

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM  
**KHOA CÔNG NGHỆ THÔNG TIN**



**Bài tập thực hành TUẦN 5**  
**Môn học: Phương pháp phát triển phần mềm hướng đối tượng**

**Giảng viên:** ThS. Phạm Trọng Huynh

**Sinh viên thực hiện:**

- Nguyễn Phú Thọ

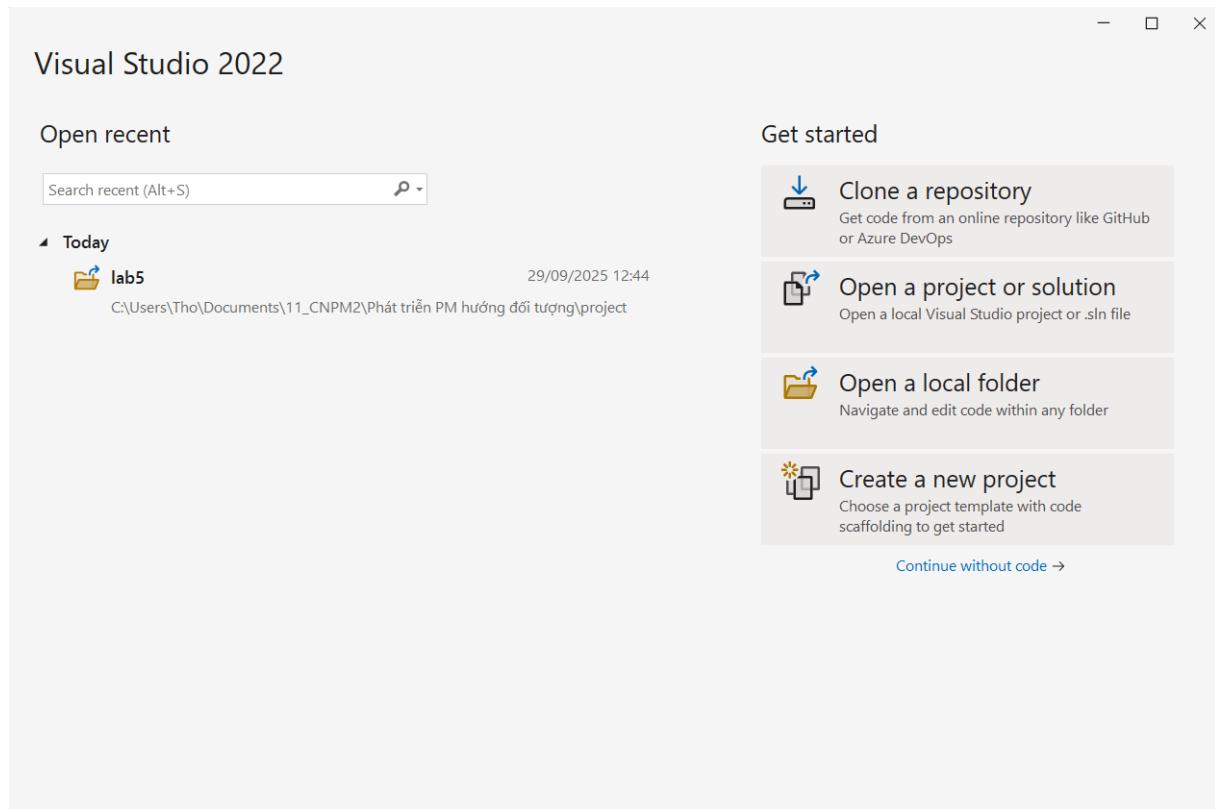
**Mã SV:** 1150080158

**Lớp:** 11\_DHCNPM\_2

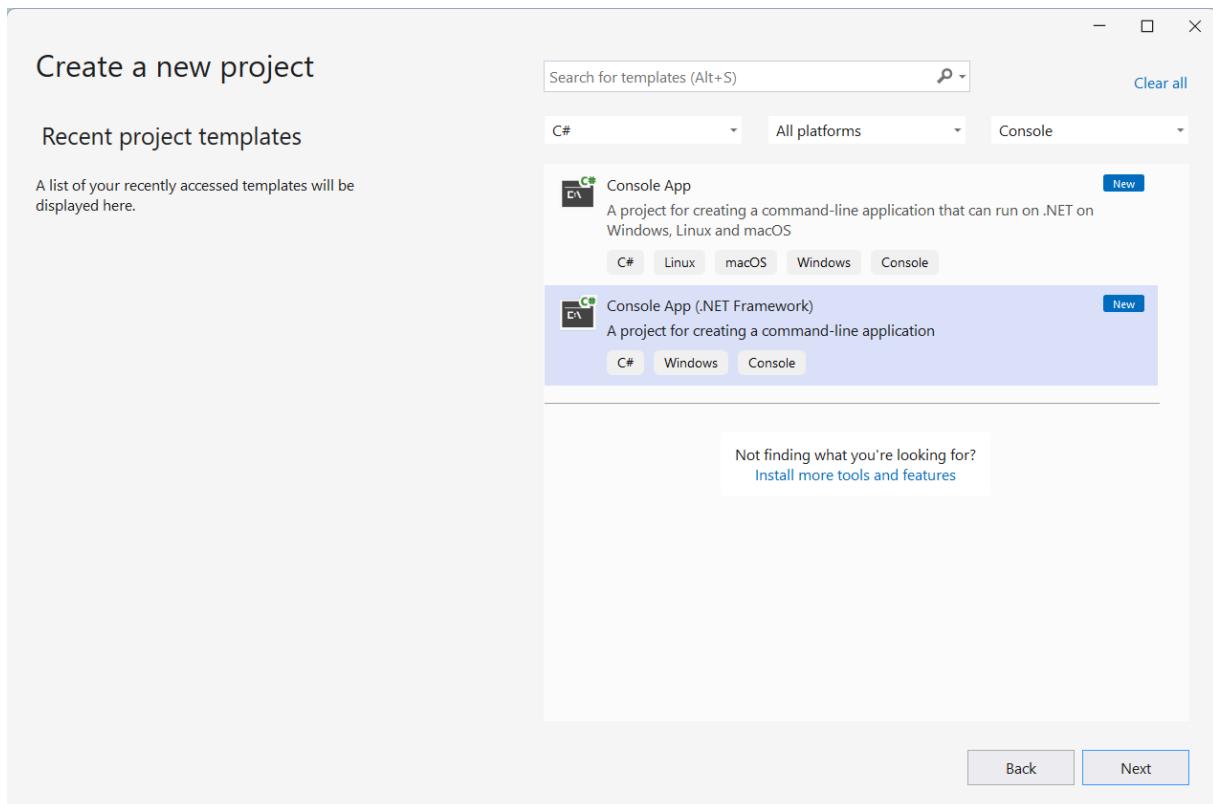
## LAB 1.

### 1. Tạo Project Console Application trên Microsoft visual studio 2022

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

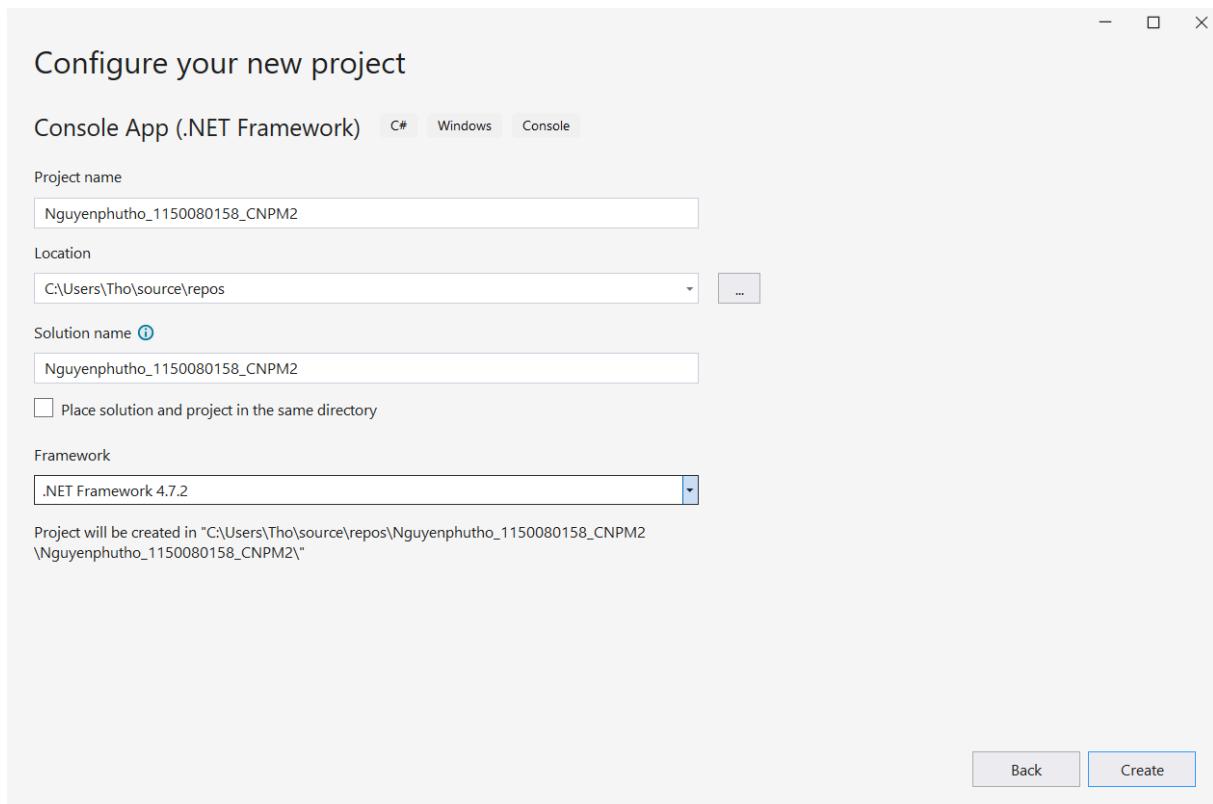


Bước 2: Chọn Create a new project trên màn hình khởi động. Trong màn hình tiếp theo, ở mục All languages → chọn C# trong danh sách ngôn ngữ lập trình.



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

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



Đây là kết quả!

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Nguyenphutho_1150080158_CNPBM2
{
    internal class Program
    {
        static void Main(string[] args)
        {
        }
    }
}

```

\* Phân tích đề bài:

- Vào: chiều dài a, chiều rộng b.
- Ra: Chu vi P, diện tích S.

{Chương trình này thực hiện nhập chiều dài a và chiều rộng b từ bàn phím. Tính và hiển thị chu vi và diện tích của hình chữ nhật}

### Program Tinh\_CV\_DT

#### 1. Nhập dữ liệu

Read(a, b);

#### 2. Tính chu vi và diện tích

P := (a + b) \* 2;

S:= a \* b;

#### 3. Hiển thị kết quả

Write(P, S);

End.

```

Microsoft Visual Studio Debug Console
Nhap chieu dai a: 5
Nhap chieu rong b: 3
Chu vi la: 16
Dien tich la: 15

C:\Users\Tho\source\repos\Nguyenphutho_1150080158_CNPBM2\Nguyenphutho_1150080158_CNPBM2\bin\Debug\Nguyenphutho_1150080158_CNPBM2.exe (process 106348) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

Code bai 1

using System;

```
namespace Nguyenphutho_1150080158_CNPM2
```

```
{
```

```
    internal static class Bai1_THO
```

```
{
```

```
    public static void Run()
```

```
{
```

```
        Console.Write("Nhap chieu dai a: ");
```

```
        double a = Convert.ToDouble(Console.ReadLine());
```

```
        Console.Write("Nhap chieu rong b: ");
```

```
        double b = Convert.ToDouble(Console.ReadLine());
```

```
        if (a <= 0 || b <= 0)
```

```
{
```

```
            Console.WriteLine("Chieu dai va chieu rong phai la so duong!");
```

```
            return;
```

```
}
```

```
        double P = 2 * (a + b);
```

```
        double S = a * b;
```

```
        Console.WriteLine("Chu vi la: " + P);
```

```
        Console.WriteLine("Dien tich la: " + S);
```

```
}
```

```
}
```

```
}
```

## **2. Biến là gì?**

- Biến là tên một ô nhớ trong bộ nhớ, dùng để chứa dữ liệu, giá trị của biến có thể thay đổi trong quá trình thực thi chương trình.
- Dữ liệu gồm 3 loại:
  - + Dữ liệu đầu vào.
  - + Dữ liệu đầu ra.
  - + Dữ liệu trung gian trong quá trình xử lý dữ liệu.
- Quy tắc đặt tên biến:
  - + Là một dãy các ký tự chữ cái (A-Z, a-z), chữ số (0-9), dấu gạch dưới.
  - + Không được bắt đầu bằng chữ số, được phép bắt đầu bằng dấu gạch dưới.
  - + Không được trùng với từ khóa trong C#.
  - + C# PHÂN BIỆT KÝ TỰ HOA VÀ THƯỜNG.

## **3. Kiểu dữ liệu**

C# là một ngôn ngữ lập trình kiểm soát chặt chẽ về kiểu dữ liệu. Nghĩa là bạn phải khai báo kiểu dữ liệu với mỗi một biến.

Bảng sau mô tả các kiểu dữ liệu được xây dựng sẵn:

## **4. Các toán tử số học**

- + Cộng
- Trừ
- \* Nhân
- / Chia
- % Chia lấy phần dư
  - ++i, i++ Tăng giá trị i lên 1 đơn vị
  - i, i-- Giảm giá trị i xuống 1 đơn vị

## **5. Chuyển đổi kiểu dữ liệu**

Có 2 dạng chuyển đổi kiểu dữ liệu là ngầm định (implicit) và tường minh (explicit). Chuyển đổi theo dạng ngầm định do trình biên dịch thực hiện. Còn dạng tường minh thì bạn ép kiểu từ kiểu này sang kiểu khác.

### **5.1. Chuyển đổi ngầm (implicit conversion)**

Chuyển đổi ngầm định sẽ được tự động thực hiện và bạn được đảm bảo rằng dữ

liệu sẽ được không mất mát dữ liệu. Ví dụ như ta chuyển đổi một số nguyên kiểu short (2 byte) qua một số nguyên kiểu int (4 byte).

## 5.2. Chuyển đổi tường minh (explicit conversion)

Nếu ta muốn chuyển đổi kiểu mà trình biên dịch không tự động thì bạn phải thực hiện ép kiểu. Khi ta ép một kiểu dữ liệu này chuyển đổi qua một kiểu dữ liệu, ta có tình ép trình biên dịch phải tuân theo, thực hiện việc chuyển đổi.

