

BÀI TẬP LẬP TRÌNH GIAO DIỆN

2. Bài tập chương 2

A. Bài tập có hướng dẫn

1. Xây dựng ứng dụng các phép tính đơn giản với giao diện như sau:
 - Nhập hai số, click vào một phép toán, cho kết quả như sau:

Các phép tính

CÁC PHÉP TÍNH ĐƠN GIẢN

Số thứ 1: 10 Số thứ 2: 3

+ - x /

Kết quả: 3.33 Đóng

- Nếu số thứ 2 là 0, click nút chia, cho kết quả như sau:

Các phép tính

CÁC PHÉP TÍNH ĐƠN GIẢN

Số thứ 1: 10 Số thứ 2: 0

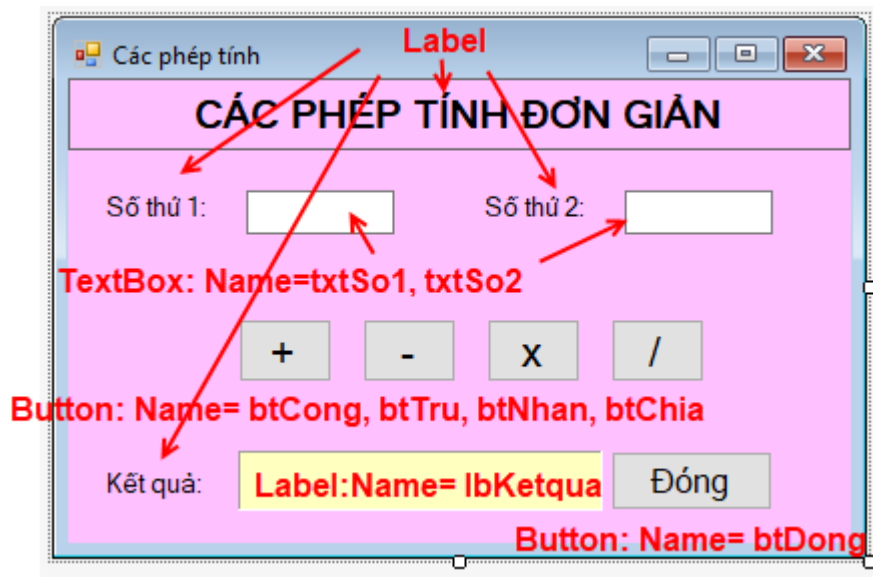
+ - x /

Kết quả: Số chia phải khác 0 Đóng

Hướng dẫn:

- Tạo project, chọn ứng dụng Window Form Application với project name là “Cacpheptinh”

- Thiết kế giao diện như sau:



- Double click lên button btCong (+), chuyển sang phần code behind, viết code xử lý trong hàm btCong_Click như sau:

```
private void btCong_Click(object sender, EventArgs e)
{
    //Lấy giá trị trong hai textbox chuyển về số
    int a = int.Parse(txtSo1.Text);
    int b = int.Parse(txtSo2.Text);
    //Hiển thị kết quả
    lbKetqua.Text = String.Format("{0}", a + b);
}
```

- Thao tác tương tự cho hai button trừ và nhân.
- Riêng với code tính phép chia, kết quả phải là số thực:

```
private void btChia_Click(object sender, EventArgs e)
{
    int a = int.Parse(txtSo1.Text);
    int b = int.Parse(txtSo2.Text);
    if (b != 0) //nếu số chia khác 0 mới thực hiện
        lbKetqua.Text = String.Format("{0:0.00}", (double)a / b);
    else
        lbKetqua.Text = "Số chia phải khác 0";
}
```

- Double click lên button btDong (Đóng), viết code như sau:

```
private void btDong_Click(object sender, EventArgs e)
{
}
```

```
Close();  
}
```

- Biên dịch và chạy chương trình, nhập các giá trị và click lần lượt lên các button xem kết quả.
2. Tương tự bài thực hành số 1, nhưng thay đổi giao diện như hình bên dưới và có xử lý lỗi nhập sai giá trị:

The image displays two screenshots of a Windows application titled "Form2". The application has a pink background and a title bar with standard Windows window controls.

