

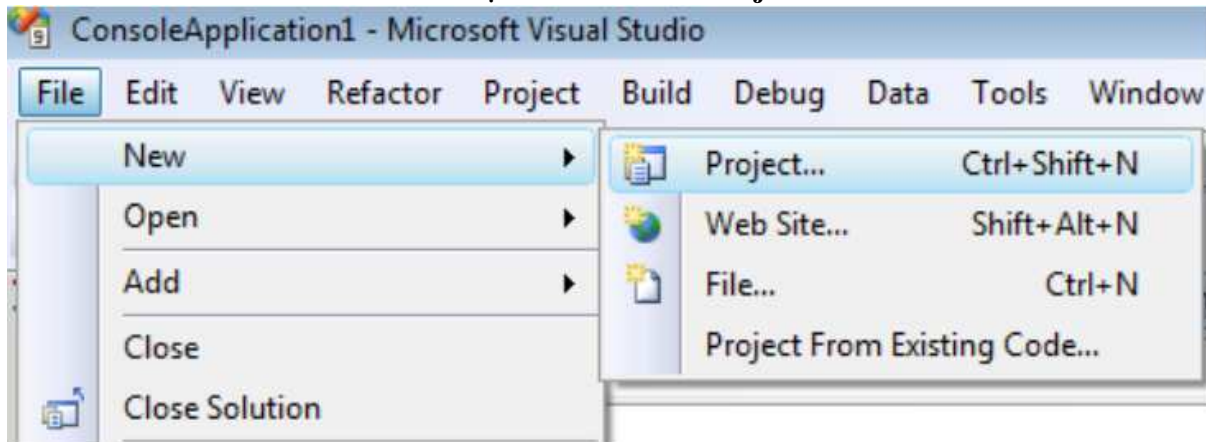
BÀI TẬP THỰC HÀNH LẬP TRÌNH WINDOWS – C#