// Ví dụ chuyển đổi kiểu tường minh

```
short x;
```

```
int y = 500;
```

```
x = (short)y;
```

\*Kiểu dữ liệu bị ép về sẽ nằm trong cặp dấu ngoặc tròn, trước khi bị chuyển đổi.

Việc ép kiểu như này sẽ nguy hiểm đến dữ liệu. Vì nếu ép từ kiểu có miền lớn sang miền nhỏ hơn. Để đảm bảo, C# cung cấp cho bạn một toán tử checked dùng kiểm tra việc ép kiểu có an toàn hay không. Nếu không an toàn thì nó sẽ tung ra một biệt lệ vào lúc chạy.

## 6. Nhập và xuất dữ liệu trong C#

### 6.1. Nhập dữ liệu từ bàn phím trong C#

Để nhập dữ liệu từ bàn phím, ta dùng phương thức ReadLine trong lớp Console.

Phương thức ReadLine sẽ đọc theo từng dòng của dữ liệu nhập vào. Phương thức này trả về một giá trị nullable kiểu string. Khi đã hết dữ liệu nhập vào mà phương thức này vẫn được gọi phương thức sẽ trả về giá trị null.

Tuy nhiên, phương thức ReadLine sẽ trả về một xâu ký tự chứa nội dung được nhập từ bàn phím. Trong trường hợp bạn muốn có một số nguyên thì phải thực hiện việc ép kiểu. Chuyển đổi từ kiểu string sang kiểu int.

### 6.2. Hiển thị dữ liệu ra màn hình trong C#

C# hỗ trợ 2 phương thức giúp bạn hiển thị dữ liệu ra ngoài màn hình.

- Phương thức WriteLine cho phép chúng ta xuất dữ liệu ra màn hình và sẽ xuống dòng khi kết thúc câu lệnh.
- Ngược lại phương thức Write cũng sẽ cho phép chúng ta xuất dữ liệu ra màn hình nhưng khi kết thúc câu lệnh đó sẽ không in xuống dòng.

## 7. Các lệnh điều khiển

### 7.1. Lệnh điều kiện

### 7.1.1. Lệnh điều kiện if – thiếu Cú pháp như sau:

```
if(<biểu thức điều kiện>)
```

```
{
```

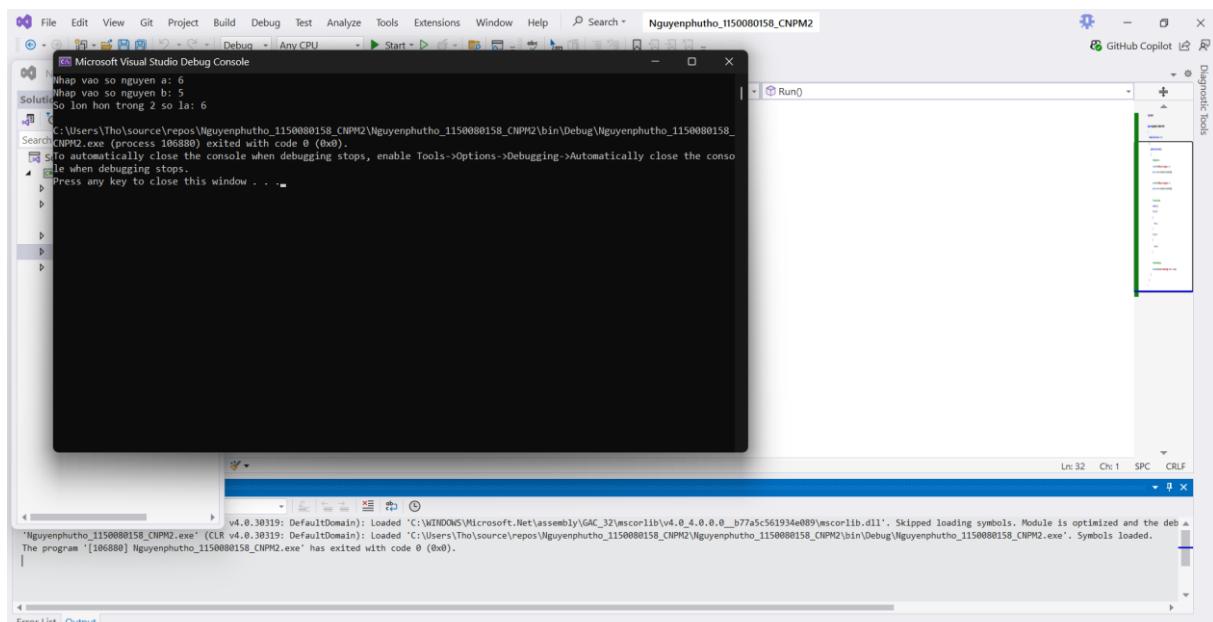
Tập các câu lệnh;

```
}
```

Cách thực hoạt động:

- Đầu tiên, ta xác định biểu thức điều kiện nhận giá trị True hoặc False.
- Nếu biểu thức điều kiện có giá trị True thì thực hiện Tập các câu lệnh.
- Còn ngược lại, ta không thực thi Tập các câu lệnh. Hay nói cách khác, ta chạy câu lệnh tiếp sau if.

**Thực hành 2:** Tìm số lớn hơn trong 2 số nguyên a và b được nhập vào từ bàn phím.



using System;

```
namespace Nguyenphutho_1150080158_CNPM2
```

```
{
```

```
    internal static class Bai2_THO
```

```
{
```

```
    public static void Run()
```

```
{
```

```
        // Nhập dữ liệu
```

```

Console.Write("Nhập vào số nguyên a: ");
int a = Convert.ToInt32(Console.ReadLine());

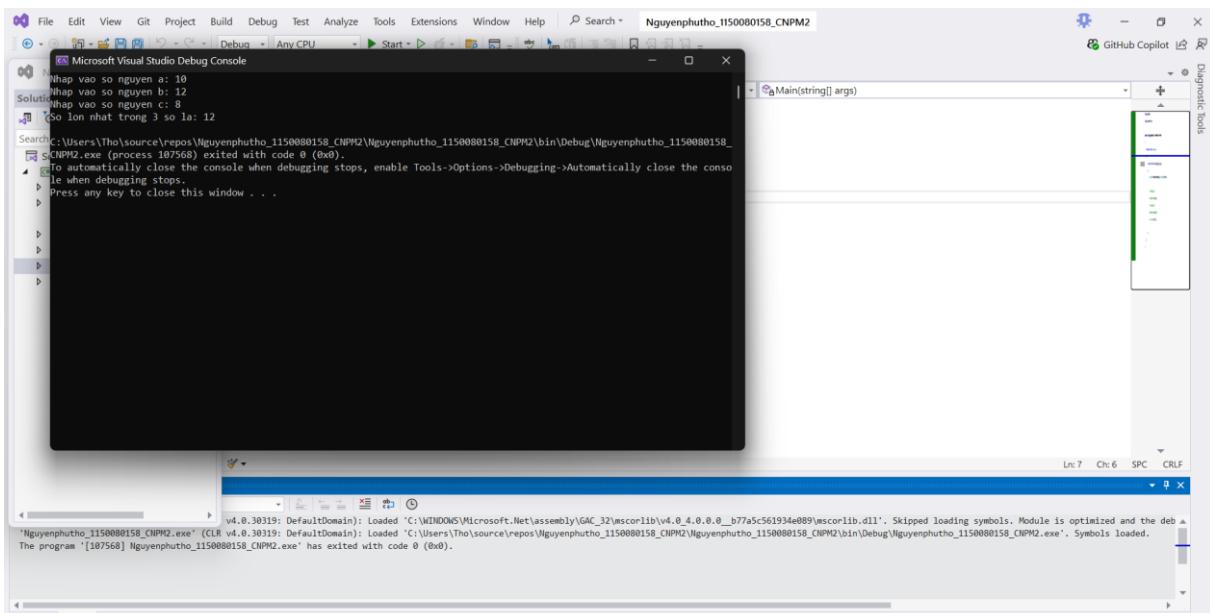
Console.Write("Nhập vào số nguyên b: ");
int b = Convert.ToInt32(Console.ReadLine());

// Tìm số lớn hơn
int max = 0;
if (a > b)
{
    max = a;
}
if (b > a)
{
    max = b;
}

// Hiển thị kết quả
Console.WriteLine("Số lớn hơn trong 2 số là: " + max);
}
}
}

```

**Thực hành 3:** Tìm số có giá trị lớn nhất trong 3 số nguyên a, b, c được nhập vào từ bàn phím.



using System;

namespace Nguyenphutho\_1150080158\_CNPM2

{

internal static class Bai3\_THO

{

public static void Run()

{

// Nhập dữ liệu

Console.Write("Nhập vào số nguyên a: ");

int a = Convert.ToInt32(Console.ReadLine());

Console.Write("Nhập vào số nguyên b: ");

int b = Convert.ToInt32(Console.ReadLine());

Console.Write("Nhập vào số nguyên c: ");

int c = Convert.ToInt32(Console.ReadLine());

// Tìm số lớn nhất trong 3 số

int max = a; // giả sử ban đầu max = a

if (b > max)

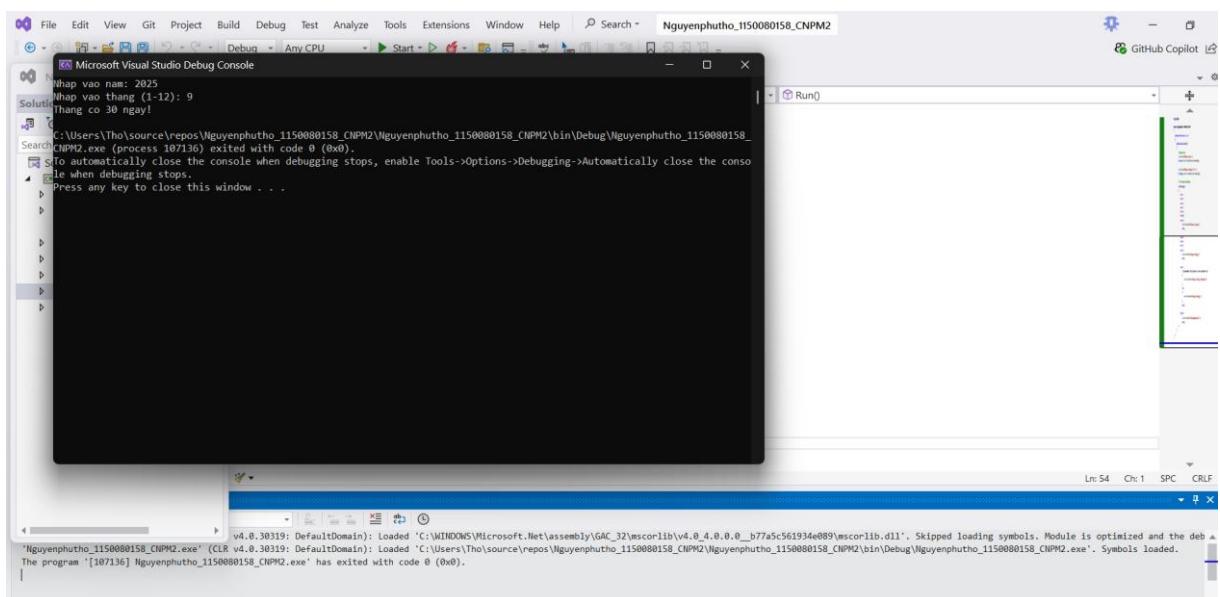
```

{
    max = b;
}
if (c > max)
{
    max = c;
}

// Hien thi ket qua
Console.WriteLine("So lon nhat trong 3 so la: " + max);
}
}
}

```

**Thực hành 4:** Cho biết một tháng có bao nhiêu ngày. Biết tháng, năm được nhập vào từ bàn phím.



using System;

namespace Nguyenphutho\_1150080158\_CNPM2

{

internal static class Bai4\_THO

{

```

public static void Run()
{
    // Nhập dữ liệu
    Console.Write("Nhập vào năm: ");
    int nam = Convert.ToInt32(Console.ReadLine());

    Console.Write("Nhập vào tháng (1-12): ");
    int thang = Convert.ToInt32(Console.ReadLine());

    // Tìm số ngày trong tháng
    switch (thang)
    {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12:
            Console.WriteLine("Tháng có 31 ngày!");
            break;

        case 4:
        case 6:
        case 9:
        case 11:
            Console.WriteLine("Tháng có 30 ngày!");
            break;

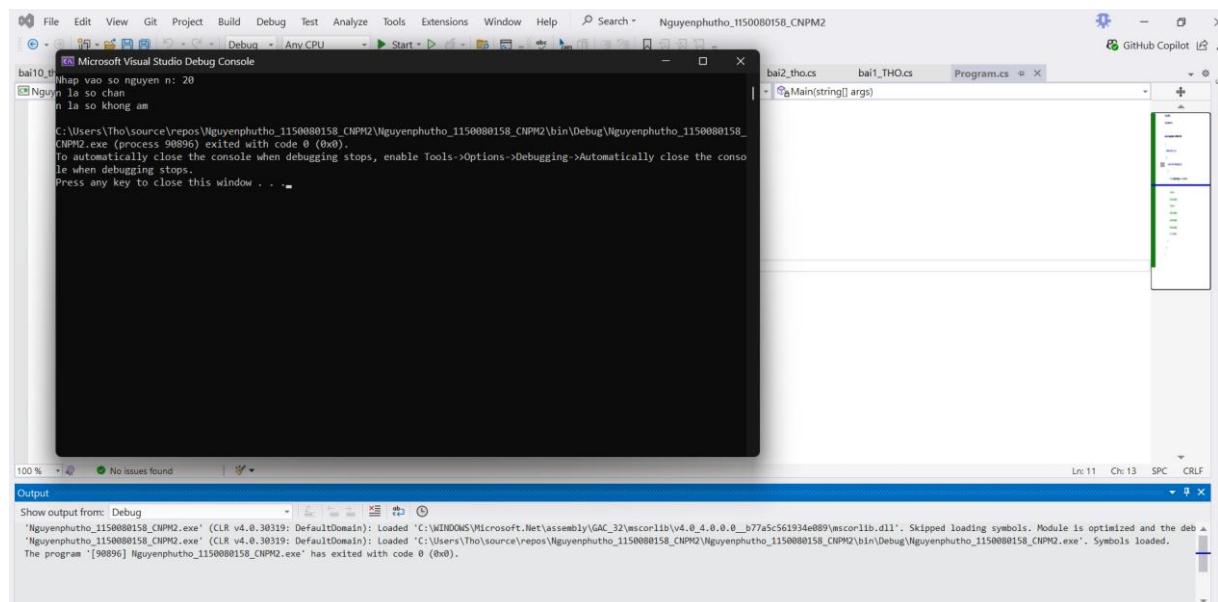
        case 2:
            if ((nam % 400 == 0) || (nam % 4 == 0 && nam % 100 != 0))
            {

