

Tên: Trần Dương Minh Đại

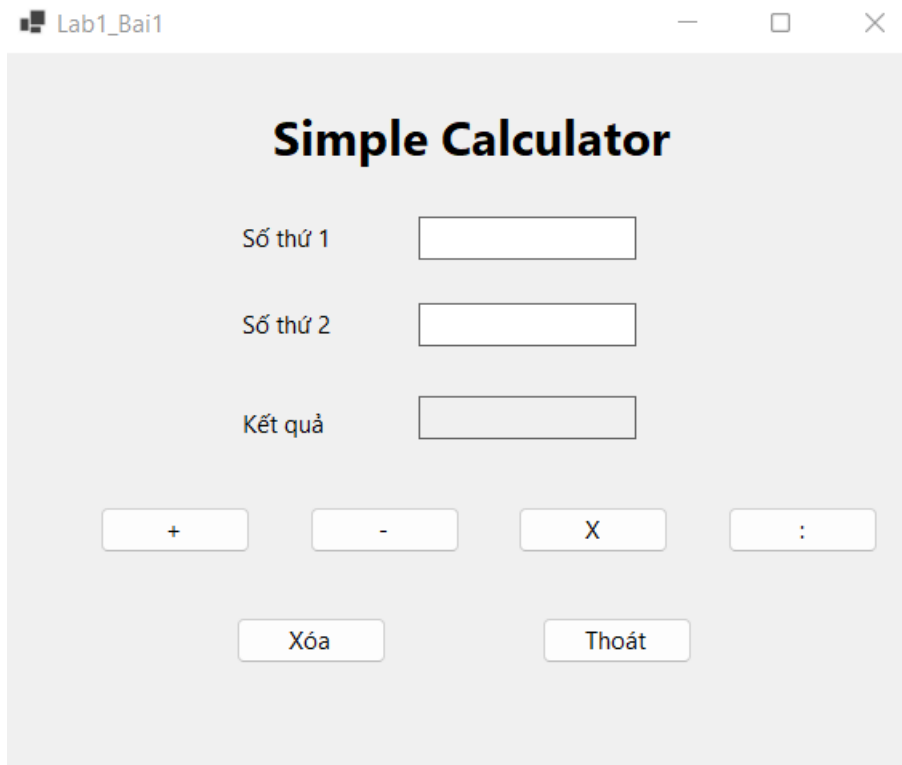
MSSV: 22520183

## Báo cáo Lab 1

### Lập trình mạng căn bản

#### 1. Chương trình tính toán đơn giản

- Screenshot bài 1:



- Đầu tiên, để đảm bảo dữ liệu hợp lệ ta cần check xem input có trống hay null ko bằng việc gọi method `IsNullorEmpty` của string .

```
if (string.IsNullOrEmpty(textBox1.Text))
{
    MessageBox.Show("Vui lòng nhập một số!");
    return;
}
```

- Sau đó là check input có phải là số nguyên hay không dựa vào method `TryParse`. Đồng thời check output có overflow hay không ở 2 phép cộng và nhân bằng cách catch exception của class `OverflowException`.

```

if (Int32.TryParse(textBox1.Text.Trim(), out int num1) && Int32.TryParse(textBox2.Text.Trim(), ou
{
    try
    {
        checked
        {
            int mul;
            mul = num1 * num2;
            textBox3.Text = mul.ToString();
        }
    }
    catch (OverflowException ex)
    {
        MessageBox.Show(ex.Message);
    }
}
else
{
    MessageBox.Show("Vui lòng nhập 2 số nguyên!");
}

```

+ Đối với các button +,-,x,/, chúng ta thực hiện các logic cơ bản. Với chức năng thoát, ta sẽ sử dụng phương thức Close(), còn chức năng xóa thì dùng phương thức Clear().

```

1 reference
private void Exit_Click(object sender, EventArgs e)
{
    this.Close();
}

1 reference
private void Clear_Click(object sender, EventArgs e)
{
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
}

```

- Kết quả:

Lab1\_Bai1

## Simple Calculator

Số thứ 1

Số thứ 2

Kết quả

Lab1\_Bai1

## Simple Calculator

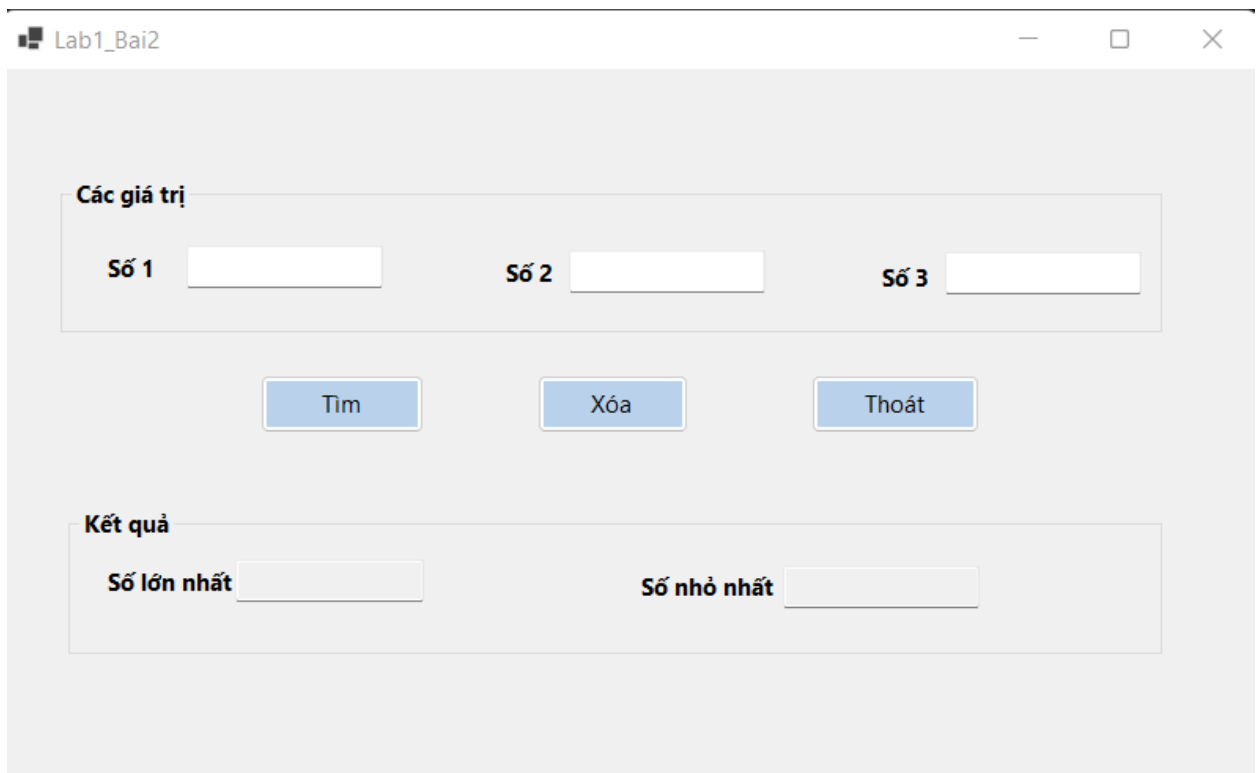
Số thứ 1

Số thứ 2

Kết quả

## 2. Số lớn nhất, nhỏ nhất

- Screenshot bài 2:



- Ta tiếp tục kiểm tra dữ liệu có trống hay không cũng như dữ liệu có đúng là số thực hoặc số nguyên hay không bằng 2 method isEmptyOrNull và TryParse

```
if (string.IsNullOrEmpty(textBox1.Text)&&string.IsNullOrEmpty(textBox2.Text)&&string.IsNullOrEmpty(textBox3.Text))
{
    MessageBox.Show("Vui lòng nhập một số!");
    return;
}
```

- Để tìm số lớn nhất ta xài Math.max và số nhỏ nhất thì dùng Math.min

```
if (float.TryParse(textBox1.Text.Trim(),out float num1)&&
float.TryParse(textBox2.Text.Trim(),out float num2)&&
float.TryParse(textBox3.Text.Trim(),out float num3)
)
{
    float max = Math.Max(Math.Max(num1, num2), num3);
    float min = Math.Min(Math.Min(num1,num2),num3);
    max_textbox.Text = max.ToString();
    min_textbox.Text = min.ToString();
}
else
{
    MessageBox.Show("Vui lòng nhập số nguyên hoặc số thực!", "Thông báo", MessageBoxButtons.OK);
}
```

- Kết quả:

Lab1\_Bai2

**Các giá trị**

Số 1  Số 2  Số 3

**Kết quả**

Số lớn nhất  Số nhỏ nhất

### 3. Đọc số ( từ 0 đến 9 )

- Screenshot bài 3:

Lab1\_Bai3

**Đọc số (0-9)**

Nhập số

Kết quả

- Ta sẽ sử dụng switch case để xử lý, nếu input là 1 thì hàm NumberToWord sẽ trả về string ‘Một’ và cứ tiếp tục như vậy cho tới số 9.

```

private string NumberToWord(int number)
{
    switch (number)
    {
        case 0:
            return "Không";
        case 1:
            return "Một";
        case 2:
            return "Hai";
        case 3:
            return "Ba";
        case 4:
            return "Bốn";
        case 5:
            return "Năm";
        case 6:
            return "Sáu";
        case 7:
            return "Bảy";
        case 8:
            return "Tám";
        case 9:
            return "Chín";
        default:
            return "";
    }
}