Phần 1: Làm quen với ứng dụng Form

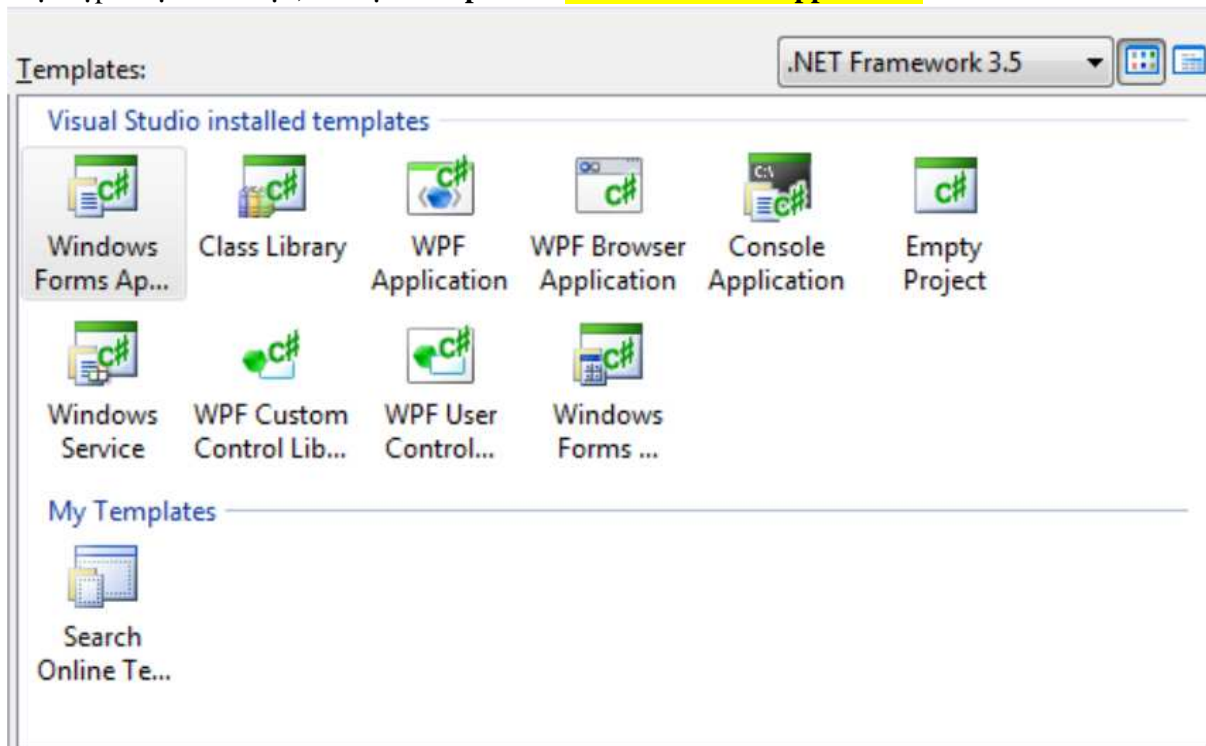
1. Bài tập mở đầu

1.1. Khởi tạo Project

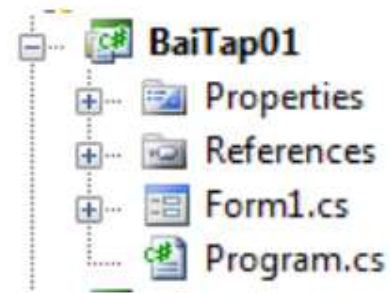
Từ Visual Studio 2005/2008/2010 ta chọn **File** → **New** → **Project**.



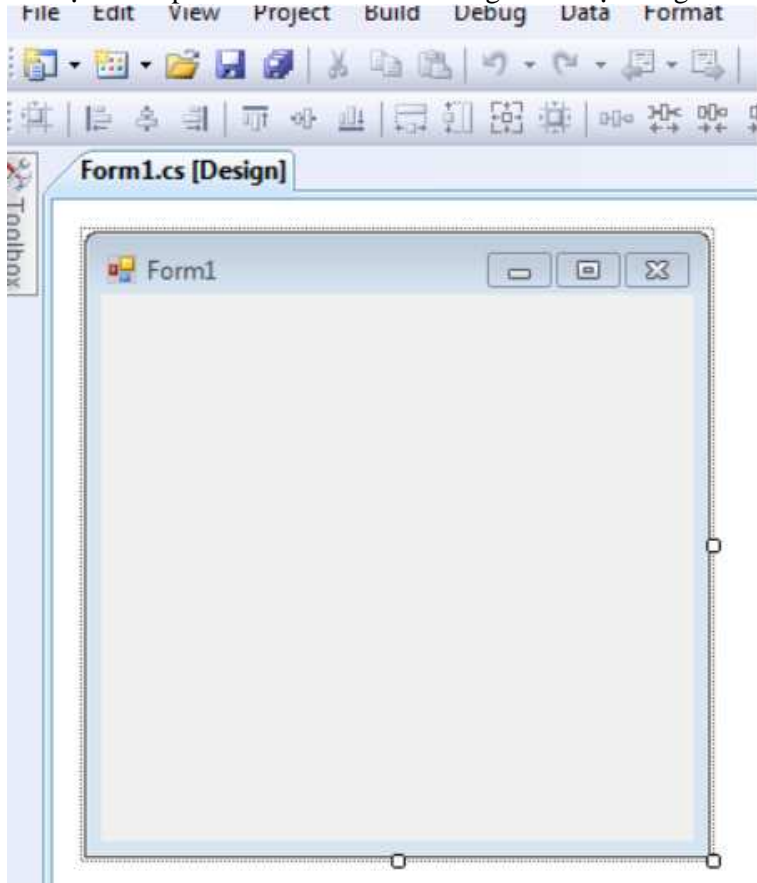
Một hộp thoại xuất hiện, ta chọn **Template** là **Windows Form Application**.



Ứng dụng được tạo ra như sau:

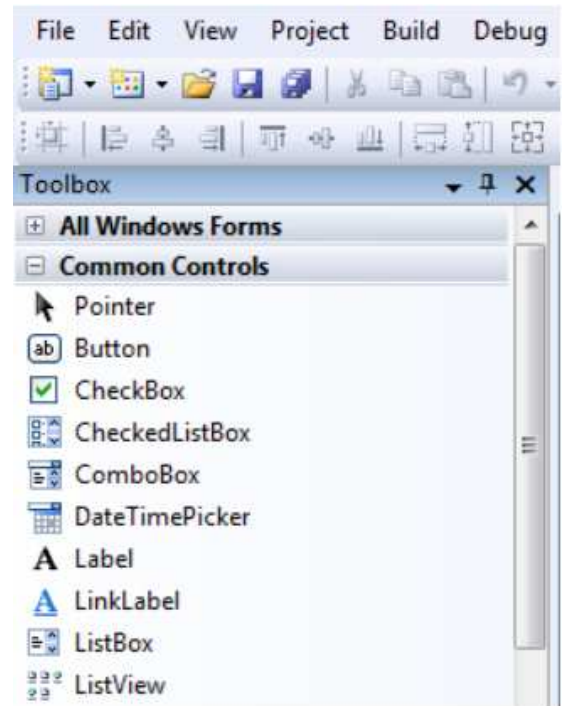


Project BaiTap01 gồm có thành phần Program.cs giống trong ứng dụng Console Application và còn có một thành phần mới là Form1.cs đang ở chế độ Design với giao diện như sau:



Chú ý cửa sổ Toolbox chứa các công cụ để thiết kế:

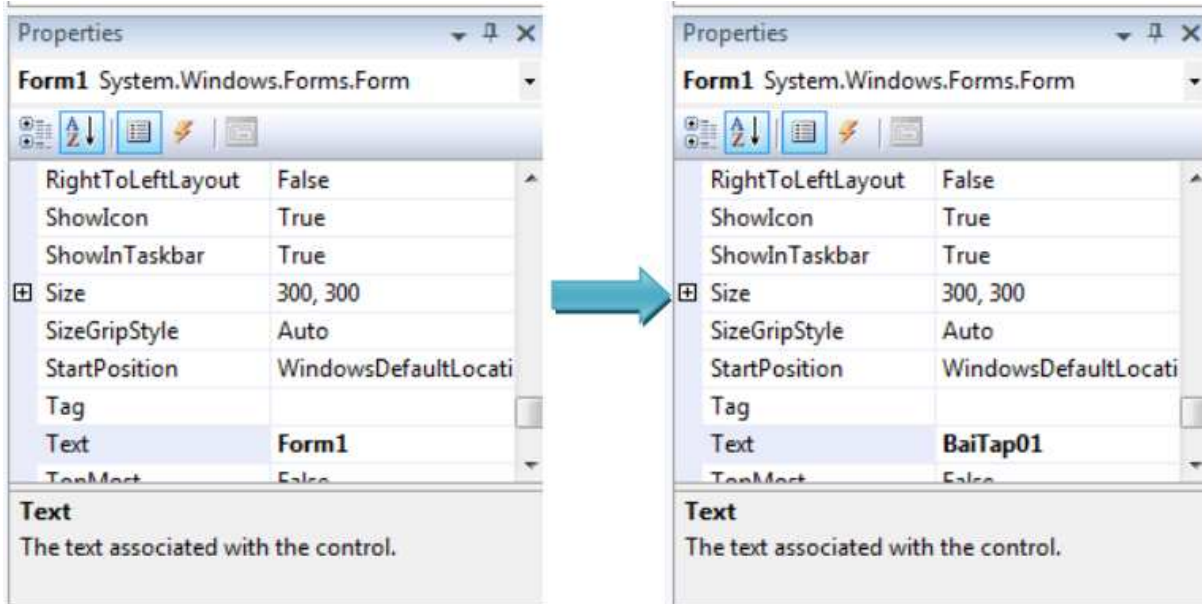
Nếu không thấy cửa sổ này, ta chọn menu **View** → **Toolbox**.



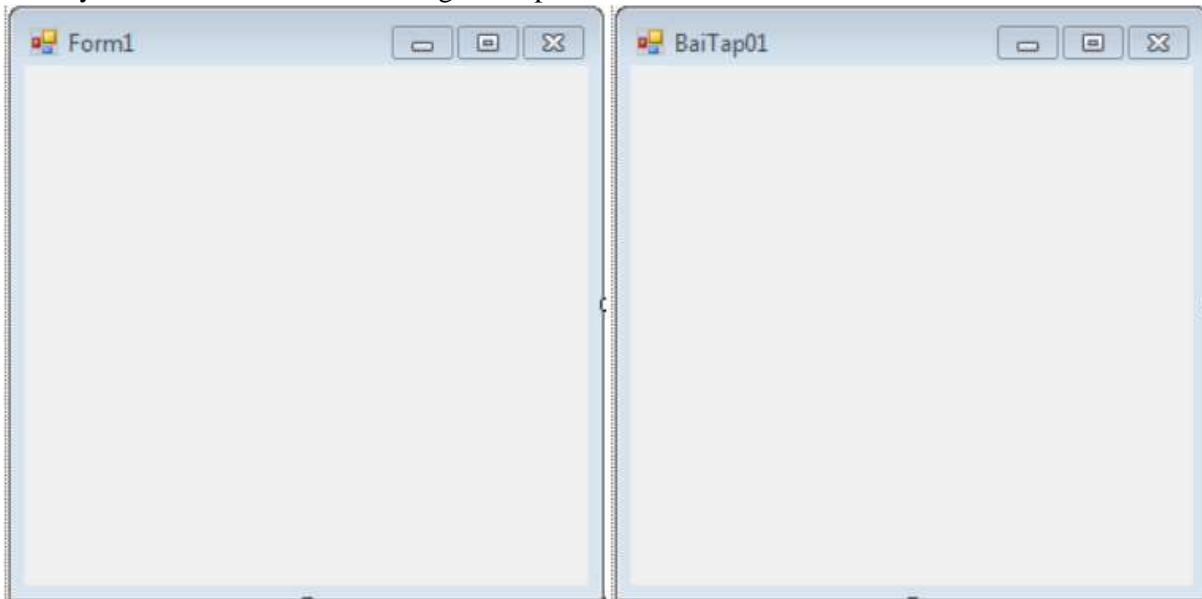
1.2. Ứng dụng mở đầu:

1.2.1. Đổi tên form:

Click lên Form1 ở cửa sổ Design, quan sát trên cửa sổ Properties, ta thấy có thuộc tính Text, giá trị mặc định là Form1, ta đổi thành BaiTap01.