```

```
        Console.WriteLine("Thang co 29 ngay! (nam nhuan)");  
    }  
    else  
    {  
        Console.WriteLine("Thang co 28 ngay!");  
    }  
    break;  
  
    default:  
        Console.WriteLine("Thang khong hop le!");  
        break;  
    }  
}  
}
```

**Thực hành 5: Viết chương trình nhập vào một số nguyên n. Cho biết:**

- a) n là số chẵn hay số lẻ?
  - b) n là số âm hay số không âm?



using System;

namespace Nguyenphutho 1150080158 CNPM2

```

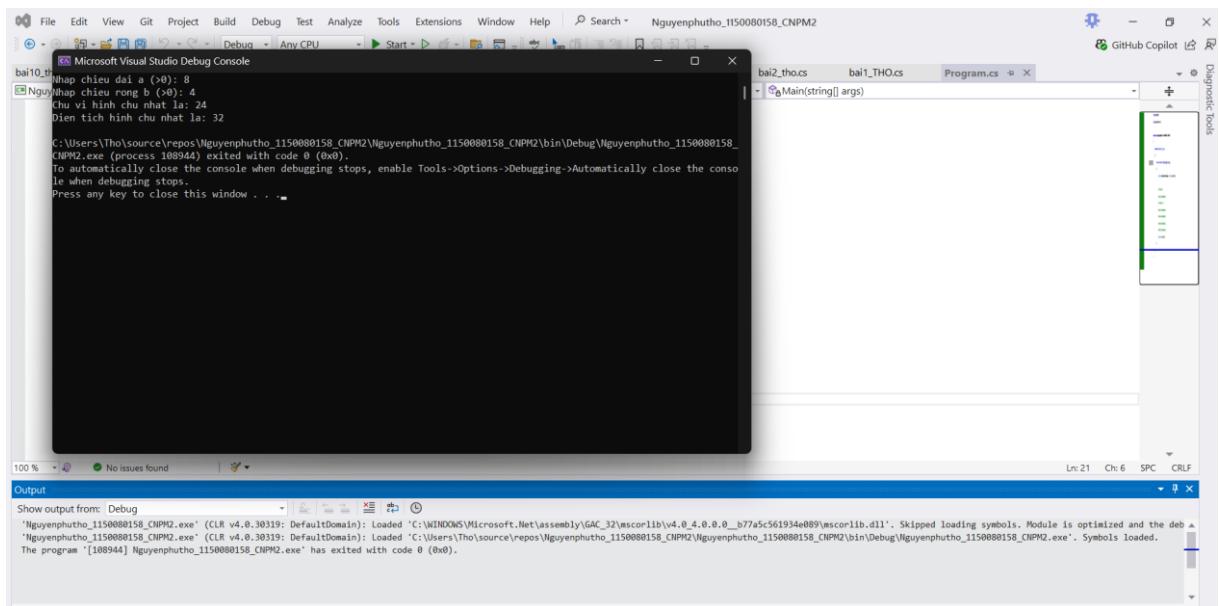
{
internal static class Bai5_THO
{
    public static void Run()
    {
        // Nhập dữ liệu
        Console.WriteLine("Nhập vào số nguyên n: ");
        int n = Convert.ToInt32(Console.ReadLine());

        // a) Kiểm tra chẵn/lẻ
        if (n % 2 == 0)
        {
            Console.WriteLine("n là số chẵn");
        }
        else
        {
            Console.WriteLine("n là số lẻ");
        }

        // b) Kiểm tra âm/không âm
        if (n < 0)
        {
            Console.WriteLine("n là số âm");
        }
        else
        {
            Console.WriteLine("n là số không âm");
        }
    }
}

```

**Thực hành 6:** Viết chương trình nhập vào 2 số thực dương chỉ chiều dài và chiều rộng của hình chữ nhật. In ra màn hình chu vi và diện tích của hình chữ nhật đó.



using System;

namespace Nguyenphutho\_1150080158\_CNPM2

{

internal static class Bai6\_THO

{

public static void Run()

{

// Nhập dữ liệu

Console.Write("Nhập chiều dài a (>0): ");

double a = Convert.ToDouble(Console.ReadLine());

Console.Write("Nhập chiều rộng b (>0): ");

double b = Convert.ToDouble(Console.ReadLine());

// Kiểm tra hợp lệ

if (a <= 0 || b <= 0)

{

Console.WriteLine("Chieu dai va chieu rong phai la so thuc duong!");

return;

```

    }

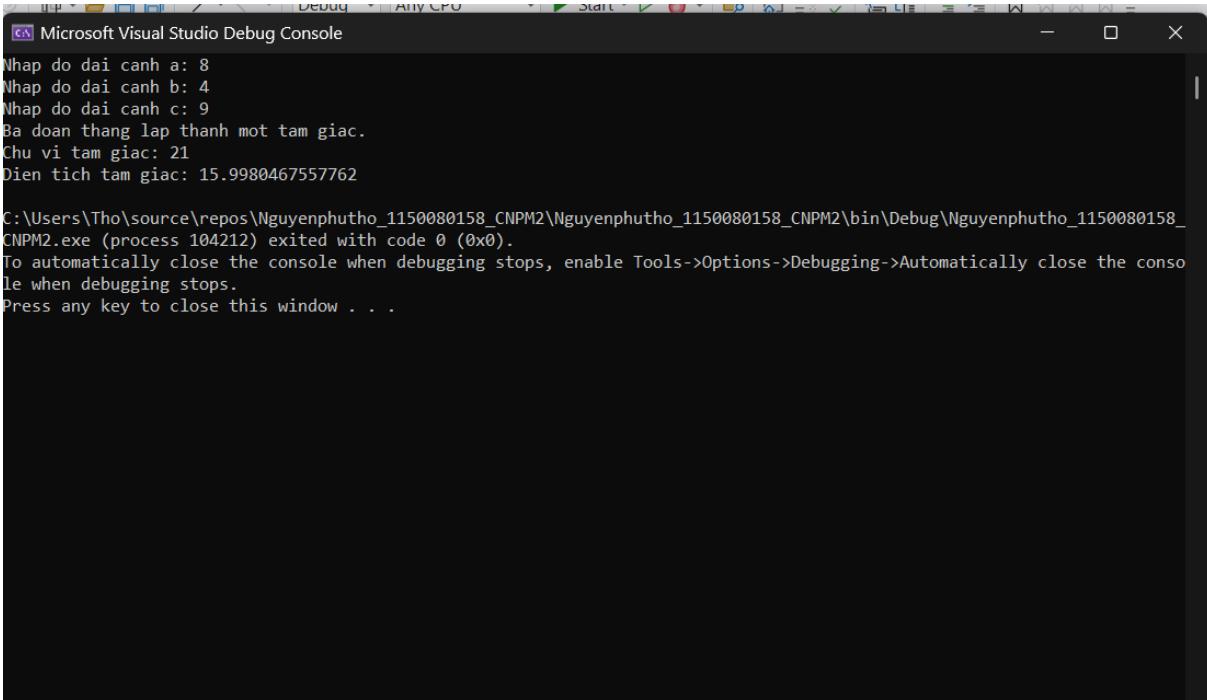
    // Tinh chu vi va dien tich
    double P = 2 * (a + b);
    double S = a * b;

    // Hien thi ket qua
    Console.WriteLine("Chu vi hinh chu nhat la: " + P);
    Console.WriteLine("Dien tich hinh chu nhat la: " + S);
}

}

```

**Thực hành 7:** Viết chương trình nhập vào ba số thực chỉ độ dài của ba đoạn thẳng. Kiểm tra nếu ba đoạn thẳng này lập thành được một tam giác thì hiển thị chu vi và diện tích của tam giác đó.



The screenshot shows the Microsoft Visual Studio Debug Console window. The console output is as follows:

```

Microsoft Visual Studio Debug Console
Nhap do dai canh a: 8
Nhap do dai canh b: 4
Nhap do dai canh c: 9
Ba doan thang lap thanh mot tam giac.
Chu vi tam giac: 21
Dien tich tam giac: 15.9980467557762

C:\Users\Tho\source\repos\Nguyenphutho_1150080158_CNPM2\Nguyenphutho_1150080158_CNPM2\bin\Debug\Nguyenphutho_1150080158_CNPM2.exe (process 104212) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

using System;

namespace Nguyenphutho\_1150080158\_CNPM2

{

```

internal static class Bai7_THO
{
    public static void Run()
    {
        // Nhập độ dài 3 cạnh
        Console.Write("Nhập độ dài cạnh a: ");
        double a = Convert.ToDouble(Console.ReadLine());

        Console.Write("Nhập độ dài cạnh b: ");
        double b = Convert.ToDouble(Console.ReadLine());

        Console.Write("Nhập độ dài cạnh c: ");
        double c = Convert.ToDouble(Console.ReadLine());

        // Kiểm tra hợp lệ
        if (a <= 0 || b <= 0 || c <= 0)
        {
            Console.WriteLine("Độ dài các cạnh phải là số thực dương!");
            return;
        }

        // Kiểm tra điều kiện tạo thành tam giác
        if (a + b > c && a + c > b && b + c > a)
        {
            double P = a + b + c;          // Chu vi
            double p = P / 2;              // Nửa chu vi
            double S = Math.Sqrt(p * (p - a) * (p - b) * (p - c)); // Công thức Heron

            Console.WriteLine("Ba đoạn thẳng lập thành một tam giác.");
            Console.WriteLine("Chu vi tam giác: " + P);
            Console.WriteLine("Diện tích tam giác: " + S);
        }
    }
}

```

```

        else
        {
            Console.WriteLine("Ba doan thang khong the lap thanh mot tam giac.");
        }
    }
}
}

```

**Thực hành 8:** Viết chương trình giải phương trình bậc 2:  $ax^2 + bx + c = 0$ .

using System;

```

namespace Nguyenphutho_1150080158_CNPM2
{
    internal static class Bai8_THO
    {
        public static void Run()
        {
            // Nhaph he so
            Console.Write("Nhaph a: ");
            double a = Convert.ToDouble(Console.ReadLine());

```

```

Console.WriteLine("Nhập b: ");
double b = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Nhập c: ");
double c = Convert.ToDouble(Console.ReadLine());

// Xử lý trường hợp
if (a == 0)
{
    // Phương trình bậc 1: bx + c = 0
    if (b == 0)
    {
        if (c == 0)
            Console.WriteLine("Phương trình có vô số nghiệm.");
        else
            Console.WriteLine("Phương trình vô nghiệm.");
    }
    else
    {
        double x = -c / b;
        Console.WriteLine("Phương trình bậc nhất, có nghiệm x = " + x);
    }
}
else
{
    // Phương trình bậc 2
    double delta = b * b - 4 * a * c;

    if (delta < 0)
    {
        Console.WriteLine("Phương trình vô nghiệm.");
    }
}

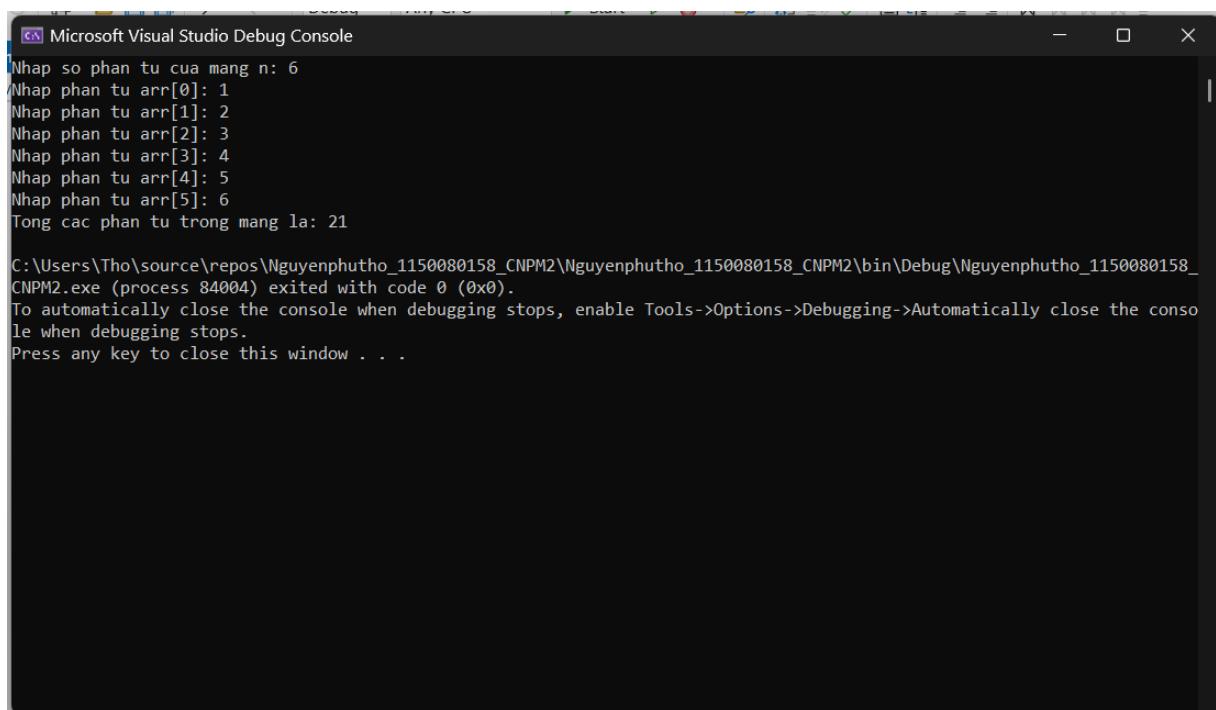
```

```
else if (delta == 0)
{
    double x = -b / (2 * a);
    Console.WriteLine("Phuong trinh co nghiem kep: x = " + x);
}

else
{
    double x1 = (-b + Math.Sqrt(delta)) / (2 * a);
    double x2 = (-b - Math.Sqrt(delta)) / (2 * a);
    Console.WriteLine("Phuong trinh co 2 nghiem phan biet:");
    Console.WriteLine("x1 = " + x1);
    Console.WriteLine("x2 = " + x2);
}

}
}
```

**Thực hành 9:** Tính tổng các phần tử trong mảng.



