

Trường ĐH CT TP. HCM Khoa: CNTT Bộ môn: CNPM Môn: Lập trình .NET	<h1 style="text-align: center;">BÀI 3</h1> <h2 style="text-align: center;">GIAO DIỆN</h2>	
---	---	--

### A. MỤC TIÊU:

- Thiết kế được giao diện Windows Form từ các control trong C#.
- Sử dụng được các Controls Textbox, Label, Button.

### B. DỤNG CỤ - THIẾT BỊ THỰC HÀNH CHO MỘT SV:

STT	Chủng loại – Quy cách vật tư	Số lượng	Đơn vị	Ghi chú
1	Computer	1	1	

### C. NỘI DUNG THỰC HÀNH

#### 1. Cơ sở lý thuyết

##### 1.1. Kiến thức cần nhớ

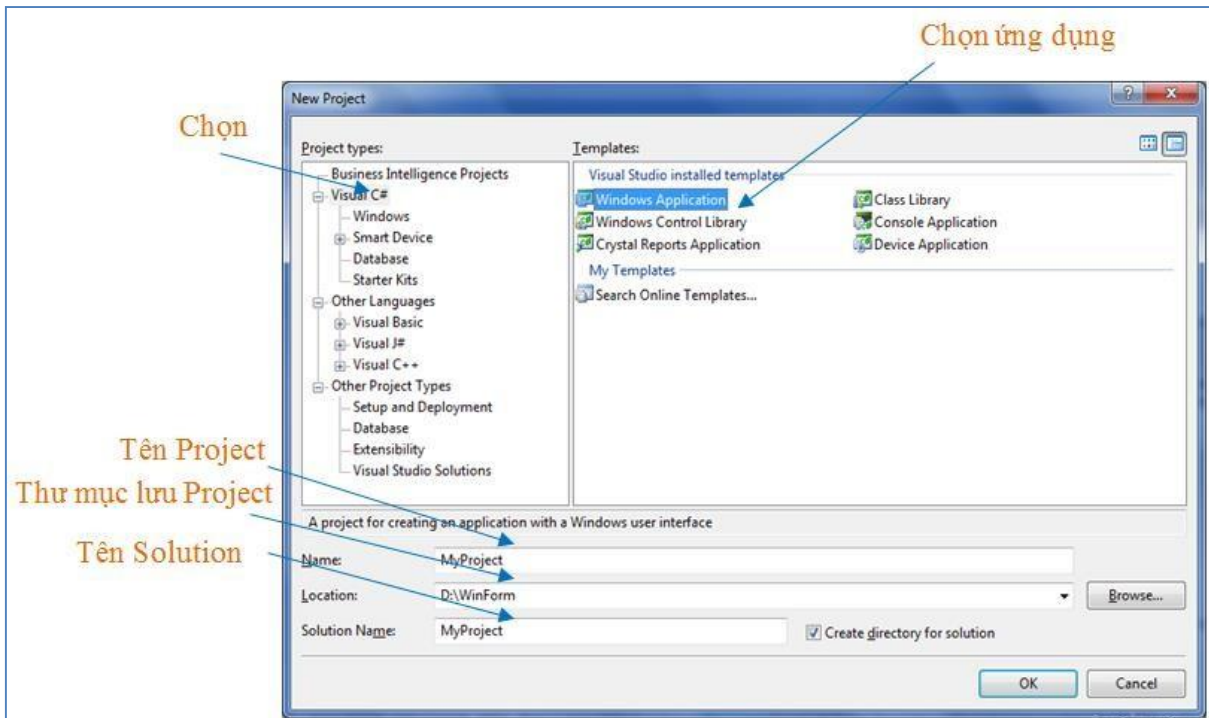
Mỗi Solution hoặc Project có các loại tập tin sau:

File Name	Diễn giải
MyName.sln	Đây là file solution (file text) Lưu trữ thông tin về solution và project. Nếu project đã có sẵn thì ta sẽ mở file này để tiếp tục project.
frmMyName.cs	Lưu trữ code trong form.
frmMyName.Designer.cs	Lưu trữ code design trong form.
frmMyName.rexs	Đây là file resource, định nghĩa tất cả system resources sử dụng trong form.
MyName.csproj	File project này mô tả thông tin project và danh sách các file có trong project.
Program.cs	Là tập tin tự phát sinh khi tạo project và cũng là tập tin bắt đầu một ứng dụng.
Các files khác	Sau khi thực thi một project thì một số file khác sẽ tự động phát sinh trong thư mục solution.