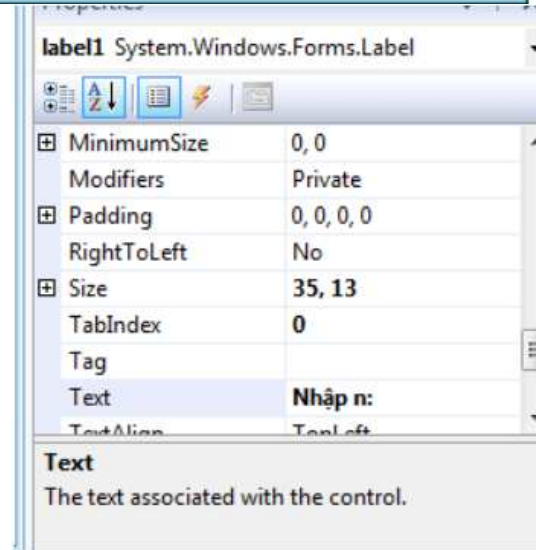
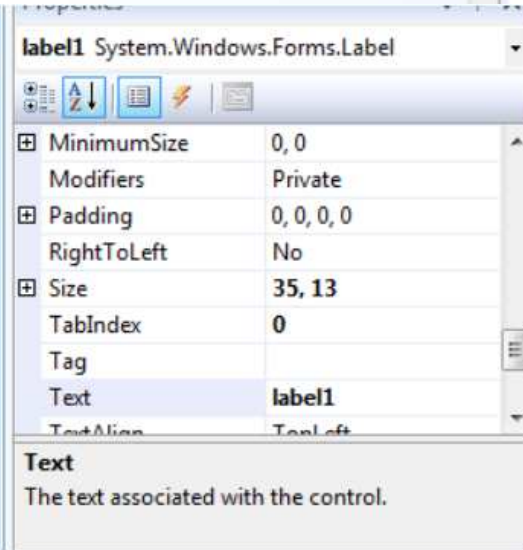
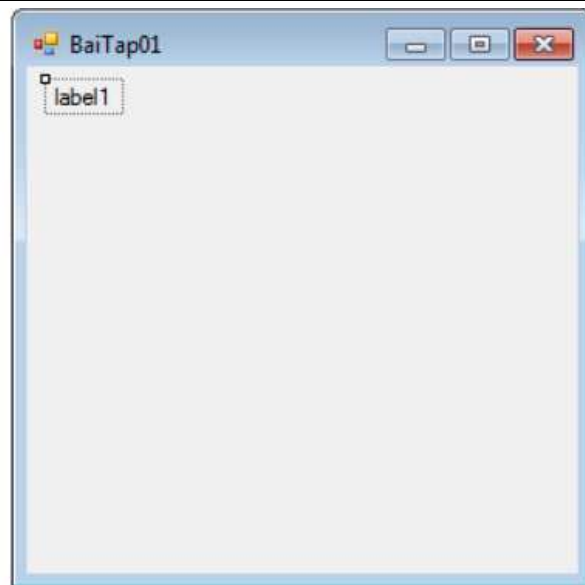
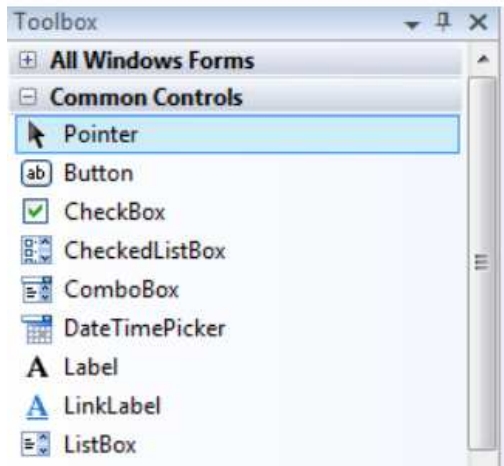


Ta thấy thanh tiêu đề Form1 đổi sang BaiTap01.