```
using System;
```

```

namespace Nguyenphutho_1150080158_CNPM2
{
    internal static class Bai9_THO
    {
        public static void Run()
        {
            Console.WriteLine("Nhập số phần tử của mảng n: ");
            int n = Convert.ToInt32(Console.ReadLine());

            if (n <= 0)
            {
                Console.WriteLine("Số phần tử phải lớn hơn 0!");
                return;
            }

            int[] arr = new int[n];

            // Nhập mảng
            for (int i = 0; i < n; i++)
            {
                Console.Write($"Nhập phần tử arr[{i}]: ");
                arr[i] = Convert.ToInt32(Console.ReadLine());
            }

            // Tính tổng
            int sum = 0;
            for (int i = 0; i < n; i++)
            {
                sum += arr[i];
            }

            // Hiển thị kết quả
        }
    }
}

```

```

        Console.WriteLine("Tong cac phan tu trong mang la: " + sum);
    }
}
}

```

**Thực hành 10:** Dùng giải thuật sắp xếp chọn (Selection Sort) để sắp xếp tăng dần mảng các số nguyên. Mảng các số nguyên được lưu trong tệp văn bản có tên là “input\_array.txt”.

```

Microsoft Visual Studio Debug Console
Nhập số phần tử của mảng n: 5
Nhập phần tử arr[0]: 1
Nhập phần tử arr[1]: 2
Nhập phần tử arr[2]: 3
Nhập phần tử arr[3]: 4
Nhập phần tử arr[4]: 5
Mảng ban đầu:
1 2 3 4 5
Mảng sau khi sắp xếp tăng dần:
1 2 3 4 5

C:\Users\Tho\source\repos\Nguyenphutho_1150080158_CNPM2\Nguyenphutho_1150080158_CNPM2\bin\Debug\Nguyenphutho_1150080158_CNPM2.exe (process 108860) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

```

using System;
using System.IO;

namespace Nguyenphutho_1150080158_CNPM2
{
    internal static class Bai10_THO
    {
        public static void Run()
        {
            int[] arr;

            string filePath = "input_array.txt";

```

```

if (File.Exists(filePath))
{
    // Doc du lieu tu file
    string content = File.ReadAllText(filePath);
    string[] parts = content.Split(new char[] { ' ', '\n', '\r', '\t' },
        StringSplitOptions.RemoveEmptyEntries);

    arr = new int[parts.Length];
    for (int i = 0; i < parts.Length; i++)
    {
        arr[i] = int.Parse(parts[i]);
    }

    Console.WriteLine("Doc mang tu file input_array.txt thanh cong!");
}

else
{
    // Nguoi dung tu nhap
    Console.Write("Nhập số phần tử của mảng n: ");
    int n = Convert.ToInt32(Console.ReadLine());

    arr = new int[n];
    for (int i = 0; i < n; i++)
    {
        Console.Write($"Nhập phần tử arr[{i}]: ");
        arr[i] = Convert.ToInt32(Console.ReadLine());
    }
}

Console.WriteLine("Mảng ban dau:");
PrintArray(arr);

```

```

// Selection Sort
for (int i = 0; i < arr.Length - 1; i++)
{
    int minIndex = i;
    for (int j = i + 1; j < arr.Length; j++)
    {
        if (arr[j] < arr[minIndex])
        {
            minIndex = j;
        }
    }
}

```

```

// Doi cho
if (minIndex != i)
{
    int temp = arr[i];
    arr[i] = arr[minIndex];
    arr[minIndex] = temp;
}

```

```

Console.WriteLine("Mang sau khi sap xep tang dan:");
PrintArray(arr);
}

```

```

private static void PrintArray(int[] arr)
{
    for (int i = 0; i < arr.Length; i++)
    {
        Console.Write(arr[i] + " ");
    }
    Console.WriteLine();
}

```

```

    }
}

}

```

**Thực hành 11:** Chèn thêm một số nguyên được nhập vào từ bàn phím vào mảng  
đã sắp xếp tăng dần nhưng không làm mất tính tăng dần của mảng.

```

Microsoft Visual Studio Debug Console
Nhập số phần tử n: 7
Nhập arr[0] (mảng tăng dần): 1
Nhập arr[1] (mảng tăng dần): 2
Nhập arr[2] (mảng tăng dần): 3
Nhập arr[3] (mảng tăng dần): 4
Nhập arr[4] (mảng tăng dần): 5
Nhập arr[5] (mảng tăng dần): 6
Nhập arr[6] (mảng tăng dần): 7
Nhập số nguyên x cần chèn: 2
Mảng ban đầu (tăng dần):
1 2 3 4 5 6 7
Chèn 2 vào vị trí 1:
1 2 2 3 4 5 6 7

C:\Users\Tho\source\repos\Nguyenphutho_1150080158_CNPM2\Nguyenphutho_1150080158_CNPM2\bin\Debug\Nguyenphutho_1150080158_CNPM2.exe (process 105232) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

using System;

```

namespace Nguyenphutho_1150080158_CNPM2
{
    internal static class Bai11_THO
    {
        public static void Run()
        {
            Console.Write("Nhập số phần tử n: ");
            int n = Convert.ToInt32(Console.ReadLine());
            if (n < 0)
            {
                Console.WriteLine("n không hợp lệ!");
            }
            return;
        }
    }
}

```

```
}
```

```
int[] arr = new int[n];
```

```
for (int i = 0; i < n; i++)
```

```
{
```

```
    Console.WriteLine($"Nhập arr[{i}] (mảng tăng dần): ");
```

```
    arr[i] = Convert.ToInt32(Console.ReadLine());
```

```
}
```

```
// Kiểm tra mảng có tăng dần không (nếu chưa, sẽ sắp xếp lại để an toàn)
```

```
if (!IsNonDecreasing(arr))
```

```
{
```

```
    Console.WriteLine("Cảnh báo: Mảng chưa tăng dần. Se tu dòng sắp xếp tăng  
    dần trước khi chen.");
```

```
    Array.Sort(arr);
```

```
}
```

```
Console.WriteLine("Nhập số nguyên x cần chèn: ");
```

```
int x = Convert.ToInt32(Console.ReadLine());
```

```
// Tìm vị trí chèn bằng cách kiểm nghiệm (vị trí đầu tiên  $\geq x$ )
```

```
int pos = LowerBound(arr, x);
```

```
// Tạo mảng mới có kích thước  $n+1$  và chèn x
```

```
int[] result = new int[n + 1];
```

```
for (int i = 0; i < pos; i++) result[i] = arr[i];
```

```
result[pos] = x;
```

```
for (int i = pos; i < n; i++) result[i + 1] = arr[i];
```

```
// In kết quả
```

```
Console.WriteLine("Mảng ban đầu (tăng dần):");
```

```
PrintArray(arr);
```

```

Console.WriteLine($"Chen {x} vao vi tri {pos}:");
PrintArray(result);
}

// Kiem tra mang khong giam (tang dan cho phep phan tu bang nhau)
private static bool IsNonDecreasing(int[] a)
{
    for (int i = 1; i < a.Length; i++)
        if (a[i] < a[i - 1]) return false;
    return true;
}

// LowerBound: tra ve chi so phan tu dau tien >= x (neu tat ca < x thi tra ve
a.Length)
private static int LowerBound(int[] a, int x)
{
    int l = 0, r = a.Length; // [l, r)
    while (l < r)
    {
        int m = l + (r - l) / 2;
        if (a[m] >= x) r = m;
        else l = m + 1;
    }
    return l;
}

private static void PrintArray(int[] a)
{
    for (int i = 0; i < a.Length; i++)
        Console.Write(a[i] + " ");
    Console.WriteLine();
}