Top Screenshot:

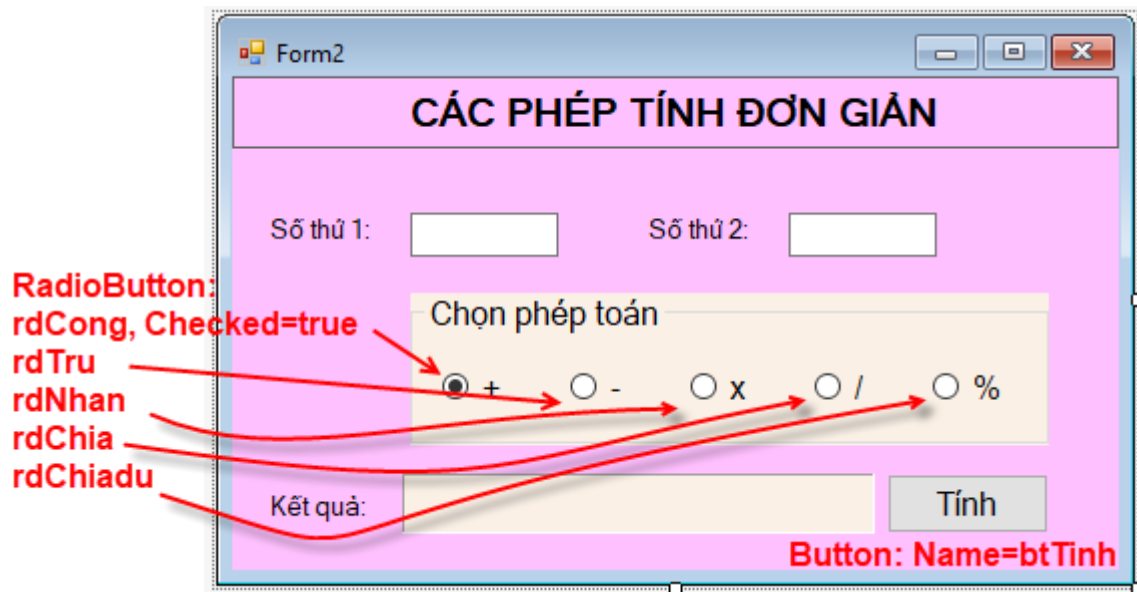
- Header:** "CÁC PHÉP TÍNH ĐƠN GIẢN" (Simple Operations).
- Inputs:** "Số thứ 1:" (Number 1) with a text box containing "5"; "Số thứ 2:" (Number 2) with a text box containing "a".
- Operation Selection:** A section titled "Chọn phép toán" (Choose operation) with five radio buttons: "+" (selected), "-", "x", "/", and "%".
- Result:** "Kết quả:" (Result) with a text box containing "Bạn phải nhập hai số" (You must enter two numbers) and a "Tính" (Calculate) button.

Bottom Screenshot:

- Header:** "CÁC PHÉP TÍNH ĐƠN GIẢN" (Simple Operations).
- Inputs:** "Số thứ 1:" (Number 1) with a text box containing "999999"; "Số thứ 2:" (Number 2) with a text box containing "99999".
- Operation Selection:** A section titled "Chọn phép toán" (Choose operation) with five radio buttons: "+" (unselected), "-" (unselected), "x" (selected), "/" (unselected), and "%" (unselected).
- Result:** "Kết quả:" (Result) with a text box containing "Không thể xử lý số quá lớn" (Cannot process large number) and a "Tính" (Calculate) button.