## Tạo một ứng dụng Windows Form

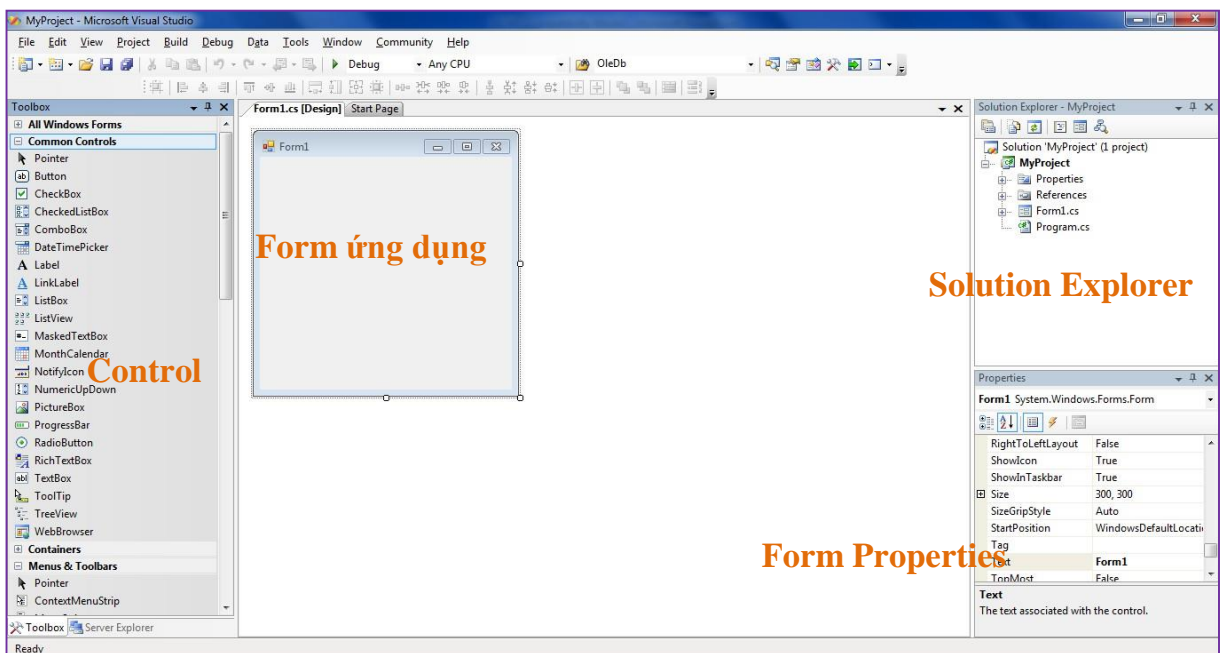
Khởi động chương trình: Start → All program → Microsoft Visual Studio 2012 → Microsoft Visual Studio 2012.

Tạo một project mới: File → New → Project



Hình 3. Cửa sổ tạo projec mới

Sau khi chọn ngôn ngữ, ứng dụng, đặt tên cho project, ... click OK, cửa sổ thiết kế giao diện (Form Design) sẽ hiện ra:

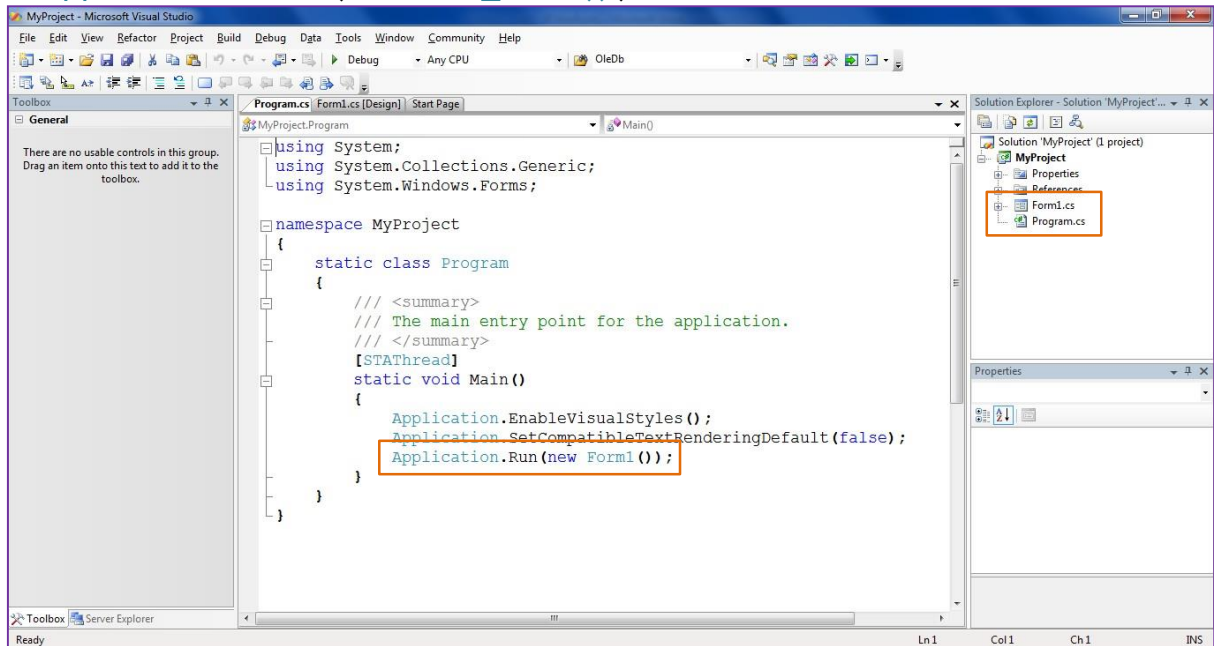


Hình 4. Cửa sổ thiết kế giao diện

### Lưu ý:

1. Trong cùng một project có thể có nhiều form, muốn chọn form khởi động đầu tiên ta mở file Program.cs và thay tên form cần mở trong lệnh