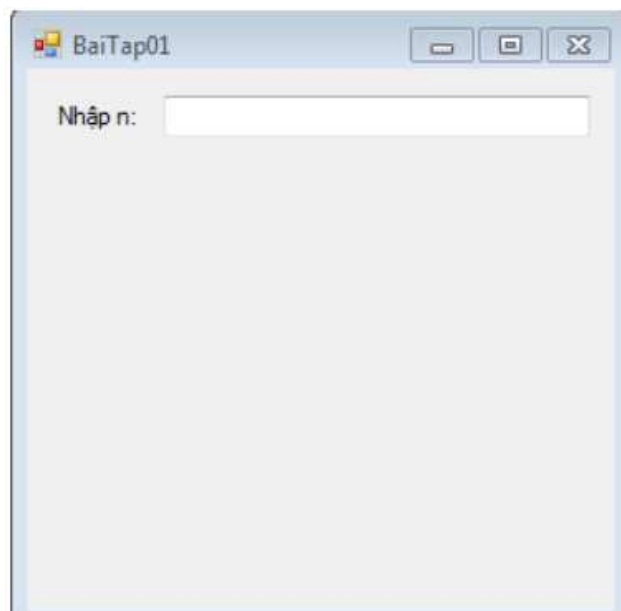
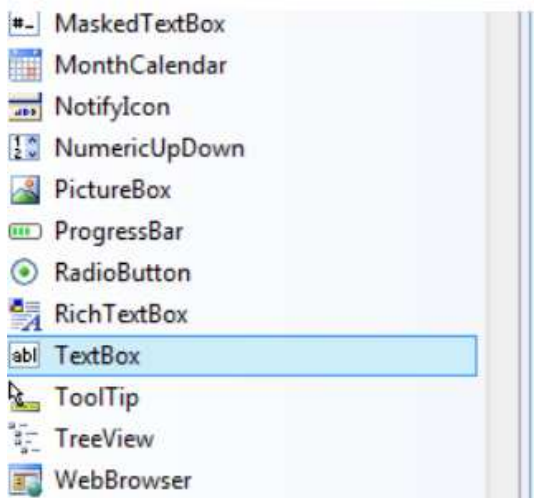


1.2.2. Kéo thả các control từ Toolbox vào Form

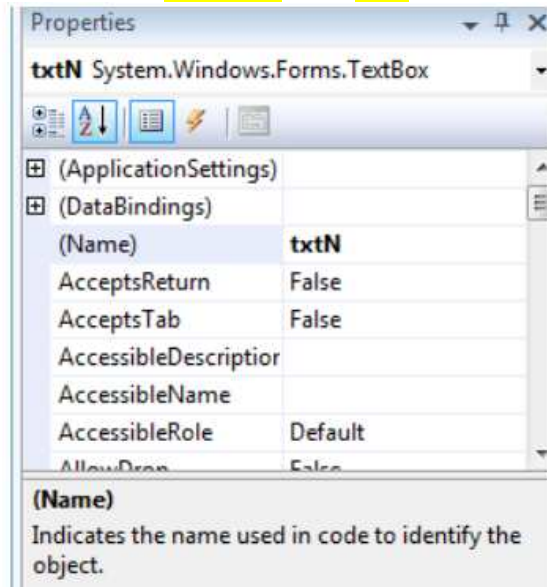
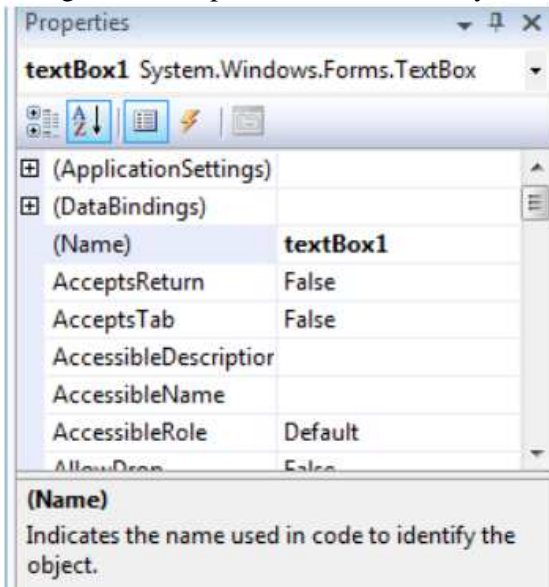
Từ thanh Toolbox, kéo thả một Label bỏ vào Form, sau đó di chuyển chuột đến cửa sổ Properties, chọn thuộc tính Text, đổi từ **label1** sang **Nhập n.**