Hướng dẫn:

- Tạo project với project name là Calculator.
- Thiết kế giao diện như sau:



- Sử dụng try..catch..finally, throw, checked... để xử lý các lỗi:
 - Để trống ô, nhập sai giá trị.
 - Chia cho 0 khi thực hiện phép chia và chia lấy phần dư (%).
 - Tràn có khi thực hiện phép nhân.
- Double click lên Button Tính và viết code xử lý như sau:

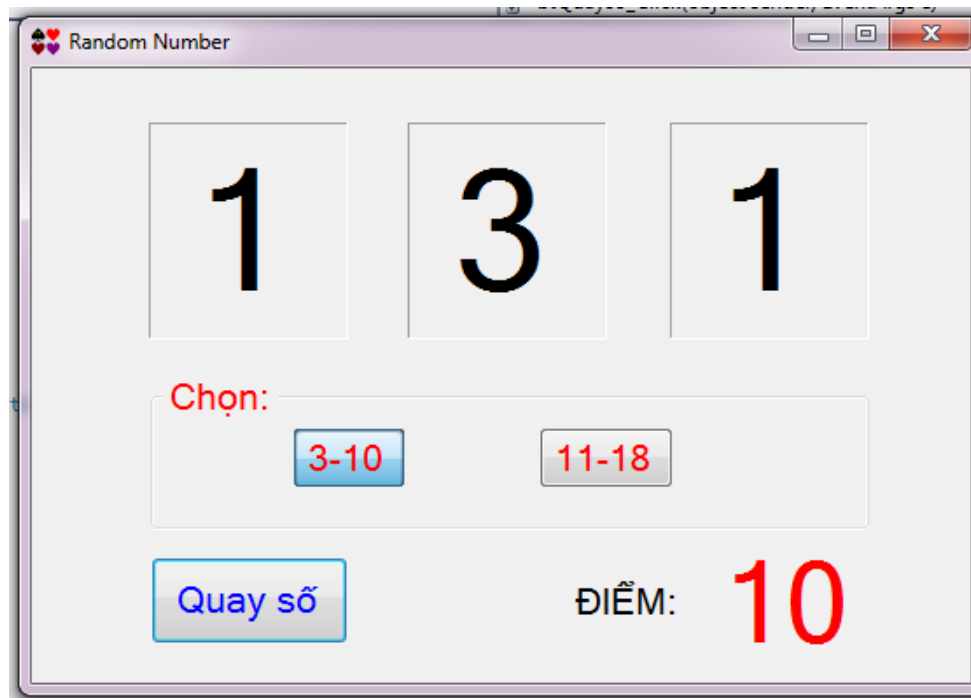
```
private void btTinh_Click(object sender, EventArgs e)
{
    try
    {
        int a = checked(int.Parse(txtSol1.Text));
        int b = checked(int.Parse(txtSo2.Text));
        if (rdCong.Checked)
            lbKetqua.Text = String.Format("{0}", checked(a + b));
        else if (rdtru.Checked)
            lbKetqua.Text = String.Format("{0}", a - b);
        else if (rdNhan.Checked)
            lbKetqua.Text = String.Format("{0}", checked(a * b));
        else if (rdChia.Checked)
        {
            if (b == 0)
                throw new DivideByZeroException("Số chia phải khác 0");
            lbKetqua.Text = String.Format("{0:0.00}", (double)a / b);
        }
        else
            lbKetqua.Text = String.Format("{0}", a % b);
    }
    catch (FormatException) {
        lbKetqua.Text = "Bạn phải nhập hai số";
    }
}
```

```

    }
    catch (DivideByZeroException ex) {
        lbKetqua.Text = ex.Message ;
    }
    catch(OverflowException) {
        lbKetqua.Text = "Không thể xử lý số quá lớn";
    }
}

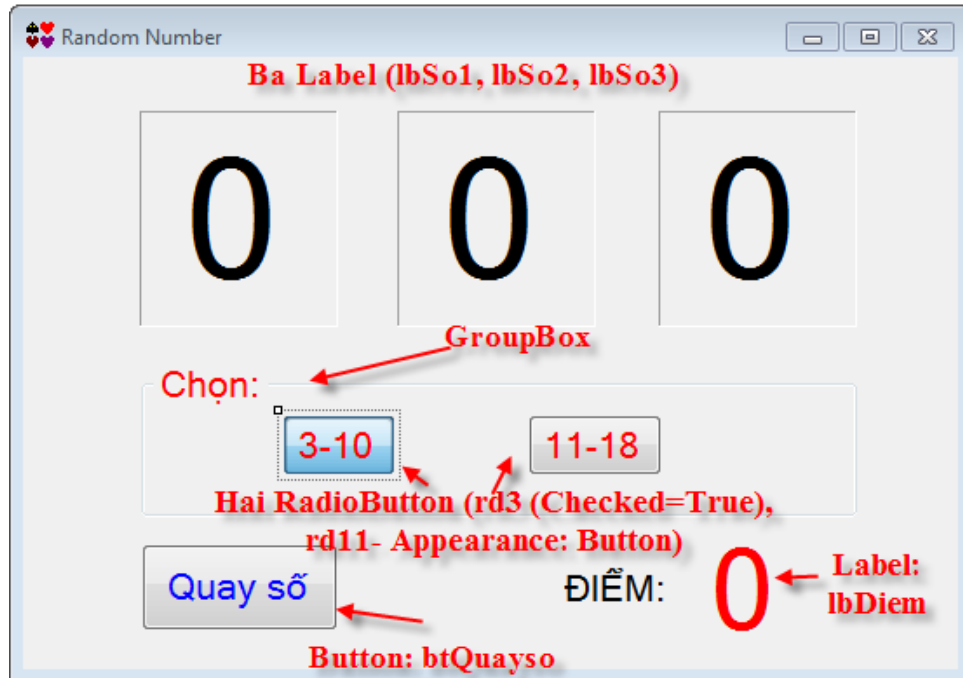
```