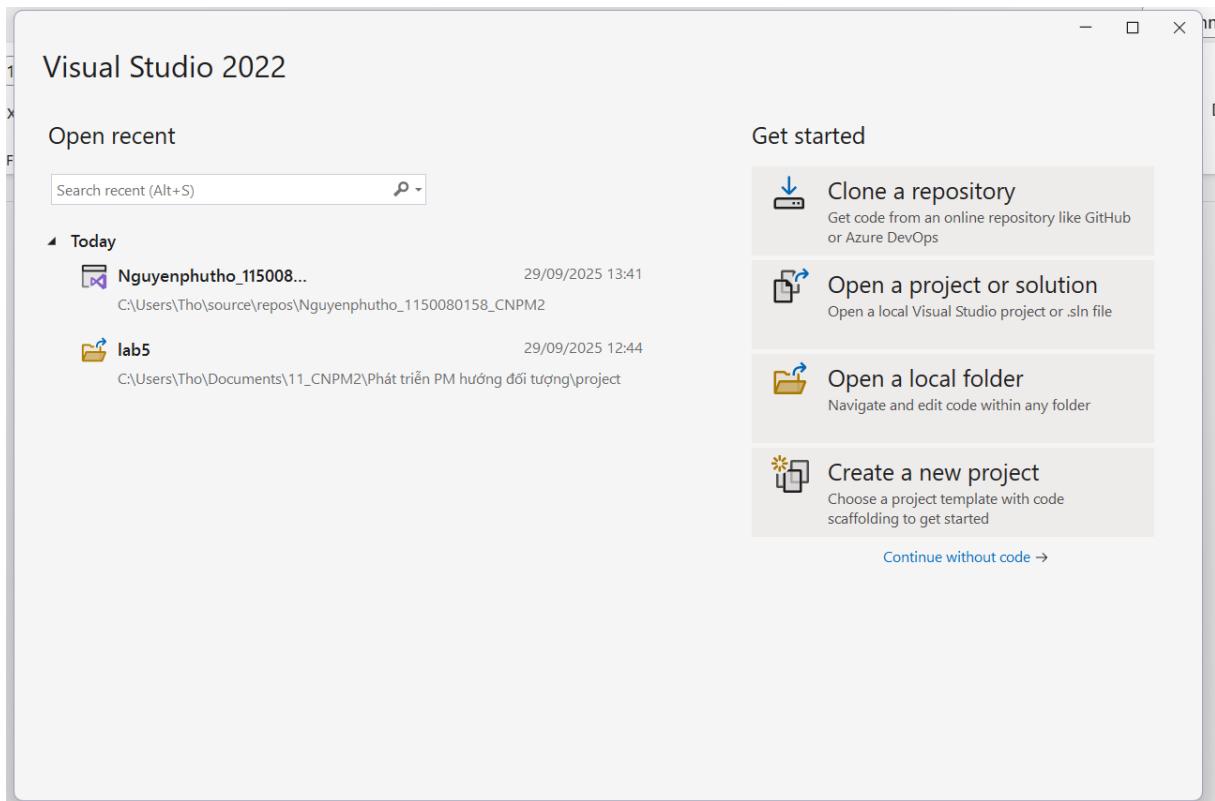
```

```
    }  
}  
}
```

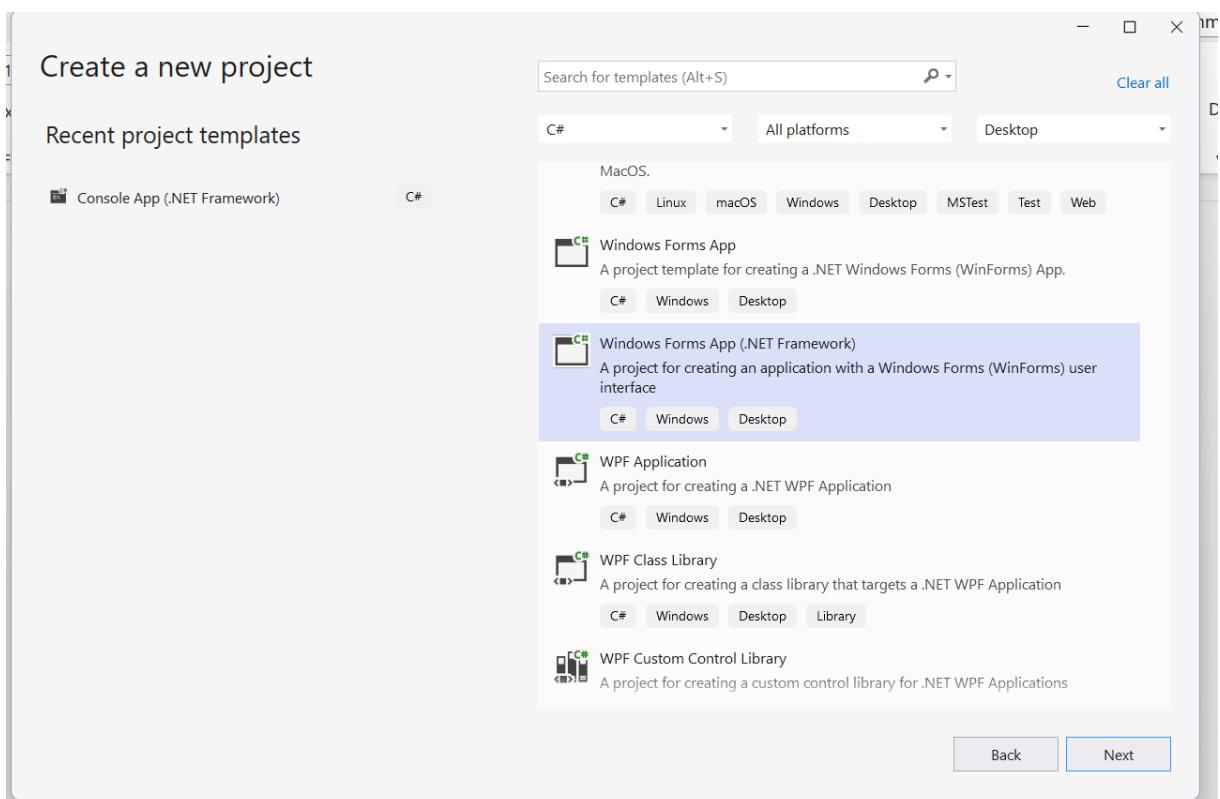
## LAB 2

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

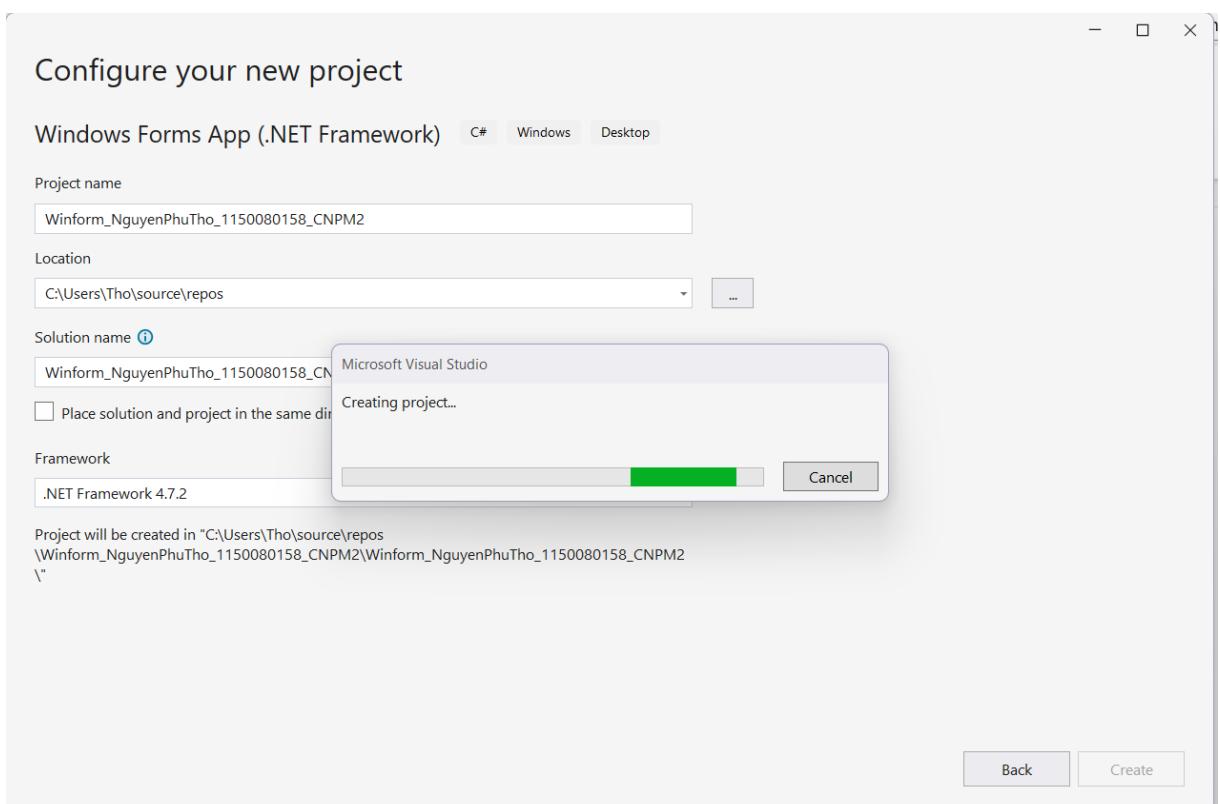
Bước 1: Khởi động chương trình visual studio 2022. Hình ảnh dưới đây là màn hình khởi động của ứng dụng:



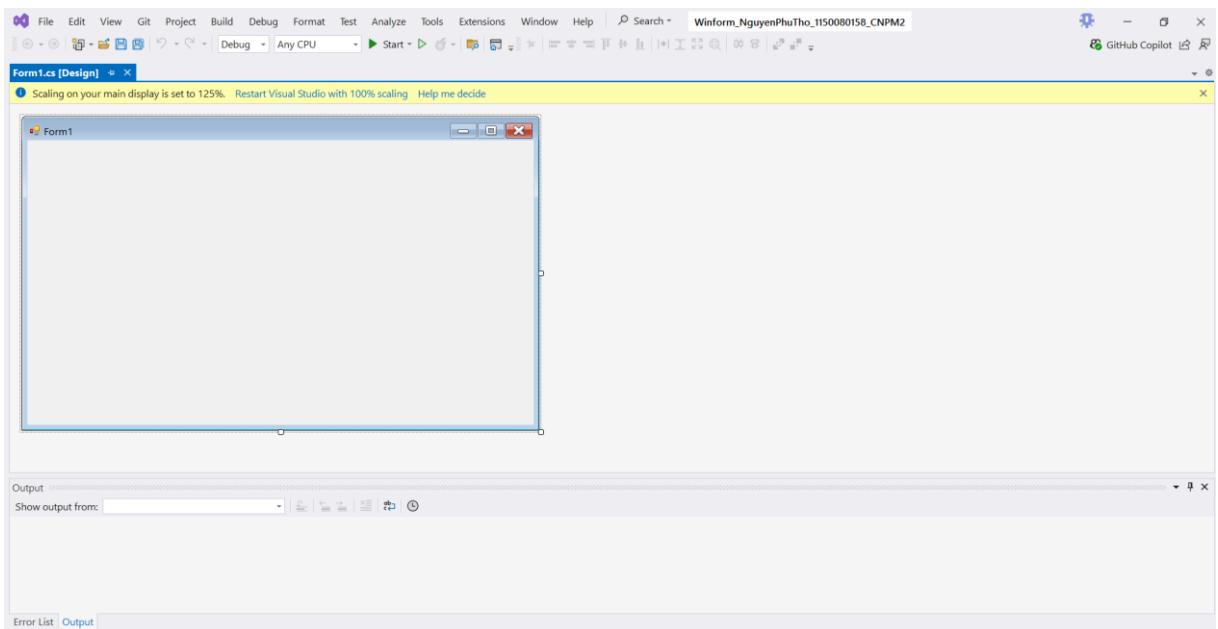
Bước 2: Chọn Create a new project trên màn hình khởi động. Trong màn hình tiếp theo, ở mục All languages → chọn C# trong danh sách ngôn ngữ lập trình. Mục All platforms → chọn Windows. Ở mục All project types → chọn Desktop.



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 như sau:

Form này sẽ cho phép người dùng nhập các giá trị là số thực vào 2 biến a và b.

Tùy vào các chức năng của các phép toán (cộng, trừ, nhân, chia) mà hiển thị ở phần kết quả.

Đặt tên biến cho một số control như sau:

- TextBox cho phép nhập a: txtA
- TextBox cho phép nhập b: txtB
- TextBox cho phép hiển thị kết quả phép tính: txtKetQua
- Button phép toán cộng: btnCong
- Button phép toán trừ: btnTru
- Button phép toán nhân: btnNhan
- Button phép toán chia: btnChia
- Button phép toán xóa dữ liệu: btnXoa
- Button phép toán thoát chương trình: btnThoat

May tinh 4 phep

Nhap a:

Nhap b:

Ket qua:

Cong Tru Nhan Chia Xoa  
Thoat

Cộng:

May tinh 4 phep

Nhap a: 3

Nhap b: 2

Ket qua: 5

Cong Tru Nhan Chia Xoa  
Thoat

Trừ :

May tinh 4 phep

Nhap a: 3

Nhap b: 2

Ket qua: 1

Cong Tru Nhan Chia Xoa  
Thoat

Nhân:

May tinh 4 phep

Nhap a: 3

Nhap b: 2

Ket qua: 6

Cong Tru Nhan Chia Xoa  
Thoat

Chia:

May tinh 4 phep

Nhap a: 3

Nhap b: 2

Ket qua: 1.5

Cong Tru Nhan Chia Xoa  
Thoat

**Code:**

```
using System;
using System.Drawing;
using System.Windows.Forms;

namespace Winform_NguyenPhuTho_1150080158_CNPM2
{
    public partial class Form1 : Form
    {
        // Khai báo control ở cấp lớp (field) để các handler “thấy” được
        private TextBox txtA;
        private TextBox txtB;
        private TextBox txtKetQua;
        private Button btnCong, btnTru, btnNhan, btnChia, btnXoa, btnThoat;
        private Label lblA, lblB, lblKq;

        public Form1()
        {
            // Thay cho InitializeComponent của Designer
            InitializeComponentManual();
        }

        private void InitializeComponentManual()
        {
            this.Text = "May tinh 4 phep";
            this.StartPosition = FormStartPosition.CenterScreen;
            this.ClientSize = new Size(420, 240);
            this.Font = new Font("Segoe UI", 10);

            // Label A
            lblA = new Label
```

```

{
    AutoSize = true,
    Text = "Nhập a:",
    Location = new Point(20, 20)
};

// TextBox A
txtA = new TextBox
{
    Name = "txtA",
    Location = new Point(120, 16),
    Width = 260
};

// Label B
lblB = new Label
{
    AutoSize = true,
    Text = "Nhập b:",
    Location = new Point(20, 60)
};

// TextBox B
txtB = new TextBox
{
    Name = "txtB",
    Location = new Point(120, 56),
    Width = 260
};

// Label KQ
lblKq = new Label
{
    AutoSize = true,

```

```

    Text = "Kết quả:",
    Location = new Point(20, 100)
};

// TextBox Kết Quả (read-only)
txtKetQua = new TextBox
{
    Name = "txtKetQua",
    Location = new Point(120, 96),
    Width = 260,
    ReadOnly = true
};

// Buttons
btnCong = new Button { Name = "btnCong", Text = "Cong", Location = new
Point(20, 140), Size = new Size(60, 32) };

btnTru = new Button { Name = "btnTru", Text = "Tru", Location = new
Point(90, 140), Size = new Size(60, 32) };

btnNhan = new Button { Name = "btnNhan", Text = "Nhan", Location = new
Point(160, 140), Size = new Size(60, 32) };

btnChia = new Button { Name = "btnChia", Text = "Chia", Location = new
Point(230, 140), Size = new Size(60, 32) };

btnXoa = new Button { Name = "btnXoa", Text = "Xoa", Location = new
Point(300, 140), Size = new Size(60, 32) };

btnThoat = new Button { Name = "btnThoat", Text = "Thoát", Location = new
Point(300, 180), Size = new Size(60, 32) };

// Gắn sự kiện
btnCong.Click += (s, e) => Tinh((a, b) => a + b);
btnTru.Click += (s, e) => Tinh((a, b) => a - b);
btnNhan.Click += (s, e) => Tinh((a, b) => a * b);
btnChia.Click += (s, e) =>
{

```

```

        if (!TryGetAB(out double a, out double b)) return;
        if (b == 0)
        {
            MessageBox.Show("Khong the chia cho 0!", "Loi",
MessageBoxButtons.OK, MessageBoxIcon.Warning);
            return;
        }
        txtKetQua.Text = (a / b).ToString();
    };
    btnXoa.Click += (s, e) =>
    {
        txtA.Clear();
        txtB.Clear();
        txtKetQua.Clear();
        txtA.Focus();
    };
    btnThoat.Click += (s, e) => this.Close();

    // Thêm control lên form
    this.Controls.AddRange(new Control[]
    {
        lblA, txtA,
        lblB, txtB,
        lblKq, txtKetQua,
        btnCong, btnTru, btnNhan, btnChia, btnXoa, btnThoat
    });
}

```

```

private bool TryGetAB(out double a, out double b)
{
    if (!double.TryParse(txtA.Text, out a))
    {

```

```

        MessageBox.Show("Vui long nhap so hop le cho a!", "Thong bao",
MessageBoxButtons.OK, MessageBoxIcon.Information);

        b = 0; return false;
    }

    if (!double.TryParse(txtB.Text, out b))
    {

        MessageBox.Show("Vui long nhap so hop le cho b!", "Thong bao",
MessageBoxButtons.OK, MessageBoxIcon.Information);

        return false;
    }

    return true;
}

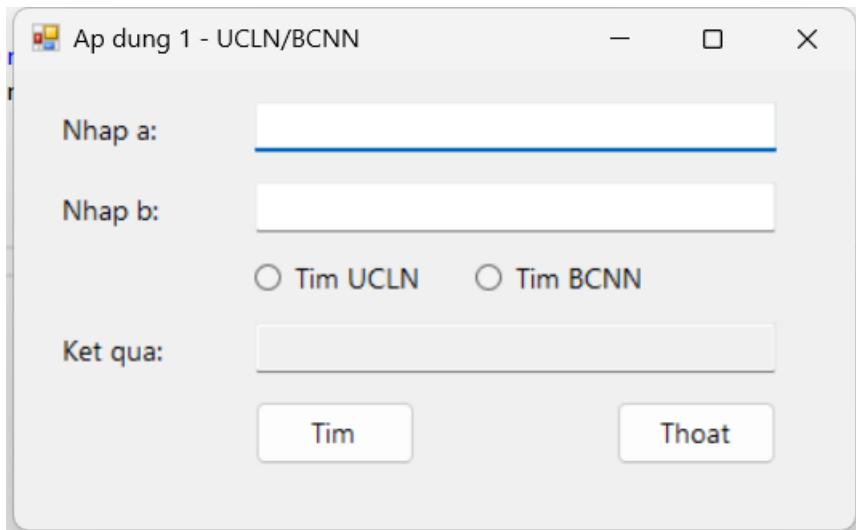
```

```

private void Tinh(Func<double, double, double> op)
{
    if (!TryGetAB(out double a, out double b)) return;
    txtKetQua.Text = op(a, b).ToString();
}

```

**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.Drawing;
using System.Windows.Forms;

namespace Nguyenphutho_1150080158_CNPM2
{
    public class ApDung1 : Form
    {
        // Controls

        private TextBox txtA, txtB, txtKetQua;
        private RadioButton radUSCLN, radBCNN;
        private Button btnTim, btnThoat;
        private Label lblA, lblB, lblKQ;
    }
}

```

```

public ApDung1()
{
    InitUI(); // không dùng InitializeComponent của Designer
}

// ===== Logic UCLN/BCNN =====

private int GCD(int x, int y)
{
    x = Math.Abs(x);
    y = Math.Abs(y);
    while (y != 0)
    {
        int t = y;
        y = x % y;
        x = t;
    }
    return x;
}

private int LCM(int x, int y)
{
    if (x == 0 || y == 0) return 0;
    return Math.Abs(x * y) / GCD(x, y);
}

// ===== Event handlers (KHÔNG dùng object?) =====

private void BtnTim_Click(object sender, EventArgs e)
{
    int a, b;
    if (!int.TryParse(txtA.Text, out a))
    {
        MessageBox.Show("Vui long nhap so nguyen hop le cho a!");
    }
}

```

```

        return;
    }

    if (!int.TryParse(txtB.Text, out b))
    {
        MessageBox.Show("Vui long nhap so nguyen hop le cho b!");
        return;
    }

    if (radUSCLN.Checked)
    {
        txtKetQua.Text = "UCLN = " + GCD(a, b);
    }
    else if (radBCNN.Checked)
    {
        txtKetQua.Text = "BCNN = " + LCM(a, b);
    }
    else
    {
        MessageBox.Show("Hay chon 'Tim UCLN' hoac 'Tim BCNN' truoc khi bam Tim!");
    }
}

private void BtnThoat_Click(object sender, EventArgs e)
{
    Close();
}

// ===== UI builder =====

private void InitUI()
{
    Text = "Ap dung 1 - UCLN/BCNN";
}