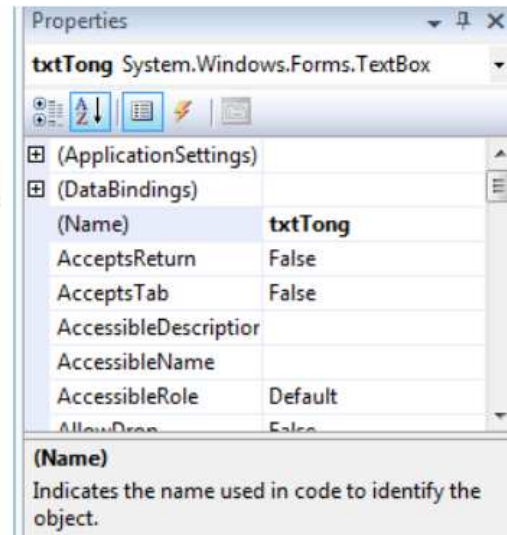
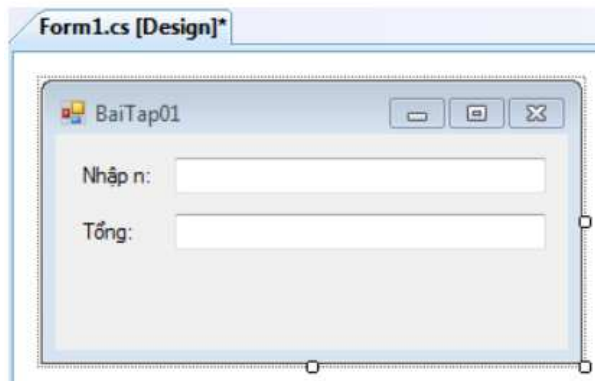
Tiếp theo kéo thả một Textbox bỏ vào form.



Trong cửa sổ Properties của textbox này, thuộc tính Name ta đổi từ **textBox1** thành **txtN**.

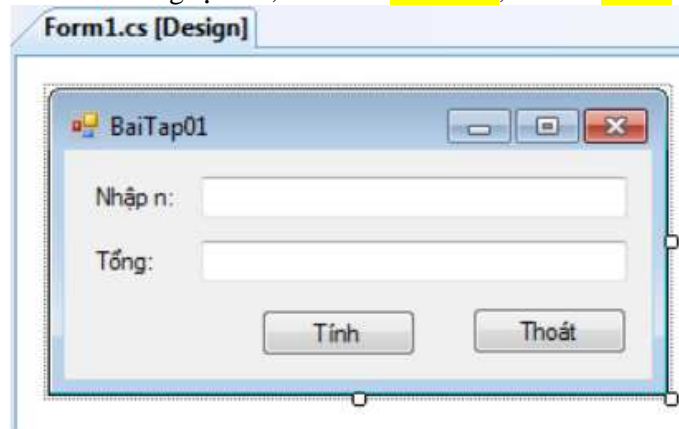


Tương tự thêm vào form một label có thuộc tính Text đổi thành **Tổng** và một textbox có thuộc tính Name đổi thành **txtTong**.



Tiếp theo, kéo một Button từ Toolbox và thả vào Form, đổi thuộc tính Name của button này là **btnTinh**, text là **Tính**.

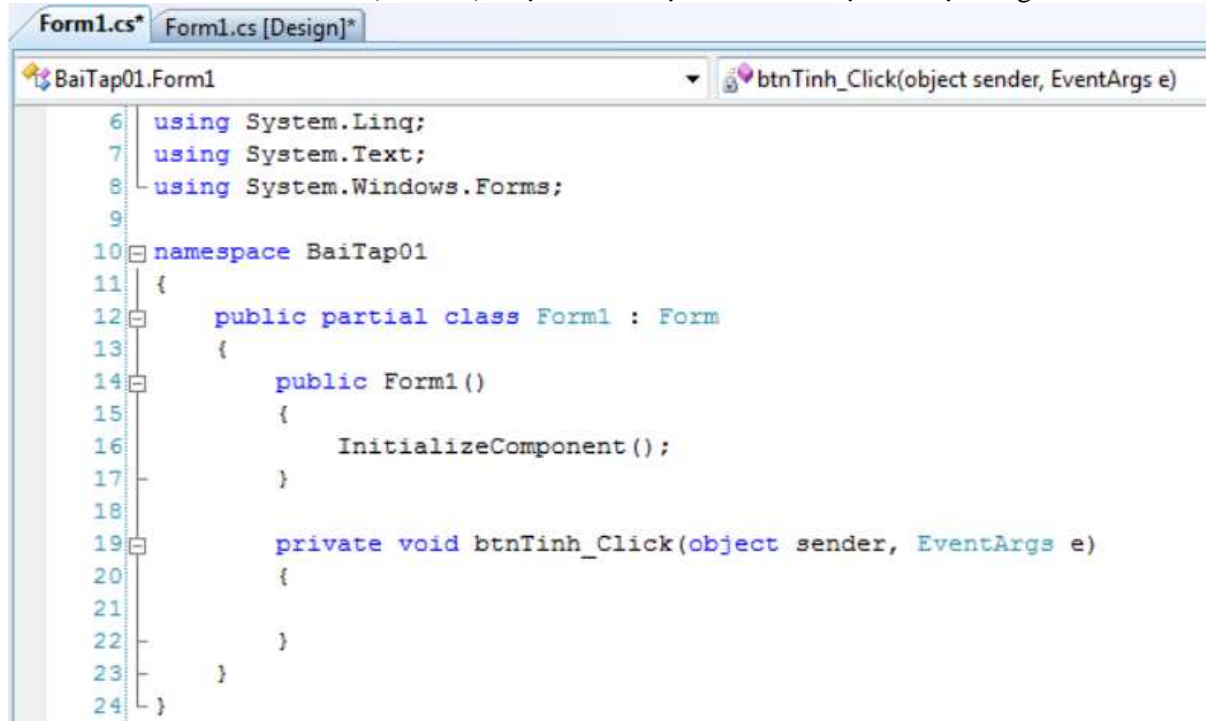
Kéo thả thêm một Button nữa tương tự trên, Name = **btnThoat**, Text = **Thoát**.