```

- Tính hợp lệ của dữ liệu sẽ được kiểm tra input có từ  $0 \leq n \leq 9$  cũng như bắt buộc là số nguyên

```

private void read_button_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(textBox1.Text))
    {
        MessageBox.Show("Vui lòng nhập một số!");
        return;
    }

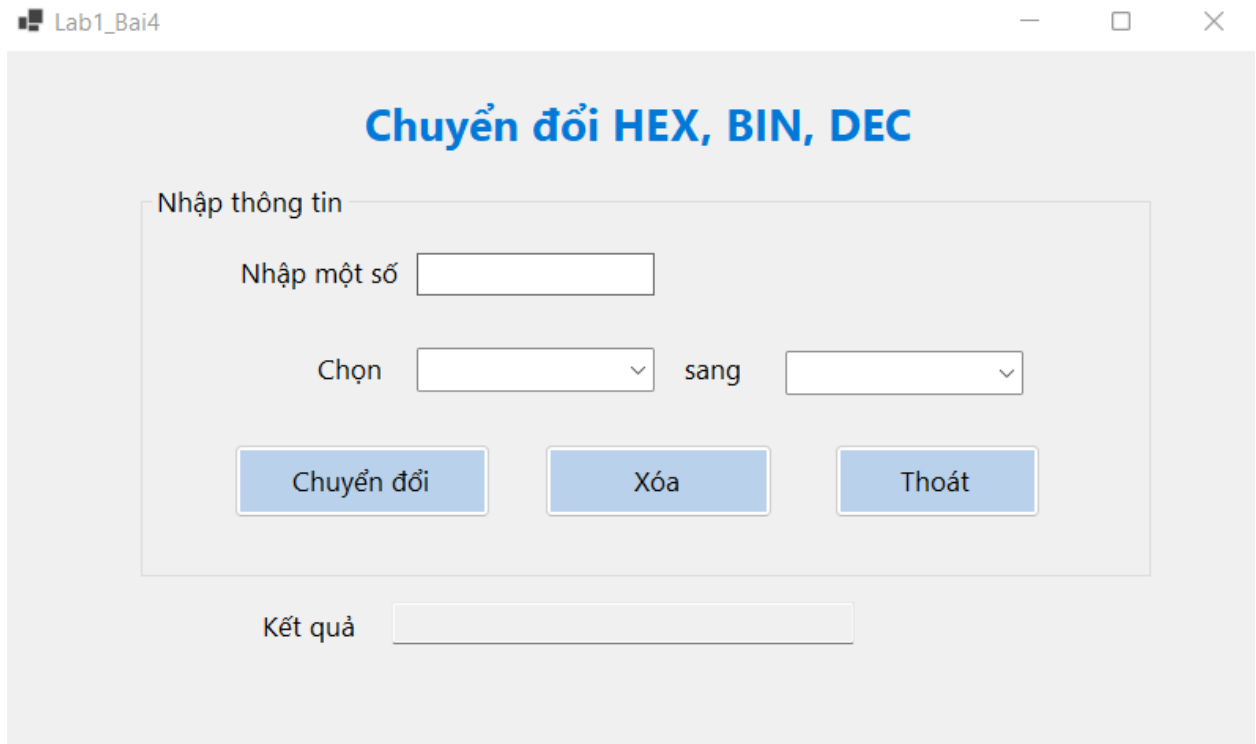
    if (int.TryParse(textBox1.Text.Trim(), out int number) && number >= 0 && number <= 9)
    {
        textBox2.Text = NumberToWord(number).ToString();
    }
    else
    {
        MessageBox.Show("Vui lòng nhập một số nguyên từ 0 đến 9.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }
}

```

- Kết quả:

#### 4. Đổi hệ cơ số

- Screenshot bài 4:



Lab1\_Bai4

### Chuyển đổi HEX, BIN, DEC

Nhập thông tin

Nhập một số

Chọn  sang

Kết quả

- Trước tiên ta check dữ liệu hợp lệ bằng try catch và truyền tham số Exception ex. Từ đó nếu user có nhập dữ liệu không nằm trong range của kiểu dữ liệu cần chuyển hay dữ liệu nhập vào không đúng loại dữ liệu cần chuyển ( ví dụ nhập 25 mà chọn chuyển từ binary ) -> thì sẽ hiện ra thông báo lỗi.
- Đầu tiên, nếu user chuyển cùng kiểu dữ liệu như binary – binary thì ta vẫn sẽ output ra số input đã nhập. Tiếp theo, nếu user chuyển từ binary – dec hay hex thì ta sẽ dùng

```
result = Convert.ToInt64(inputValue, 2).ToString();
```

( Với '2' là chuyển từ hệ 2 binary)

Với binary – hex:

```
result = Convert.ToString(Convert.ToInt64(inputValue, 2), 16).ToUpper();
```

Tương tự với các trường hợp còn lại.