```

```

StartPosition = FormStartPosition.CenterScreen;
ClientSize = new Size(420, 230);
Font = new Font("Segoe UI", 10);

lblA = new Label { Text = "Nhập a:", AutoSize = true, Location = new Point(20,
20) };

txtA = new TextBox { Name = "txtA", Location = new Point(120, 16), Width =
260 };

lblB = new Label { Text = "Nhập b:", AutoSize = true, Location = new Point(20,
60) };

txtB = new TextBox { Name = "txtB", Location = new Point(120, 56), Width =
260 };

radUSCLN = new RadioButton { Name = "radUSCLN", Text = "Tim UCLN",
Location = new Point(120, 92), AutoSize = true };

radBCNN = new RadioButton { Name = "radBCNN", Text = "Tim BCNN",
Location = new Point(230, 92), AutoSize = true };

lblKQ = new Label { Text = "Kết quả:", AutoSize = true, Location = new
Point(20, 130) };

txtKetQua = new TextBox { Name = "txtKetQua", Location = new Point(120,
126), Width = 260, ReadOnly = true };

btnTim = new Button { Text = "Tim", Location = new Point(120, 165), Size =
new Size(80, 32) };

btnThoat = new Button { Text = "Thoát", Location = new Point(300, 165), Size
= new Size(80, 32) };

btnTim.Click += BtnTim_Click;
btnThoat.Click += BtnThoat_Click;

```

```

Controls.AddRange(new Control[]
{
    lblA, txtA,
    lblB, txtB,
    radUSCLN, radBCNN,
    lblKQ, txtKetQua,
    btnTim, btnThoat
});

}

}

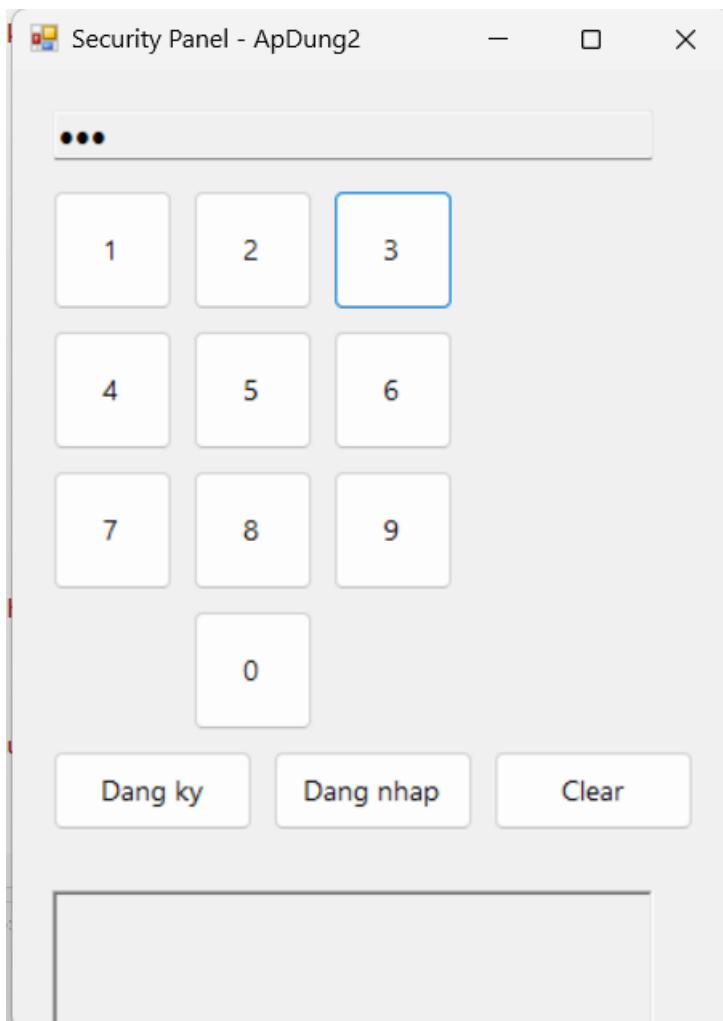
```

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

Giao diện ứng dụng:

Đăng kí:



Thông báo:

29/09/2025 19:15:04: Da dang ky mat khau moi.

Khi nhập đúng:

29/09/2025 19:15:04: Da dang ky mat khau moi.  
29/09/2025 19:15:24: Dang nhap thanh cong -  
Chap nhan.

Khi nhập sai:

29/09/2025 19:15:04: Da dang ky mat khau moi.  
29/09/2025 19:15:24: Dang nhap thanh cong -  
Chap nhan.  
29/09/2025 19:15:46: Sai mat khau - Tu choi.

using System;

using System.Drawing;

```

using System.Windows.Forms;

namespace Nguyenphutho_1150080158_CNPM2
{
    public class ApDung2 : Form
    {
        private TextBox txtPassword;
        private Button[] numButtons;
        private Button btnDangKy, btnDangNhap, btnClear;
        private RichTextBox rtbLog;

        private string currentPassword = null; // mật mã hiện tại (do user đăng ký)

        public ApDung2()
        {
            InitUI();
        }

        private void InitUI()
        {
            Text = "Security Panel - ApDung2";
            StartPosition = FormStartPosition.CenterScreen;
            ClientSize = new Size(360, 480);
            Font = new Font("Segoe UI", 10);

            // TextBox mật khẩu (ẩn ký tự nhập)
            txtPassword = new TextBox
            {
                Location = new Point(20, 20),
                Width = 300,
                UseSystemPasswordChar = true,
                ReadOnly = true
            }
        }
    }
}

```

```

};

// Bàn phím số
numButtons = new Button[10];
int startX = 20, startY = 60;
int btnSize = 60, gap = 10;
for (int i = 1; i <= 9; i++)
{
    numButtons[i] = new Button
    {
        Text = i.ToString(),
        Size = new Size(btnSize, btnSize),
        Location = new Point(startX + ((i - 1) % 3) * (btnSize + gap),
                            startY + ((i - 1) / 3) * (btnSize + gap))
    };
    numButtons[i].Click += NumButton_Click;
    Controls.Add(numButtons[i]);
}

// Nút 0
numButtons[0] = new Button
{
    Text = "0",
    Size = new Size(btnSize, btnSize),
    Location = new Point(startX + (btnSize + gap),
                        startY + 3 * (btnSize + gap))
};
numButtons[0].Click += NumButton_Click;
Controls.Add(numButtons[0]);

// Nút Đăng ký
btnDangKy = new Button

```

```

{
    Text = "Dang ky",
    Size = new Size(100, 40),
    Location = new Point(20, startY + 4 * (btnSize + gap))
};

btnDangKy.Click += BtnDangKy_Click;

// Nút Đăng nhập
btnDangNhap = new Button
{
    Text = "Dang nhap",
    Size = new Size(100, 40),
    Location = new Point(130, startY + 4 * (btnSize + gap))
};

btnDangNhap.Click += BtnDangNhap_Click;

// Nút Clear
btnClear = new Button
{
    Text = "Clear",
    Size = new Size(100, 40),
    Location = new Point(240, startY + 4 * (btnSize + gap))
};

btnClear.Click += (s, e) => txtPassword.Clear();

// RichTextBox log
rtbLog = new RichTextBox
{
    Location = new Point(20, startY + 5 * (btnSize + gap)),
    Size = new Size(300, 140),
    ReadOnly = true
};

```

```

        Controls.AddRange(new Control[] { txtPassword, btnDangKy, btnDangNhap,
        btnClear, rtbLog });
    }

private void NumButton_Click(object sender, EventArgs e)
{
    Button btn = sender as Button;
    if (btn != null) txtPassword.Text += btn.Text;
}

private void BtnDangKy_Click(object sender, EventArgs e)
{
    string input = txtPassword.Text.Trim();
    if (string.IsNullOrEmpty(input))
    {
        MessageBox.Show("Vui long nhap mat khau truoc khi Dang ky!");
        return;
    }

    currentPassword = input;
    txtPassword.Clear();
    rtbLog.AppendText($" {DateTime.Now}: Da dang ky mat khau moi.\n");
}

private void BtnDangNhap_Click(object sender, EventArgs e)
{
    string input = txtPassword.Text.Trim();
    txtPassword.Clear();

    if (string.IsNullOrEmpty(currentPassword))
    {

```

```

        MessageBox.Show("Chua co mat khau nao duoc dang ky!");
        return;
    }

    if (input == currentPassword)
    {
        rtbLog.AppendText($"{DateTime.Now}: Dang nhap thanh cong - Chap
nhan.\n");
    }
    else
    {
        rtbLog.AppendText($"{DateTime.Now}: Sai mat khau - Tu choi.\n");
    }
}
}

```

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.

Tham khảo giao diện ứng dụng sau:



## BẢNG TÍNH TIỀN DỊCH VỤ NHA KHOA

Tên khách hàng:

### Chọn dịch vụ

- Lấy cao răng (50.000đ)**
- Tẩy trắng răng (100.000đ)**
- Hàn răng (100.000đ/chiếc)**
- Bẻ răng (10.000đ/chiếc)**
- Bọc răng (1.000.000đ/chiếc)**

0	▲	▼
0	▲	▼
0	▲	▼

**Tổng tiền: 0 VND**

Phòng khám nha khoa - Thanh toán dịch vụ

## BẢNG TÍNH TIỀN DỊCH VỤ NHA KHOA

Tên khách hàng: Thọ

**Chọn dịch vụ**

<input checked="" type="checkbox"/> Lấy cao răng (50.000đ)	2
<input checked="" type="checkbox"/> Tẩy trắng răng (100.000đ)	
<input checked="" type="checkbox"/> Hàn răng (100.000đ/chiếc)	
<input checked="" type="checkbox"/> Bé răng (10.000đ/chiếc)	
<input checked="" type="checkbox"/> Bọc răng (1.000.000đ/chiếc)	2

**Tính tiền** **Thoát**

**Tổng tiền: 2,370,000 VND**

```
using System;
using System.Drawing;
using System.Windows.Forms;
```

```
namespace Nguyenphutho_1150080158_CNPM2
{
    public class ThucHanh2 : Form
    {
        private TextBox txtTenKH;
        private CheckBox chkCaoRang, chkTayTrang, chkHanRang, chkBeRang,
        chkBocRang;
        private NumericUpDown numHanRang, numBeRang, numBocRang;
        private Button btnTinh, btnThoat;
        private Label lblKetQua;

        public ThucHanh2()
        {
            InitializeComponent();
        }

        private void btnTinh_Click(object sender, EventArgs e)
        {
            int hanRang = numHanRang.Value;
            int beRang = numBeRang.Value;
            int bocRang = numBocRang.Value;
            double total = (hanRang * 100000) + (beRang * 10000) + (bocRang * 1000000);
            lblKetQua.Text = "Tổng tiền: " + total.ToString("###,###,###") + " VND";
        }

        private void btnThoat_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

```

    {
        InitUI();
    }

private void InitUI()
{
    Text = "🦷 Phòng khám nha khoa - Thanh toán dịch vụ";
    StartPosition = FormStartPosition.CenterScreen;
    ClientSize = new Size(600, 450);
    BackColor = Color.WhiteSmoke;
    Font = new Font("Segoe UI", 10);

    // Tiêu đề
    Label lblTitle = new Label
    {
        Text = "BẢNG TÍNH TIỀN DỊCH VỤ NHA KHOA",
        Font = new Font("Segoe UI", 14, FontStyle.Bold),
        ForeColor = Color.DarkBlue,
        AutoSize = true,
        Location = new Point(140, 20)
    };

    // Tên khách hàng
    Label lblTenKH = new Label { Text = "Tên khách hàng:", Location = new Point(40, 70), AutoSize = true };
    txtTenKH = new TextBox { Location = new Point(180, 66), Width = 350,
        BorderStyle = BorderStyle.FixedSingle };

    // Nhóm dịch vụ
    GroupBox grpDV = new GroupBox
    {
        Text = "Chọn dịch vụ",

```

```
    Font = new Font("Segoe UI", 11, FontStyle.Bold),  
    Location = new Point(40, 110),  
    Size = new Size(500, 200)  
};
```

```
chkCaoRang = new CheckBox { Text = "L\u00e1y cao r\u00e1ng (50.000\u00a3)", Location =  
new Point(20, 30), AutoSize = true };
```

```
chkTayTrang = new CheckBox { Text = "T\u00e1y tr\u00e1ng r\u00e1ng (100.000\u00a3)", Location =  
new Point(20, 60), AutoSize = true };
```

```
chkHanRang = new CheckBox { Text = "H\u00e1n r\u00e1ng (100.000\u00a3/chi\u00e9c)", Location =  
new Point(20, 90), AutoSize = true };
```

```
numHanRang = new NumericUpDown { Location = new Point(350, 90), Width = 60, Minimum = 0, Maximum = 50 };
```

```
chkBeRang = new CheckBox { Text = "B\u00e9 r\u00e1ng (10.000\u00a3/chi\u00e9c)", Location =  
new Point(20, 120), AutoSize = true };
```

```
numBeRang = new NumericUpDown { Location = new Point(350, 120), Width = 60, Minimum = 0, Maximum = 50 };
```

```
chkBocRang = new CheckBox { Text = "B\u00f3c r\u00e1ng (1.000.000\u00a3/chi\u00e9c)", Location =  
new Point(20, 150), AutoSize = true };
```

```
numBocRang = new NumericUpDown { Location = new Point(350, 150), Width = 60, Minimum = 0, Maximum = 50 };
```

```
grpDV.Controls.AddRange(new Control[]
```

```
{
```

```
    chkCaoRang, chkTayTrang,
```

```
    chkHanRang, numHanRang,
```

```
    chkBeRang, numBeRang,
```

```
    chkBocRang, numBocRang
```

```
});
```

```

// Nút thao tác
btnTinh = new Button
{
    Text = "土豪金 Tính tiền",
    Location = new Point(180, 330),
    Size = new Size(120, 40),
    BackColor = Color.LightGreen,
    FlatStyle = FlatStyle.Popup
};

btnTinh.Click += BtnTinh_Click;

btnThoat = new Button
{
    Text = "Thoát",
    Location = new Point(330, 330),
    Size = new Size(120, 40),
    BackColor = Color.LightCoral,
    FlatStyle = FlatStyle.Popup
};

btnThoat.Click += (s, e) => this.Close();

// Kết quả
lblKetQua = new Label
{
    Text = "Tổng tiền: 0 VND",
    Font = new Font("Segoe UI", 12, FontStyle.Bold),
    ForeColor = Color.DarkRed,
    AutoSize = true,
    Location = new Point(180, 390)
};

```

```

        Controls.AddRange(new Control[]
    {
        lblTitle, lblTenKH, txtTenKH,
        grpDV, btnTinh, btnThoat, lblKetQua
    });
}