Đưa chuột về các góc của Form để thay đổi kích thước cửa sổ.

1.2.3. Xử lý sự kiện click chuột lên button

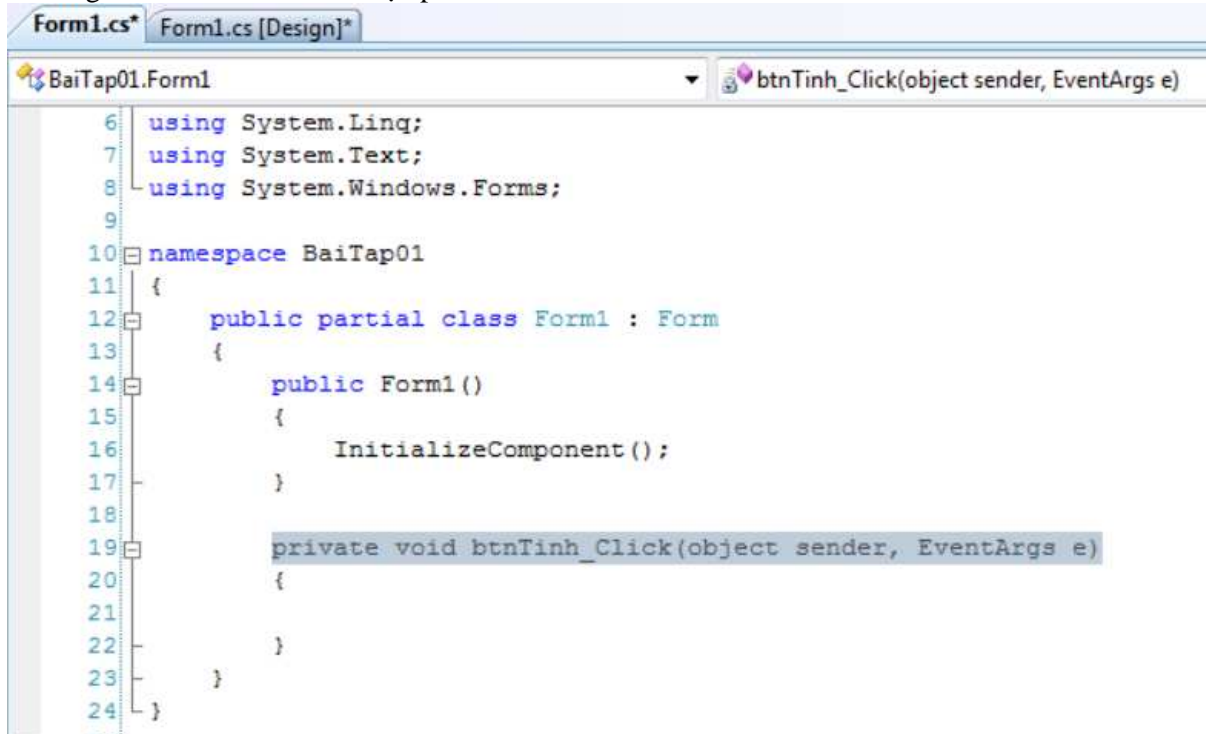
Double click vào button Tính (btnTinh), một cửa sổ soạn thảo xuất hiện với nội dung như sau:



```
6 using System.Linq;
7 using System.Text;
8 using System.Windows.Forms;
9
10 namespace BaiTap01
11 {
12     public partial class Form1 : Form
13     {
14         public Form1()
15         {
16             InitializeComponent();
17         }
18
19         private void btnTinh_Click(object sender, EventArgs e)
20         {
21         }
22     }
23 }
24
```

Chú ý: Cửa sổ này là Form1.cs tương ứng với Form1.cs[Design].