- Biên dịch, chạy chương trình, lần lượt nhập các giá trị đúng, sai để kiểm tra kết quả
3. Viết ứng dụng RandomNumber (chương trình game quay số: người chơi chọn một đáp án (3-10 hoặc 11-18), sau đó nhấn nút quay số, các số sẽ sinh ngẫu nhiên, nếu tổng ba số đúng trong khoản người chơi đã chọn, điểm được cộng thêm 10, ngược lại điểm bị trừ đi 10 (điểm ban đầu bằng 0).



Hướng dẫn:

- Tạo project với project name là RandomNumber, thiết kế giao diện như sau:



- Khai báo các biến:

```
Random Rand = new Random();
int Diem = 0;
```

- Trong đó, đối tượng Random được sử dụng để sinh số ngẫu nhiên thông qua phương thức Next. Phương thức này có cú pháp như sau:
 - Next(): trả về một số nguyên ngẫu nhiên từ 0.
 - Next(int maxValue): trả về một số nguyên ngẫu nhiên từ 0 đến maxValue - 1.
 - Next(int minValue, maxValue): trả về một số nguyên ngẫu nhiên từ minValue đến maxValue - 1.
- Double click vào nút Quay số, viết code xử lý sự kiện click vào nút này:

```
private void btQuayso_Click(object sender, EventArgs e)
{
    //Lấy ngẫu nhiên 3 số nguyên từ 1- 6
    int so1 = Rand.Next(1, 7);
    int so2 = Rand.Next(1, 7);
    int so3 = Rand.Next(1, 7);
    //hiển thị trên giao diện
    lbSo1.Text = so1.ToString();
    lbSo2.Text = so2.ToString();
    lbSo3.Text = so3.ToString();
    //kiểm tra, hiển thị kết quả
    if(rd3.Checked) //chọn 3-10
```

```

{
    if ((so1 + so2 + so3) <= 10)
        Diem += 10;
    else
        Diem -= 10;
}
else // chọn 11-18
{
    if ((so1 + so2 + so3) <= 10)
        Diem -= 10;
    else
        Diem += 10;
}
lbDiem.Text = Diem.ToString();
}

```

4. Viết chương trình Behoctoan (bé học toán) thực hiện các phép toán cộng, trừ, nhân, chia trên số nguyên, có giao diện và chức năng như sau:

- Chương trình hiển thị một phép toán, người chơi sẽ nhập kết quả, nếu chọn đúng:



- Nếu chọn sai:

**Yêu cầu:**

- Giao diện đầu tiên khi chương trình chạy lên hiển thị phép toán $0 + 0$.
- Click vào nút Tiếp tục, sinh ngẫu nhiên một biểu thức trong các phép toán cộng, trừ, nhân, chia, có kiểm tra không cho sinh ngẫu nhiên phép toán chia cho 0.
- Người chơi nhập kết quả, click nút Xem, chương trình sẽ hiển thị kết quả đúng hoặc sai.
- Có thể click vào các button để nhập số vào ô Trả lời thay vì nhập bằng bàn phím
- Chương trình chỉ thực hiện các phép toán trên số nguyên nhỏ hơn 10.
- Xử lý các ngoại lệ như: không nhập hoặc nhập sai giá trị.

Hướng dẫn:

- Thiết kế giao diện.



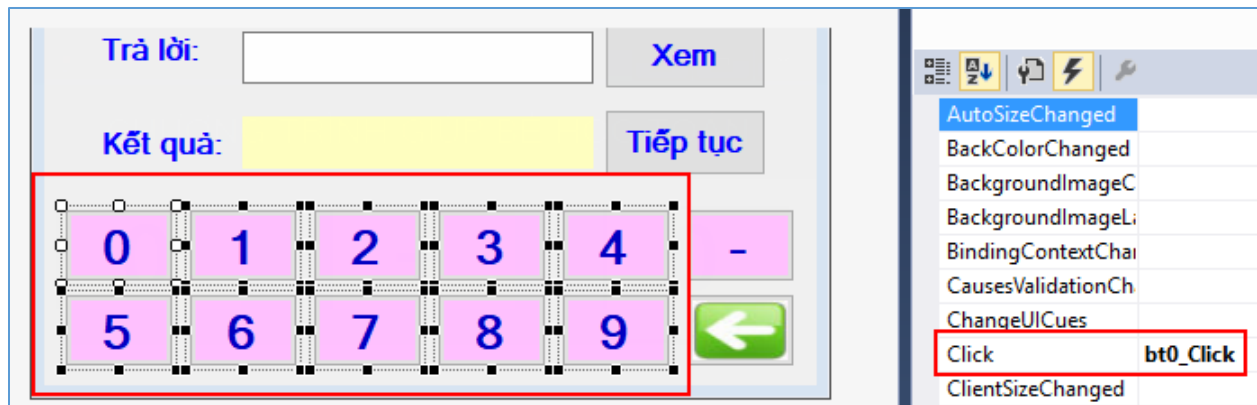
- Khai báo các biến cần thiết:

```
Random rand = new Random();
string[] toantu = { "+", "-", "x", ":" };
int vitri = 0;
```

- Viết code xử lý sự kiện click trên button Tiếp tục

```
do
{
    int sol = rand.Next(10);
    int so2 = rand.Next(10);
    vitri = rand.Next(4);
}while (so2 == 0 && vitri == 3);
lbSo1.Text = sol.ToString();
lbSo2.Text = so2.ToString();
lbToantu.Text = toantu[vitri];
lbKetqua.Text = "";
txtDoan.Text = "";
```

- Viết code xử lý sự kiện click trên các button số: viết một phương thức chung cho tất cả các button từ **bt0** - **bt9** (không viết code cho từng nút).



```
private void bt0_Click(object sender, EventArgs e)
{
    Button bt = (Button)sender;
    txtDoan.Text += bt.Text;
}
```