`Application.Run(new Ten_form());`

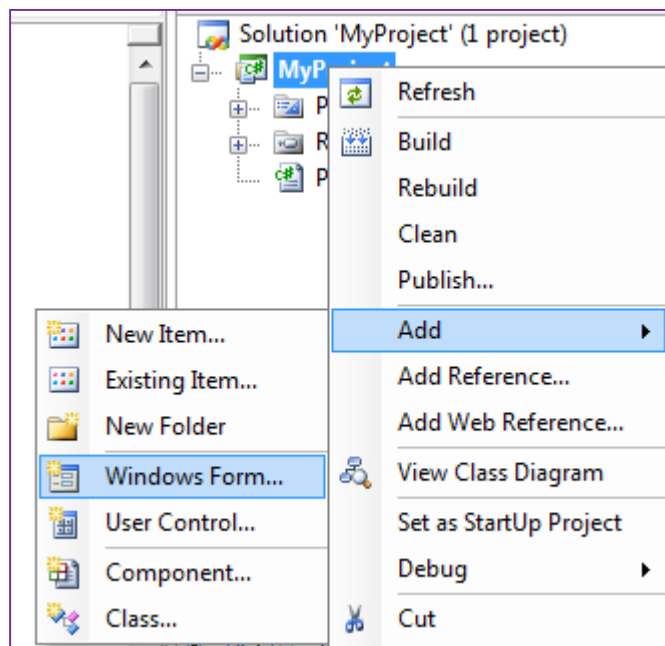


Hình 5. Tập tin Program.cs

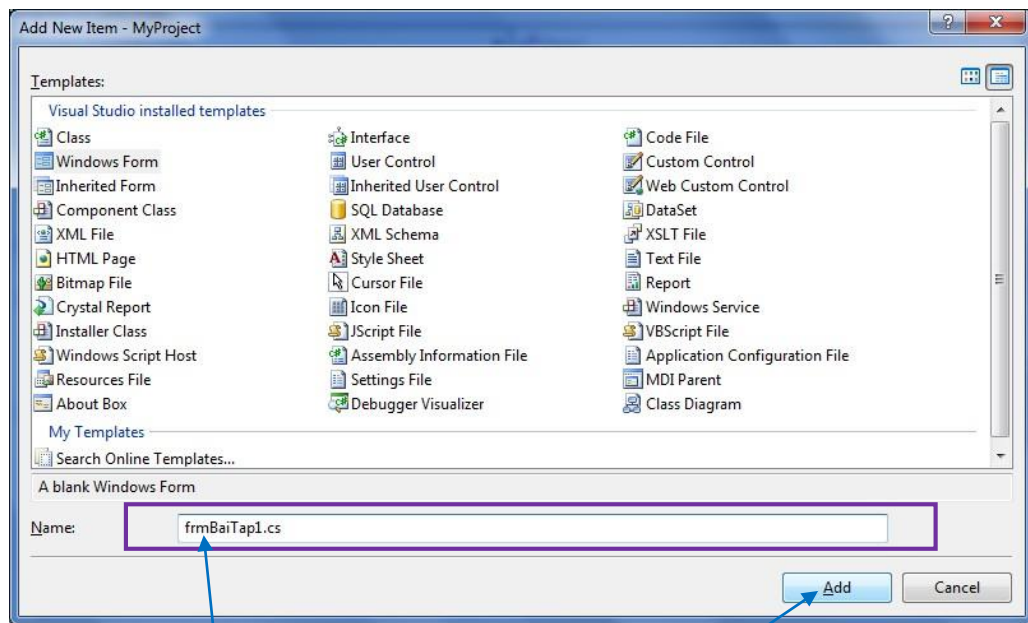
2. Trước khi làm bài tập thì nên xóa Form1 (form được tạo mặc định), tạo Form mới và đặt tên theo yêu cầu. Cách thực hiện như sau:

Trong cửa sổ Solution Explorer, Click chuột phải vào Form1 → Delete

Sau đó Click chuột phải trên MyProject → Add → Windows Form...



Cửa sổ **Add New Item** xuất hiện, đặt tên cho form mới và chọn Add.



Đặt tên Form

Click Add Form

Hình 6. Cửa sổ Add New Item vào My Project

## 1.2. Giới thiệu bài tập mẫu

Thiết kế form sau:

**Yêu cầu:**

- Khi Form khởi động thực hiện nhập tên, năm sinh vào TextBox YourName và Year of Birth tương ứng. Nếu YourName không được nhập dữ liệu, năm sinh không phải là số thì thông báo lỗi (dùng **ErrorProvider**).
- **Errorpvider**: Cung cấp thông tin lỗi của điều khiển trên Form, thường được dùng với control input (ví dụ: TextBox) ràng buộc với một điều kiện nhập.

- Khi người dùng nhấn button Show sẽ hiển thị thông tin nhập vào MessageBox bao gồm: tên, tuổi (năm hiện tại – năm sinh)



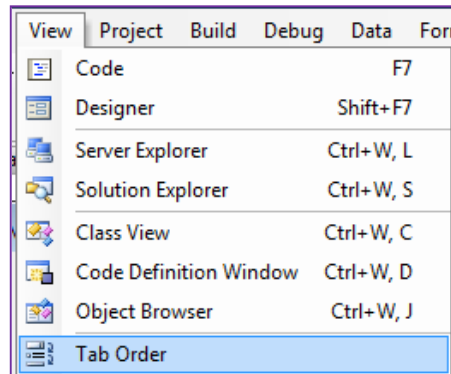
- Khi người dùng nhấn button Clear sẽ xóa thông tin đã nhập trên các TextBox, đặt con trỏ văn bản vào Textbox YourName.
- Khi nhấn button Exit xác nhận người dùng có thực sự muốn thoát khỏi chương trình không? Yes: thoát, No: không.