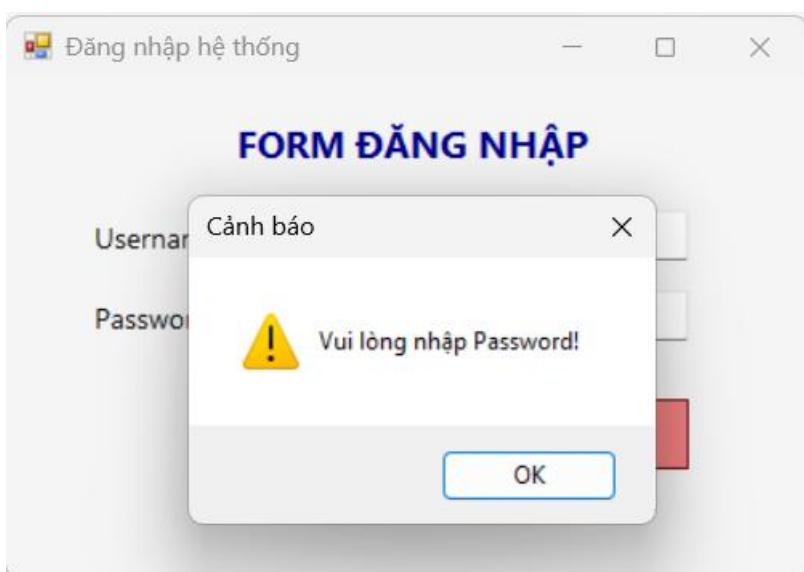
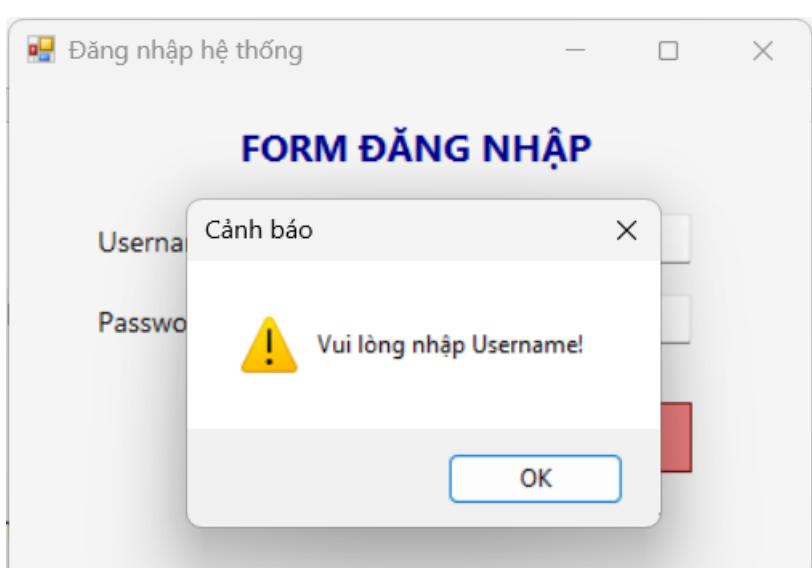
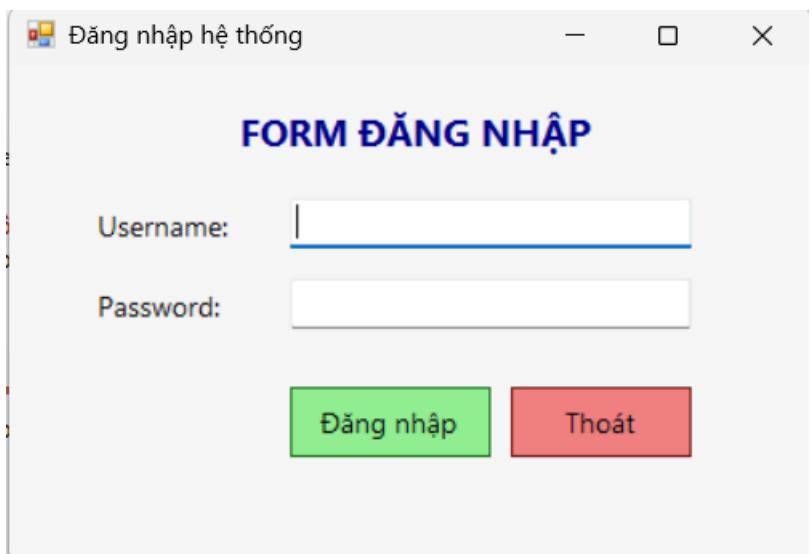
private void BtnTinh_Click(object sender, EventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtTenKH.Text))
    {
        MessageBox.Show("Tên khách hàng không được để trống!", "Cảnh báo",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }

    double tong = 0;
    if (chkCaoRang.Checked) tong += 50000;
    if (chkTayTrang.Checked) tong += 100000;
    if (chkHanRang.Checked) tong += (int)numHanRang.Value * 100000;
    if (chkBeRang.Checked) tong += (int)numBeRang.Value * 10000;
    if (chkBocRang.Checked) tong += (int)numBocRang.Value * 1000000;

    lblKetQua.Text = $"Tổng tiền: {tong:N0} VND";
}
}
}

```

**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.Drawing;  
using System.Windows.Forms;
```

```

namespace Nguyenphutho_1150080158_CNPM2
{
    public class Apdung3 : Form
    {
        private TextBox txtUsername, txtPassword;
        private Button btnLogin, btnThoat;

        public Apdung3()
        {
            InitUI();
        }

        private void InitUI()
        {
            Text = "Đăng nhập hệ thống";
            StartPosition = FormStartPosition.CenterScreen;
            ClientSize = new Size(400, 250);
            BackColor = Color.WhiteSmoke;
            Font = new Font("Segoe UI", 10);

            // Tiêu đề
            Label lblTitle = new Label
            {
                Text = "FORM ĐĂNG NHẬP",
                Font = new Font("Segoe UI", 14, FontStyle.Bold),
                ForeColor = Color.DarkBlue,
                AutoSize = true,
                Location = new Point(110, 20)
            };

            // Username

```

```

Label lblUser = new Label { Text = "Username:", Location = new Point(40, 70),
AutoSize = true };

txtUsername = new TextBox { Location = new Point(140, 66), Width = 200 };

// Password

Label lblPass = new Label { Text = "Password:", Location = new Point(40, 110),
AutoSize = true };

txtPassword = new TextBox
{
    Location = new Point(140, 106),
    Width = 200,
    UseSystemPasswordChar = true
};

// Nút Đăng nhập

btnLogin = new Button
{
    Text = "Đăng nhập",
    Location = new Point(140, 160),
    Size = new Size(100, 35),
    BackColor = Color.LightGreen,
    FlatStyle = FlatStyle.Popup
};

btnLogin.Click += BtnLogin_Click;

// Nút Thoát

btnThoat = new Button
{
    Text = "Thoát",
    Location = new Point(250, 160),
    Size = new Size(90, 35),
    BackColor = Color.LightCoral,
};

```

```

        FlatStyle = FlatStyle.Popup
    };
    btnThoat.Click += (s, e) => this.Close();

    Controls.AddRange(new Control[] { lblTitle, lblUser, txtUsername, lblPass,
txtPassword, btnLogin, btnThoat });
}

private void BtnLogin_Click(object sender, EventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtUsername.Text))
    {
        MessageBox.Show("Vui lòng nhập Username!", "Cảnh báo",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtUsername.Focus();
        return;
    }

    if (string.IsNullOrWhiteSpace(txtPassword.Text))
    {
        MessageBox.Show("Vui lòng nhập Password!", "Cảnh báo",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtPassword.Focus();
        return;
    }

    // Nếu muốn, có thể kiểm tra user/pass cố định
    if (txtUsername.Text == "admin" && txtPassword.Text == "123")
    {
        MessageBox.Show("Đăng nhập thành công!", "Thông báo",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

```

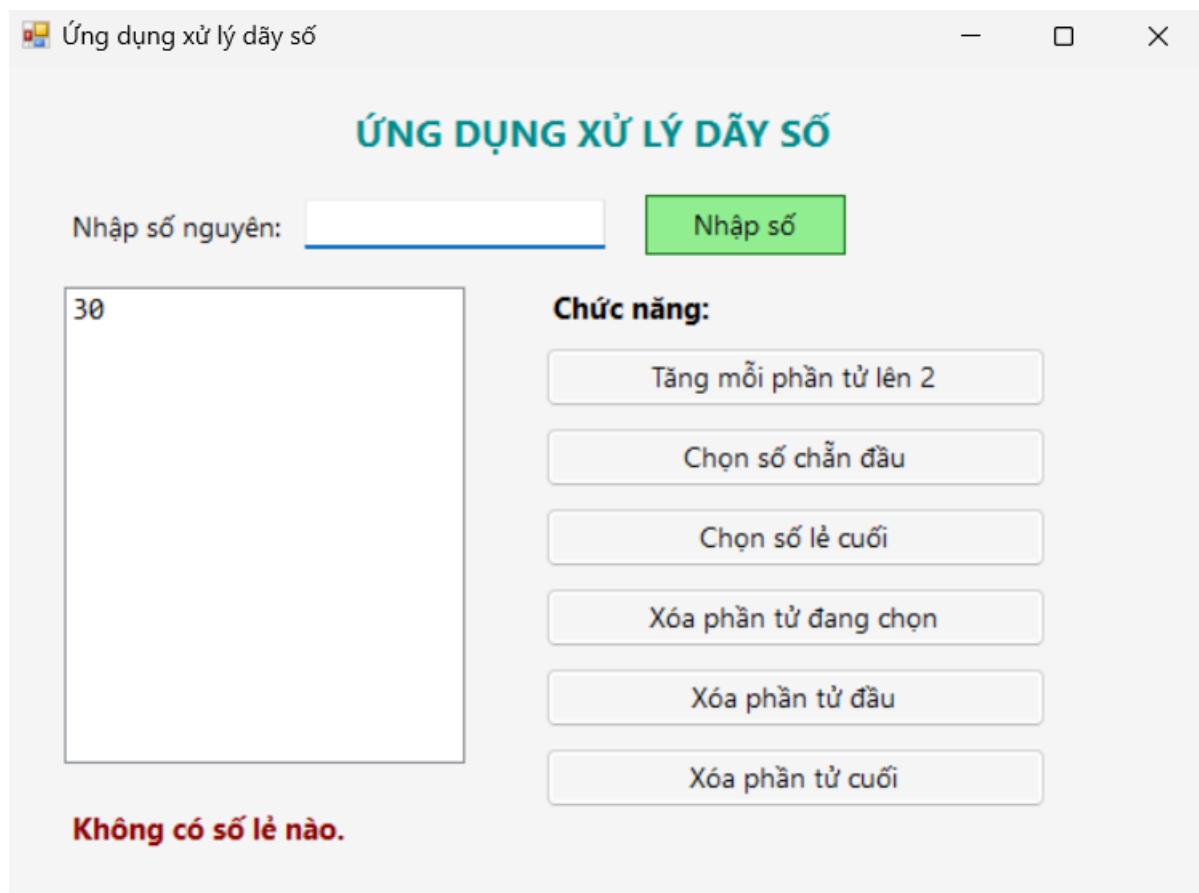
else
{
    MessageBox.Show("Sai Username hoặc Password!", "Lỗi",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}
}
}
}

```

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

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.



Ứng dụng xử lý dãy số

## ỨNG DỤNG XỬ LÝ DÃY SỐ

Nhập số nguyên:  Nhập số

32

**Chức năng:**

- Tăng mỗi phần tử lên 2
- Chọn số chẵn đầu
- Chọn số lẻ cuối
- Xóa phần tử đang chọn
- Xóa phần tử đầu
- Xóa phần tử cuối

**Không có số lẻ nào.**

Ứng dụng xử lý dãy số

## ỨNG DỤNG XỬ LÝ DÃY SỐ

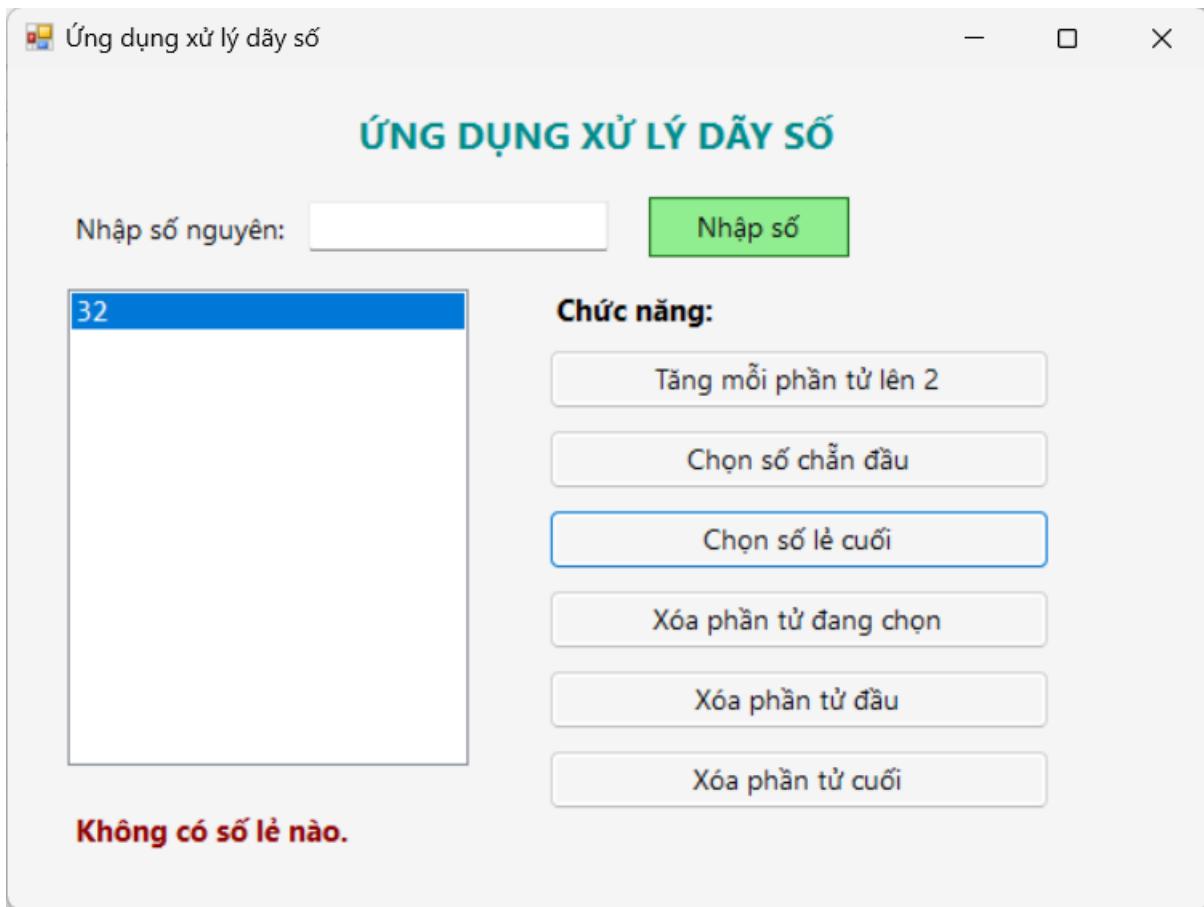
Nhập số nguyên:  Nhập số

32

**Chức năng:**

- Tăng mỗi phần tử lên 2
- Chọn số chẵn đầu
- Chọn số lẻ cuối
- Xóa phần tử đang chọn
- Xóa phần tử đầu
- Xóa phần tử cuối

**Số chẵn đầu tiên: 32**