- Viết code xử lý sự kiện click trên button **Xem**.

```
private void btXem_Click(object sender, EventArgs e)
{
    int ketqua = 0;
    switch (vitri)
    {
        case 0:
            ketqua = int.Parse(lbSo1.Text) + int.Parse(lbSo2.Text);
            break;
        case 1:
            ketqua = int.Parse(lbSo1.Text) - int.Parse(lbSo2.Text);
            break;
        case 2:
            ketqua = int.Parse(lbSo1.Text) * int.Parse(lbSo2.Text);
            break;
        case 3:
            ketqua = int.Parse(lbSo1.Text) / int.Parse(lbSo2.Text);
            break;
    }

    try
    {
        int doan = int.Parse(txtDoan.Text);
        if (doan == ketqua)
            lbKetqua.Text = "Đúng rồi!";
        else
            lbKetqua.Text =
                "Sai rồi! Kết quả là " + ketqua.ToString();
    }
    catch (FormatException){}
```

```

        MessageBox.Show("Bạn phải nhập số!");
    }
}

```

- Viết code xử lý sự kiện click trên button dấu –

```

private void btDau_Click(object sender, EventArgs e)
{
    if (txtDoan.Text != "") return;
    txtDoan.Text = "-";
}

```

- Viết code xử lý sự kiện click trên button btXoa (←)

```

private void btXoa_Click(object sender, EventArgs e)
{
    if (txtDoan.Text == "") return;
    txtDoan.Text =
        txtDoan.Text.Substring(0, txtDoan.Text.Length - 1);
}

```

- Build và chạy ứng dụng.

5. Viết chương trình StringProcess để xử lý chuỗi với giao diện và chức năng như sau:

Yêu cầu:

- Chuỗi “Chương trình minh họa xử lý chuỗi” chuyển động (cuộn) từ trái sang phải.
- Nút Chèn: chèn chuỗi s2 vào chuỗi s1 tại vị trí được nhập trong ô vị trí.
- Nút Thay thế s2 bằng s3: thay thế chuỗi con s2 trong s1 (nếu có) bằng chuỗi s3 .
- Đảo từ trong s1: đảo các từ trong chuỗi s1.
- Xóa s2 trong s1: xóa chuỗi con s2 trong s1 (nếu có).
- Chuẩn hóa chuỗi: chuẩn hóa chuỗi s1 theo nguyên tắc:

- Xóa các khoảng trắng thừa, đầu, giữa và cuối chuỗi, chỉ giữ lại giữa các từ một khoảng trắng.
- Chuyển thành chữ hoa các ký tự đầu mỗi từ.
- Khôi phục: trả về chuỗi ban đầu

Hướng dẫn:

- Thay thế chuỗi s2 bằng s3: sử dụng hàm Replace.

```
txtS1.Text = txtS1.Text.Replace(txtS2.Text, txtS3.Text);
```

- Chèn chuỗi s2 vào s1 tại vị trí chỉ định: sử dụng hàm Insert

```
txtS1.Text = txtS1.Text.Insert(int.Parse(txtVitri.Text),  
txtS2.Text);
```

- Xóa s2 trong s1: Tìm vị trí s2 trong s1 bằng hàm IndexOf, nếu có, xóa chuỗi bằng hàm Remove.
 - Đảo từ: Tách chuỗi (dựa vào các khoảng trắng như ' ', '\n', '\t') bằng hàm Split vào một mảng string → Duyệt qua mảng từ phần tử cuối cùng, gán lại cho textbox.
 - Chuẩn hóa chuỗi: tách chuỗi, duyệt qua từng phần tử trong mảng string, chuyển sang chữ hoa ký tự đầu chuỗi, sau đó gán dồn cho textbox.
6. Viết chương trình sắp xếp mảng với giao diện và các chức năng như sau:

Yêu cầu:

- Nhập số phần tử, click nút tạo mảng ngẫu nhiên, các phần tử của mảng sẽ được tạo ngẫu nhiên và hiển thị trong ô “Mảng ban đầu”
- Click nút “Sắp xếp”, các phần tử mảng được sắp xếp và liệt kê trong các ô bên dưới như hình trên.