#### Hướng dẫn:

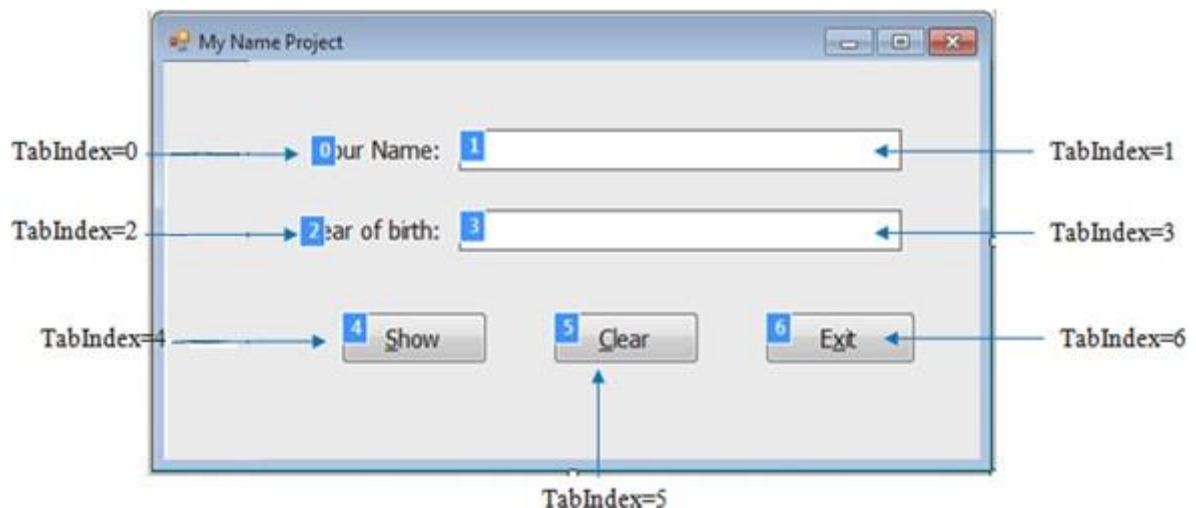
- **Danh sách các thuộc tính của các đối tượng (Object) cần thiết:**

Object	Properties	Events
frmBaiTap1	Name: frmBaiTap1 Text: My name Project FontName: Tahoma FontSize: 11 AcceptButton: btnShow (nhận sự kiện Click chuột khi nhấn Enter) CancelButton: btnExit (nhận sự kiện Click chuột khi nhấn Esc)	FormClosing
txtYourName	Name: txtYourName BorderStyle: FixSingle	Leave (mất tiêu điểm)
txtYear	Name: txtYear BorderStyle: FixSingle	TextChanged
btnShow	Name: btnShow Text: &Show	Click
btnClear	Name: btnClear Text: &Clear	Click
btnExit	Name: btnExit Text: E&xit	Click
errorProvider	Name: errorProvider1	

- **Thay đổi thứ tự nhận tiêu điểm trên Form: chọn menu View → Tab Order**



Lần lượt thực hiện click chọn từng phần tử trên Form theo thứ tự nhận tiêu điểm:



- **Viết các sự kiện tương ứng với các yêu cầu:**

Sự kiện đóng xác nhận khi form được đóng:

```
private void frmBT1_FormClosing(object sender, FormClosingEventArgs e)
{
    DialogResult r;
    r = MessageBox.Show("Bạn có muốn thoát?", "Thoát",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question,
        MessageBoxDefaultButton.Button1);
    if (r == DialogResult.No)
        e.Cancel = true;
}
```

Kiểm tra Textbox YourName đã được nhập nội dung chưa:

```
private void txtYourName_Leave(object sender, EventArgs e)
{
    Control ctr = (Control)sender;
    if (ctr.Text.Trim().Length == 0)
        this.errorProvider1.SetError(ctr, "You must enter Your Name");
}
```

```

        else
            this.errorProvider1.Clear();
    }

```

Yêu cầu Textbox Year phải nhập số, nhập sai dùng ErrorProvider thông báo lỗi:

```

private void txtYear_TextChanged(object sender, EventArgs e)
{
    Control ctr = (Control)sender;
    if (ctr.Text.Length>0 && !Char.IsDigit(ctr.Text[ctr.Text.Length-1])
        this.errorProvider1.SetError(ctr, ",This is not avalid number");
    else
        this.errorProvider1.Clear();
}

```

Hiển thị thông tin theo yêu cầu khi click vào Button Show:

```

private void btnShow_Click(object sender, EventArgs e)
{
    int age;
    string s;
    s = "My name is: " + txtYourName.Text + "\n";
    age = DateTime.Now.Year - Convert.ToInt32(txtYear.Text);
    s = s + "Age: " + age.ToString();
    MessageBox.Show(s);
}

```

Button Clear thực hiện lệnh xóa các TextBox chuẩn bị cho lần nhập tiếp theo:

```

private void btnShow_Click(object sender, EventArgs e)
{
    //clear Text
    txtYourName.Clear();
    txtYear.Clear();
    //Textbox YourName reset focus
    txtYourName.Focus();
}