- Kết quả:

## Chuyển đổi HEX, BIN, DEC

Nhập thông tin

Nhập một số

Chọn  sang

Kết quả

### 5. Quản lý điểm

- Screenshot bài 5:



Lab1\_Bai5

## Quản lý điểm

Danh sách điểm

Danh sách môn học và điểm

Kết quả

- Đầu tiên ta sẽ khai báo mảng scores để lưu chuỗi string điểm từ textBox và mảng numericScore để lưu điểm dưới dạng số.

```
//Khai báo
string input = textBox1.Text;
string[] scores = input.Split(' ');
double[] numericScores = new double[scores.Length];
//Check xem đã có dữ liệu chưa
```

- Sau đó, ta sẽ check dữ liệu hợp lệ bằng hàm TryParse cũng như xem điểm nhập vào có nằm trong khoảng từ 0 đến 10 hay không. Tiếp theo, ta sẽ appendText vào richtextbox.

```
//Check xem đã có dữ liệu chưa
if (string.IsNullOrEmpty(input))
{
    MessageBox.Show("Vui lòng nhập điểm!");
    return;
}
for (int i = 0; i < scores.Length; i++)
{
    if (double.TryParse(scores[i], out numericScores[i]) && numericScores[i] <= 10 && numericScores[i] >= 0)
    {
        richTextBox1.AppendText($"Môn {i + 1}: {numericScores[i]}đ \t");
    }
    else
    {
        MessageBox.Show("Điểm nhập không hợp lệ!");
        richTextBox1.Clear();
        return;
    }
}
```

- Cuối cùng là output kết quả điểm vào richTextBox với
- + Điểm cao nhất, nhỏ nhất, trung bình dùng method Max, Min, Average

```
double averageScore = numericScores.Average();
double maxScore = numericScores.Max();
double minScore = numericScores.Min();
int pass = numericScores.Count(score => score >= 5);
int notPass = numericScores.Count(score => score < 5);
```

- + Số môn đậu và không đậu dùng method Count và truyền vào điều kiện môn đậu, không đậu

```
int pass = numericScores.Count(score => score >= 5);
int notPass = numericScores.Count(score => score < 5);
```

- + Học lực ta tạo hàm XepLoai để phân loại như sau:

```
private string XepLoai(double averageScore, double[] numericScores)
{
    if (averageScore >= 8 && !numericScores.Any(score => score < 6.5))
    {
        return "Giỏi";
    }
    else if (averageScore >= 6.5 && !numericScores.Any(score => score < 5))
    {
        return "Khá";
    }
    else if (averageScore >= 5 && !numericScores.Any(score => score < 3.5))
    {
        return "Trung bình";
    }
    else if (averageScore >= 3.5 && !numericScores.Any(score => score < 2))
    {
        return "Yếu";
    }
    else return "Kém";
}
```

- Append vào richTextBox:

```
richTextBox2.AppendText($"Điểm trung bình: {averageScore.ToString("F2")} \n \n");
richTextBox2.AppendText($"Điểm cao nhất: {maxScore} \n \n ");
richTextBox2.AppendText($"Điểm nhỏ nhất: {minScore} ");

richTextBox3.AppendText($"Xếp loại học lực: {XepLoai(averageScore, numericScores)} \n \n");
richTextBox3.AppendText($"Số môn đậu: {pass} \n \n");
richTextBox3.AppendText($"Số môn không đậu: {notPass} ");
}
```

- Kết quả:

## Quản lý điểm

Danh sách điểm 9 8 6 7 9,5 10 8,9 7 10

Xuất

Xóa

Thoát

Danh sách môn học và điểm

Môn 1: 9đ	Môn 2: 8đ	Môn 3: 6đ	Môn 4: 7đ	Môn 5:
9,5đ	Môn 6: 10đ	Môn 7: 8,9đ	Môn 8: 7đ	Môn 9: 10đ

Kết quả

Điểm trung bình: 8,38

Xếp loại học lực: Khá

Điểm cao nhất: 10

Số môn đậu: 9

Điểm nhỏ nhất: 6

Số môn không đậu: 0