Hướng dẫn:

- Thiết kế giao diện:

- Khai báo các biến mảng cần thiết.
- Khai báo và viết code hàm ShowArray: hiển thị các phần tử mảng lên textbox (vì hàm này sẽ được sử dụng nhiều lần)
- Viết code xử lý nút tạo mảng lấy ngẫu nhiên các số từ 0-99.
- Khai báo và viết code hàm Sort (sắp xếp mảng tăng, giảm, đảo): sử dụng các hàm thư viện.
- Khai báo và viết code hàm SortSpecial (sắp xếp mảng chẵn tăng, lẻ giảm).
- Gọi các hàm trên trong sự kiện click button sắp xếp.

```
private void btSapxep_Click(object sender, EventArgs e)
{
    Sort();
    SortSpecial();
}
```

B. Bài tập tự làm:

1. Viết chương trình chuyển đổi giữa Mã ASCII và ký tự với giao diện như sau:

Chuyển mã ASCII

CHUYỂN MÃ ASCII

Chuyển mã ASCII sang ký tự

Mã ASCII:

Kết quả:

Chuyển ký tự sang mã ASCII

Ký tự:

Kết quả:

Yêu cầu: xử lý các ngoại lệ không nhập hoặc nhập sai giá trị.

- Viết chương trình giải phương trình bậc 2 với giao diện như sau:

PHƯƠNG TRÌNH BẬC 2

Chương trình giải phương trình bậc 2

a= b= c=

Kết quả:

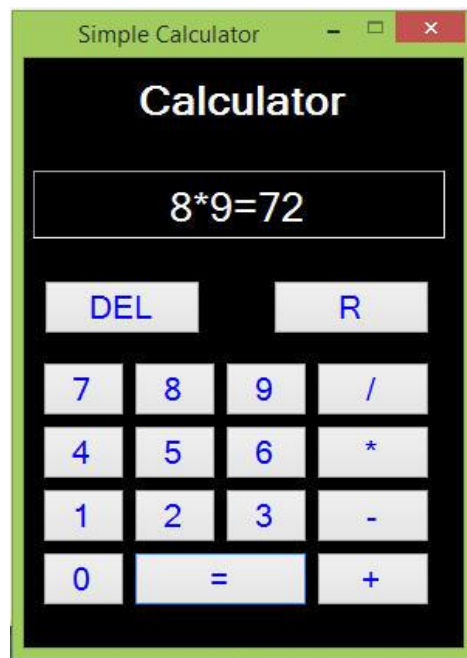
Yêu cầu:

- Sử dụng hàm.
 - Xử lý các ngoại lệ không nhập hoặc nhập sai giá trị.
- Viết chương trình quay số với giao diện như sau:



Yêu cầu:

- Giao diện đầu tiên khi chương trình chạy lên hiển thị ba số 0, điểm ban đầu là 0.
 - Click vào nút Play, sinh ngẫu nhiên ba số.
 - Nếu ba số có giá trị bằng nhau, điểm được cộng thêm 100, ngược lại, điểm bị trừ đi 10.
4. Viết chương trình máy tính đơn giản (SimpleCalculator) với giao diện và chức năng như sau:



Yêu cầu:

- Nút DEL: xóa biểu thức

- Nút R: xóa lùi biểu thức
 - Xử lý ngoại lệ trong trường hợp biểu thức nhập không hợp lệ (biểu thức hợp lệ là biểu thức gồm hai toán hạng và một toán tử như hình)
5. Viết chương trình sinh ngẫu nhiên một mảng các số nguyên với số phần tử được nhập trên giao diện. Sau đó click vào nút Thống kê, hiển thị các kết quả sau:
- Tổng các phần tử chẵn.
 - Tổng các phần tử lẻ.
 - Trung bình cộng các phần tử trong mảng.
 - Số các số nguyên tố trong mảng.
6. Viết chương trình quản lý một danh sách sinh viên, mỗi sinh viên gồm các thông tin: MSSV, Họ, Tên. Thiết kế giao diện và các chức năng xử lý cho phép:
- Nhập thêm một sinh viên vào danh sách.
 - Tìm kiếm sinh viên theo MSSV, theo Tên
 - Sắp xếp thứ tự sinh viên theo Tên.