```

Thoát chương trình khi click vào Button Exit:

```

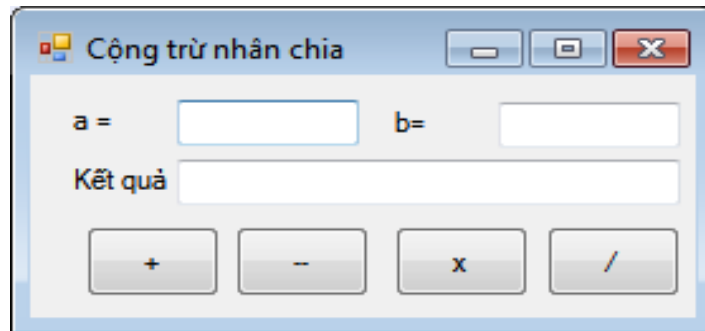
private void btnExit_Click(object sender, EventArgs e)
{
    this.Close();
}

```



## 2. Bài tập tại lớp

**Bài 1.** Thiết kế Form thực hiện phép tính  $+$ ,  $-$ ,  $\times$ ,  $/$  của hai số  $a$ ,  $b$  với giao diện như sau:

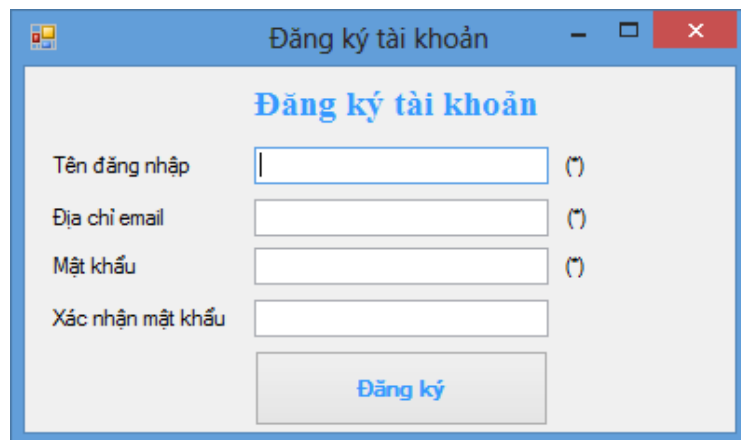


**Yêu cầu:** Khi chọn các phép toán  $+$ ,  $-$ ,  $\times$ ,  $/$  tương ứng trên các button thì hiển thị kết quả lên textbox kết quả.

- Kiểm tra tính hợp lệ của các textbox theo hai mức độ:
  - Mức 1: Nếu dữ liệu nhập vào textbox  $a$ ,  $b$  không hợp lệ thì dùng `ErrorProvider` thông báo lỗi.
  - Mức 2: Chặn không cho người dùng nhập giá trị khác số vào textbox  $a$ ,  $b$ .
- Thông báo lỗi bằng `MessageBox` nếu dữ liệu nhập không phù hợp
- Hỏi xác nhận trước khi đóng

Form.

**Bài 2.** Thiết kế form đăng ký tài khoản



**Yêu cầu:**

- Kiểm tra định dạng của email sau khi nhập và ra khỏi textbox Địa chỉ email.
- Bắt buộc nhập dữ liệu trên những textbox có (\*).
- Khi chọn button Đăng ký hoặc nhấn phím Enter sau khi nhập xong dữ liệu vào textbox Xác nhận mật khẩu thì hiển thị thông tin của tất cả các textbox lên `MessageBox`.



- Hỏi xác nhận trước khi đóng Form.

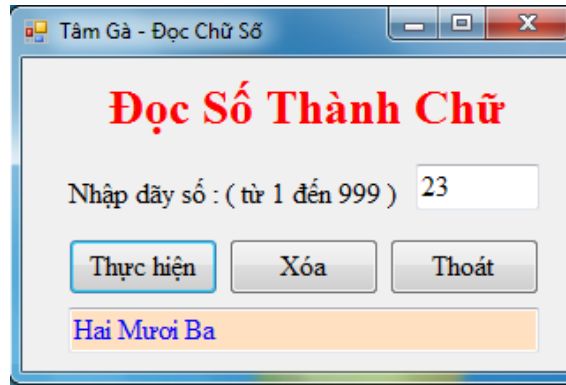
**Bài 3.** Thiết kế giao diện thực hiện yêu cầu nhập vào 2 số a, b. Xuất ra UCLN và BCNN của 2 số a và b (kiểm tra dữ liệu nhập và xuất các thông báo lỗi nếu có).

