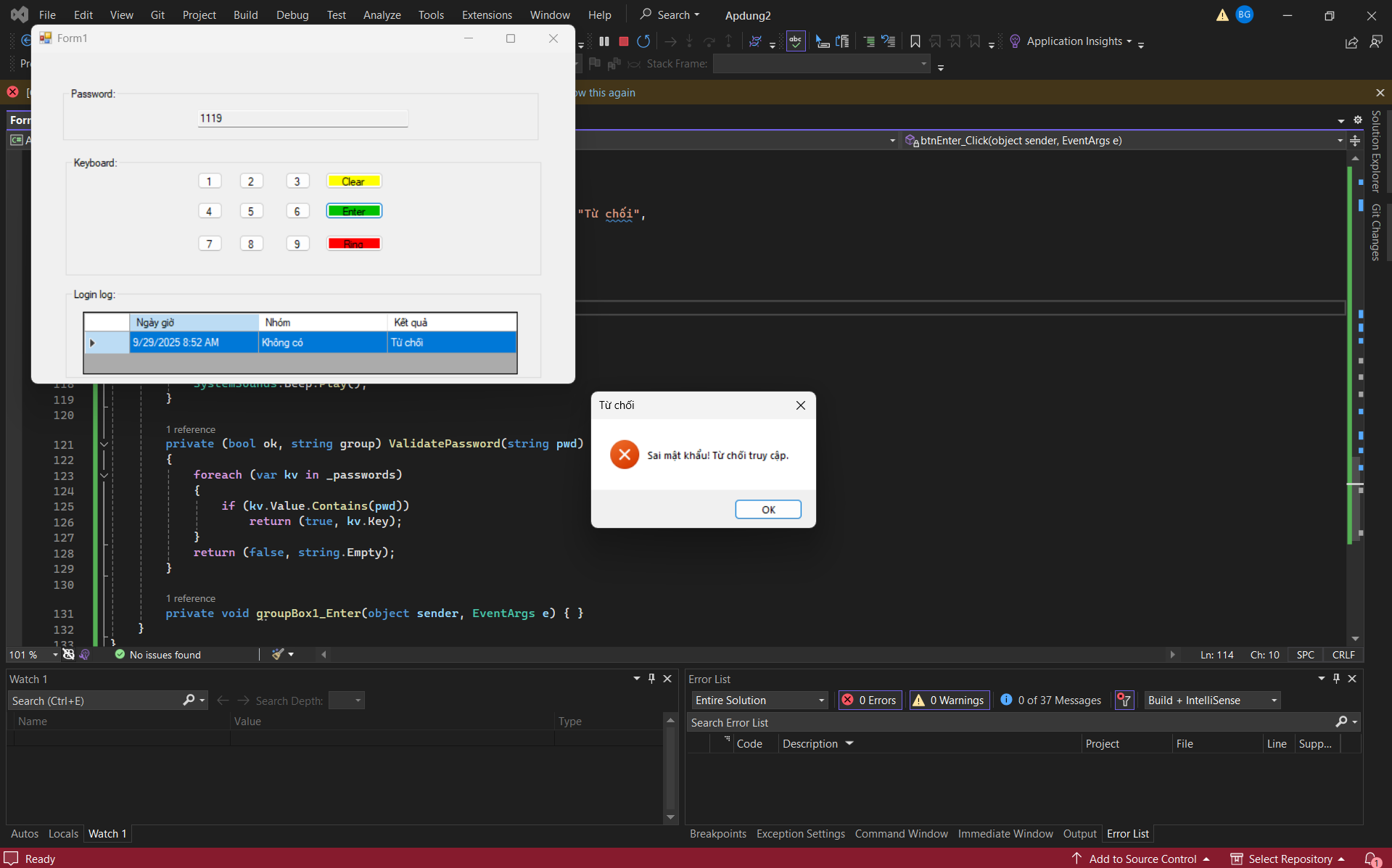
private void groupBox1\_Enter(object sender, EventArgs e) { }

}

}

Kết quả:

* Nhập mật khẩu khác sẽ hiện thông báo từ chối



* Nhập mật khẩu đúng sẽ hiện thông báo được phép vào