```
using System;
using System.Drawing;
using System.Linq;
using System.Windows.Forms;
```

```
namespace Nguyenphutho_1150080158_CNPM2
{
    public class ThucHanh3 : Form
    {
        private TextBox txtNhap;
        private Button btnNhap, btnTang2, btnChonChanDau, btnChonLeCuo,
                      btnXoaChon, btnXoaDau, btnXoaCuoi;
        private ListBox lstDaySo;
        private Label lblKetQua;
```

```
        public ThucHanh3()
        {
```

```

    InitUI();
}

private void InitUI()
{
    Text = "Ứng dụng xử lý dãy số";
    StartPosition = FormStartPosition.CenterScreen;
    ClientSize = new Size(600, 420);
    BackColor = Color.WhiteSmoke;
    Font = new Font("Segoe UI", 10);

    // Tiêu đề
    Label lblTitle = new Label
    {
        Text = "ỨNG DỤNG XỬ LÝ DÃY SỐ",
        Font = new Font("Segoe UI", 14, FontStyle.Bold),
        ForeColor = Color.DarkCyan,
        AutoSize = true,
        Location = new Point(170, 20)
    };

    // Nhập số
    Label lblNhap = new Label { Text = "Nhập số nguyên:", Location = new Point(30, 70), AutoSize = true };
    txtNhap = new TextBox { Location = new Point(150, 66), Width = 150 };
    txtNhap.KeyPress += TxtNhap_KeyPress;
    txtNhap.KeyDown += TxtNhap_KeyDown;

    btnNhap = new Button
    {
        Text = "Nhập số",
        Location = new Point(320, 64),

```

```

        Size = new Size(100, 30),
        BackColor = Color.LightGreen,
        FlatStyle = FlatStyle.Popup
    };
    btnNhap.Click += BtnNhap_Click;

    // ListBox
    lstDaySo = new ListBox
    {
        Location = new Point(30, 110),
        Size = new Size(200, 250),
        Font = new Font("Consolas", 11)
    };

    // Chức năng
    Label lblCN = new Label { Text = "Chức năng:", Location = new Point(270,
    110), AutoSize = true, Font = new Font("Segoe UI", 11, FontStyle.Bold) };

    btnTang2 = new Button { Text = "Tăng mỗi phần tử lên 2", Location = new
    Point(270, 140), Size = new Size(250, 30) };
    btnTang2.Click += BtnTang2_Click;

    btnChonChanDau = new Button { Text = "Chọn số chẵn đầu", Location = new
    Point(270, 180), Size = new Size(250, 30) };
    btnChonChanDau.Click += BtnChonChanDau_Click;

    btnChonLeCuoi = new Button { Text = "Chọn số lẻ cuối", Location = new
    Point(270, 220), Size = new Size(250, 30) };
    btnChonLeCuoi.Click += BtnChonLeCuoi_Click;

    btnXoaChon = new Button { Text = "Xóa phần tử đang chọn", Location = new
    Point(270, 260), Size = new Size(250, 30) };

```

```

btnXoaChon.Click += BtnXoaChon_Click;

btnXoaDau = new Button { Text = "Xóa phần tử đầu", Location = new Point(270, 300), Size = new Size(250, 30) };
btnXoaDau.Click += (s, e) => { if (lstDaySo.Items.Count > 0) lstDaySo.Items.RemoveAt(0); };

btnXoaCuoi = new Button { Text = "Xóa phần tử cuối", Location = new Point(270, 340), Size = new Size(250, 30) };
btnXoaCuoi.Click += (s, e) => { if (lstDaySo.Items.Count > 0) lstDaySo.Items.RemoveAt(lstDaySo.Items.Count - 1); };

lblKetQua = new Label { Text = "Kết quả:", Location = new Point(30, 370), AutoSize = true, ForeColor = Color.DarkRed, Font = new Font("Segoe UI", 11, FontStyle.Bold) };

Controls.AddRange(new Control[]
{
    lblTitle, lblNhap, txtNhap, btnNhap,
    lstDaySo, lblCN, btnTang2, btnChonChanDau,
    btnChonLeCuoi, btnXoaChon, btnXoaDau, btnXoaCuoi,
    lblKetQua
});

private void TxtNhap_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar) && e.KeyChar != '-') e.Handled = true;
}

private void TxtNhap_KeyDown(object sender, KeyEventArgs e)

```

```

    {
        if (e.KeyCode == Keys.Enter)
        {
            AddSo();
            e.SuppressKeyPress = true;
        }
    }

private void BtnNhap_Click(object sender, EventArgs e) => AddSo();

private void AddSo()
{
    if (int.TryParse(txtNhap.Text, out int so))
    {
        lstDaySo.Items.Add(so);
        txtNhap.Clear();
        txtNhap.Focus();
    }
    else
    {
        MessageBox.Show("Vui lòng nhập số nguyên hợp lệ!");
    }
}

private void BtnTang2_Click(object sender, EventArgs e)
{
    for (int i = 0; i < lstDaySo.Items.Count; i++)
    {
        int val = (int)lstDaySo.Items[i];
        lstDaySo.Items[i] = val + 2;
    }
}

```

```

private void BtnChonChanDau_Click(object sender, EventArgs e)
{
    for (int i = 0; i < lstDaySo.Items.Count; i++)
    {
        if ((int)lstDaySo.Items[i] % 2 == 0)
        {
            lstDaySo.SelectedIndex = i;
            lblKetQua.Text = $"Số chẵn đầu tiên: {lstDaySo.Items[i]}";
            return;
        }
    }
    lblKetQua.Text = "Không có số chẵn nào.";
}

```

```

private void BtnChonLeCuoi_Click(object sender, EventArgs e)
{
    for (int i = lstDaySo.Items.Count - 1; i >= 0; i--)
    {
        if ((int)lstDaySo.Items[i] % 2 != 0)
        {
            lstDaySo.SelectedIndex = i;
            lblKetQua.Text = $"Số lẻ cuối cùng: {lstDaySo.Items[i]}";
            return;
        }
    }
    lblKetQua.Text = "Không có số lẻ nào.";
}

```

```

private void BtnXoaChon_Click(object sender, EventArgs e)
{
    if (lstDaySo.SelectedIndex >= 0)

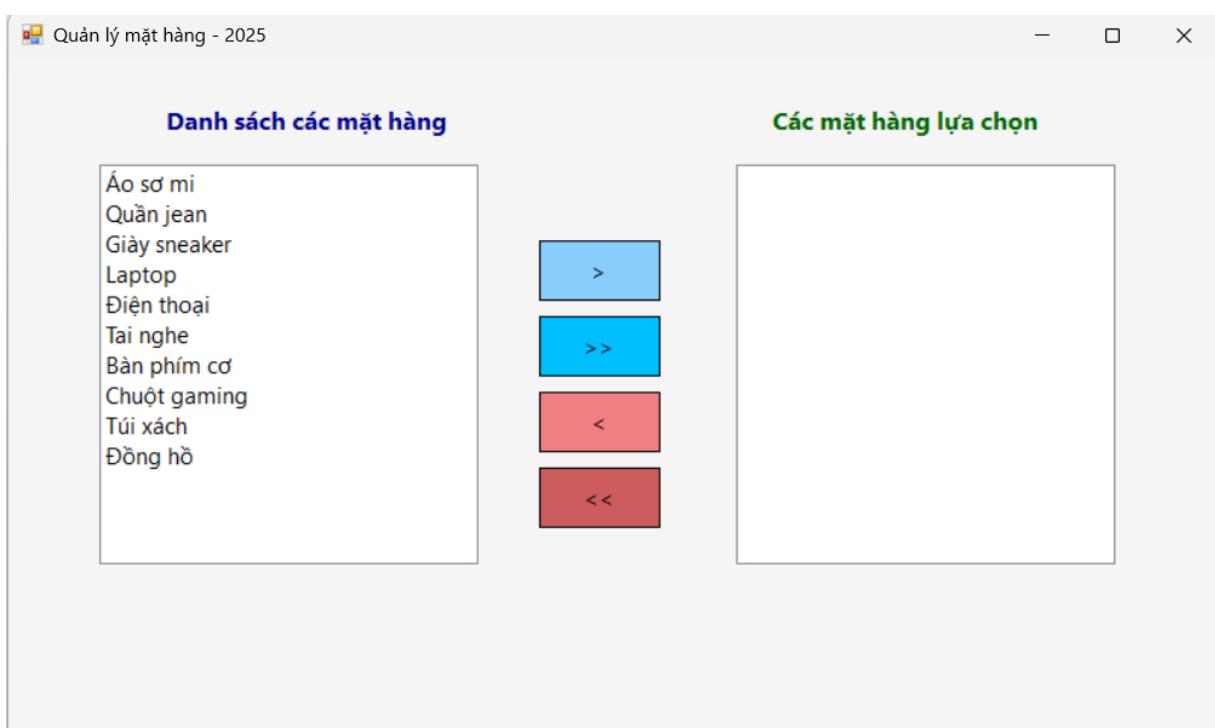
```

```

{
    lstDaySo.Items.RemoveAt(lstDaySo.SelectedIndex);
}
else
{
    MessageBox.Show("Hãy chọn phần tử để xóa!");
}
}
}

```

#### Áp dụng 4: làm bài tập 7



```

using System;
using System.Drawing;
using System.Linq;
using System.Windows.Forms;

```

```
namespace Nguyenphutho_1150080158_CNPM2
```

```
{
    public class ThucHanh4 : Form
{
}
```

```

private ListBox lstMatHang, lstLuaChon;
private Button btnChuyen1, btnChuyenAll, btnTra1, btnTraAll;
private Label lblMH, lblLC;

public ThucHanh4()
{
    InitUI();
    LoadData();
}

private void InitUI()
{
    Text = "Quản lý mặt hàng - 2025";
    StartPosition = FormStartPosition.CenterScreen;
    ClientSize = new Size(800, 450);
    BackColor = Color.WhiteSmoke;
    Font = new Font("Segoe UI", 11);

    lblMH = new Label
    {
        Text = "Danh sách các mặt hàng",
        Location = new Point(100, 30),
        AutoSize = true,
        Font = new Font("Segoe UI", 12, FontStyle.Bold),
        ForeColor = Color.DarkBlue
    };

    lblLC = new Label
    {
        Text = "Các mặt hàng lựa chọn",
        Location = new Point(500, 30),
        AutoSize = true,
    };
}

```

```

Font = new Font("Segoe UI", 12, FontStyle.Bold),
ForeColor = Color.DarkGreen
};

lstMatHang = new ListBox
{
    Location = new Point(60, 70),
    Size = new Size(250, 280),
    Font = new Font("Segoe UI", 11),
    SelectionMode = SelectionMode.MultiExtended
};

lstLuaChon = new ListBox
{
    Location = new Point(480, 70),
    Size = new Size(250, 280),
    Font = new Font("Segoe UI", 11),
    SelectionMode = SelectionMode.MultiExtended
};

// Các nút điều khiển
btnChuyen1 = new Button
{
    Text = ">",
    Location = new Point(350, 120),
    Size = new Size(80, 40),
    BackColor = Color.LightSkyBlue,
    FlatStyle = FlatStyle.Flat
};

btnChuyen1.Click += BtnChuyen1_Click;

btnChuyenAll = new Button

```

```
{  
    Text = ">>",  
    Location = new Point(350, 170),  
    Size = new Size(80, 40),  
    BackColor = Color.DeepSkyBlue,  
    FlatStyle = FlatStyle.Flat  
};  
btnChuyenAll.Click += BtnChuyenAll_Click;
```

```
btnTra1 = new Button  
{  
    Text = "<",  
    Location = new Point(350, 220),  
    Size = new Size(80, 40),  
    BackColor = Color.LightCoral,  
    FlatStyle = FlatStyle.Flat  
};  
btnTra1.Click += BtnTra1_Click;
```

```
btnTraAll = new Button  
{  
    Text = "<<",  
    Location = new Point(350, 270),  
    Size = new Size(80, 40),  
    BackColor = Color.IndianRed,  
    FlatStyle = FlatStyle.Flat  
};  
btnTraAll.Click += BtnTraAll_Click;
```

```
Controls.AddRange(new Control[]  
{  
    lblMH, lblLC, lstMatHang, lstLuaChon,
```

```

        btnChuyen1, btnChuyenAll, btnTra1, btnTraAll
    });
}

private void LoadData()
{
    string[] hang = { "Áo sơ mi", "Quần jean", "Giày sneaker", "Laptop", "Điện
thoại", "Tai nghe", "Bàn phím cơ", "Chuột gaming", "Túi xách", "Đồng hồ" };
    lstMatHang.Items.AddRange(hang);
}

private void BtnChuyen1_Click(object sender, EventArgs e)
{
    var selected = lstMatHang.SelectedItems.Cast<object>().ToList();
    foreach (var item in selected)
    {
        lstLuaChon.Items.Add(item);
    }
    foreach (var item in selected)
    {
        lstMatHang.Items.Remove(item);
    }
}

private void BtnChuyenAll_Click(object sender, EventArgs e)
{
    lstLuaChon.Items.AddRange(lstMatHang.Items);
    lstMatHang.Items.Clear();
}

private void BtnTra1_Click(object sender, EventArgs e)
{
}

```

```
var selected = lstLuaChon.SelectedItems.Cast<object>().ToList();
foreach (var item in selected)
{
    lstMatHang.Items.Add(item);
}
foreach (var item in selected)
{
    lstLuaChon.Items.Remove(item);
}
}

private void BtnTraAll_Click(object sender, EventArgs e)
{
    lstMatHang.Items.AddRange(lstLuaChon.Items);
    lstLuaChon.Items.Clear();
}
}
```