**Bài 4.** Thiết kế giao diện sau:

**Yêu cầu:**

- Nhập vào một dãy số nguyên bất kỳ (nhập số nguyên vào textbox Nhập số, nhấn button Nhập), xuất các số vừa nhập ra textbox Dãy vừa nhập.
- Bổ sung thêm button Tính tổng, Tổng chẵn, Tổng lẻ các số vừa nhập và xuất ra các textbox tương ứng.
- Tính tổng các số chẵn và tổng các số lẻ.
- Button Tiếp tục trả lại trạng thái ban đầu của form.
- Button Thoát: xác nhận và đóng form.

**Bài 5.** Viết chương trình đọc số thành chữ với giao diện sau:



**Yêu cầu:**

- Nhập vào một số nguyên dương bất kỳ từ 1 đến 999.
- Đọc số vừa nhập thành chữ số khi chọn button Thực hiện.
- Button Xóa: trả form lại trạng thái ban đầu
- Button Thoát: xác nhận và đóng form.

### 3. Bài tập nâng cao

**Bài 1.** Thiết kế giao diện sau:



**Yêu cầu:** Tạo một project giúp cho rạp chiếu phim quản lý việc bán vé của mình. Rạp có 3 hàng ghế, mỗi hàng có 5 ghế, các ghế được đánh số từ 1 đến 15 và được phân thành 3 lô như (hình trên). Giá vé theo lô như sau:

Giá vé lô A 1000/vé

Giá vé lô B 1500/vé

Giá vé lô C 2000/vé

Form giao diện trình bày là sơ đồ các chỗ ngồi tương ứng trong rạp phim để người sử dụng chọn vị trí muốn mua. Trên sơ đồ này cũng thể hiện vị trí vé đã bán và những vị

trí vé chưa bán bằng các thể hiện màu khác nhau (vé chưa bán màu trắng, vé đã bán màu vàng, vé đang chọn màu xanh). Khi người sử dụng click chuột tại một vị trí trên sơ đồ thì:

- Nếu đây là vị trí chưa bán thì đổi màu của vị trí này sang màu xanh để cho biết đây là vị trí đang chọn.
- Nếu đây là vị trí đang chọn (có màu xanh) thì đổi màu của vị trí này trở về màu trắng.
- Nếu đây là một vị trí đã bán vé thì xuất hiện một MessageBox thông báo cho người sử dụng biết.

Sau khi đã chọn các vị trí, người sử dụng có thể click chuột vào button **CHỌN** để thanh toán hoặc **HỦY BỎ**:

Nếu click vào button **CHỌN** thì:

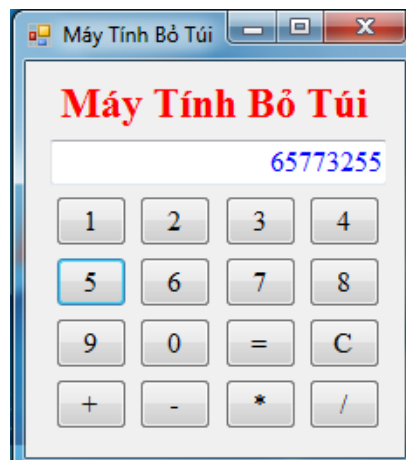
- Đổi màu các vị trí đang chọn (màu xanh) trên sơ đồ sang màu vàng (đây là vị trí đã bán).
- Xuất tổng số tiền phải trả cho số vé đã mua lên label Thành

Tiền. Nếu click vào button **HỦY BỎ** thì:

- Đổi màu các vị trí đang chọn (màu xanh) trên sơ đồ sang màu trắng.
- Xuất lên label Thành Tiền giá trị 0.

#### 4. Bài tập về nhà

**Bài 1:** Viết chương trình dành cho máy tính bỏ túi đơn giản thực hiện các phép tính cộng, trừ, nhân, chia và xóa đơn giản.



-----Hết-----