Phương thức btnTinh_Click được phát sinh do ta vừa double click vào btnTinh.



```
6 using System.Linq;
7 using System.Text;
8 using System.Windows.Forms;
9
10 namespace BaiTap01
11 {
12     public partial class Form1 : Form
13     {
14         public Form1()
15         {
16             InitializeComponent();
17         }
18
19         private void btnTinh_Click(object sender, EventArgs e)
20         {
21         }
22     }
23 }
24
```

Ta định nghĩa cho lớp này như sau:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace BaiTap01
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnTinh_Click(object sender, EventArgs e)
        {
            string text = txtN.Text;
            int n = int.Parse(text);

            long s = 0;
            for (int i = 1; i <= n; i++)
            {
                s = s + i;
            }

            txtTong.Text = s.ToString();
        }
    }
}
```

Tiếp theo ta xử lý sự kiện click chuột cho button btnThoat.

Quay lại Form1.cs[Design], double click lên button Thoát, Form1.cs có nội dung thay đổi như sau:

Bài tập thực hành lập trình Windows Form với C#

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace BaiTap01
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnTinh_Click(object sender, EventArgs e)
        {
            string text = txtN.Text;
            int n = int.Parse(text);

            long s = 0;
            for (int i = 1; i <= n; i++)
            {
                s = s + i;
            }

            txtTong.Text = s.ToString();
        }

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

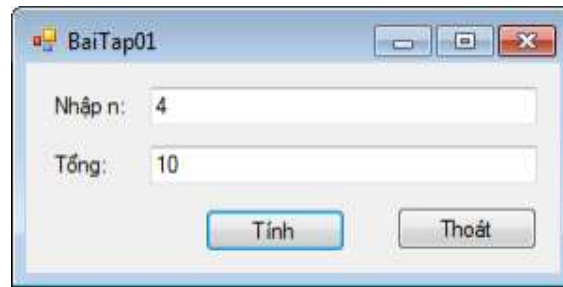
Ta xử lý cho sự kiện này như sau:

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

Chạy thử chương trình (F5):



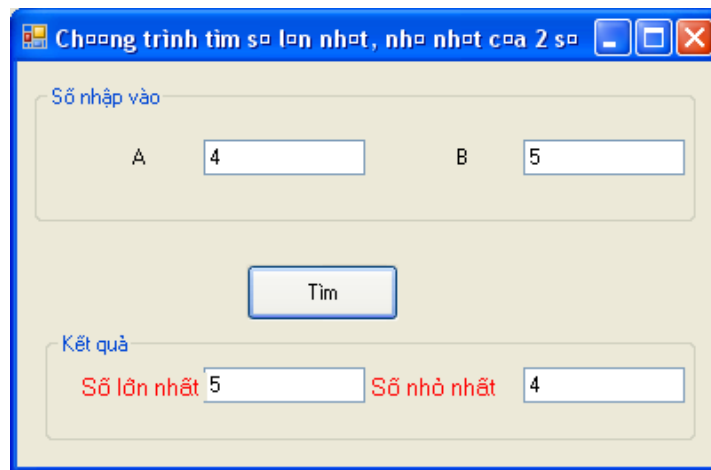
Nhập n = 4 và click vào button Tính. Kết quả xuất hiện ở txtTong:



Click vào button Thoát để kết nối chương trình.

2. Bài tập có gợi ý

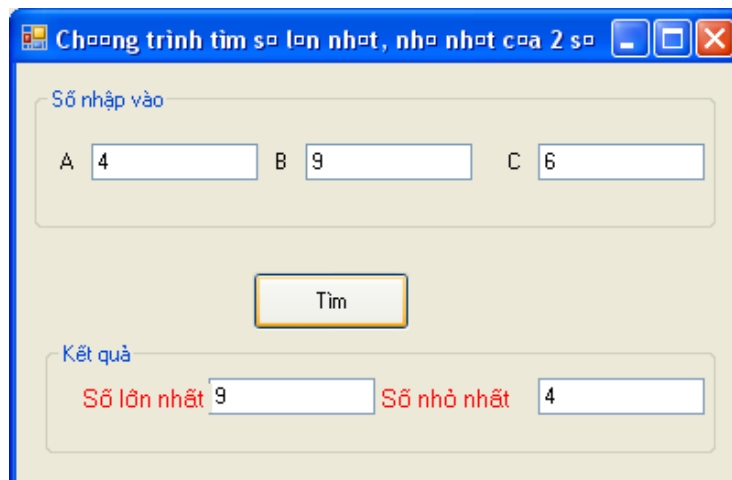
2.1. Bài 01 – Viết chương trình nhập vào 2 số a và b, cho biết số lớn nhất và nhỏ nhất trong 2 số a,b với giao diện như sau



Hình 1: Chương trình tìm Min, Max 2 số

Hướng dẫn gợi ý: sử dụng cấu trúc If ...Else

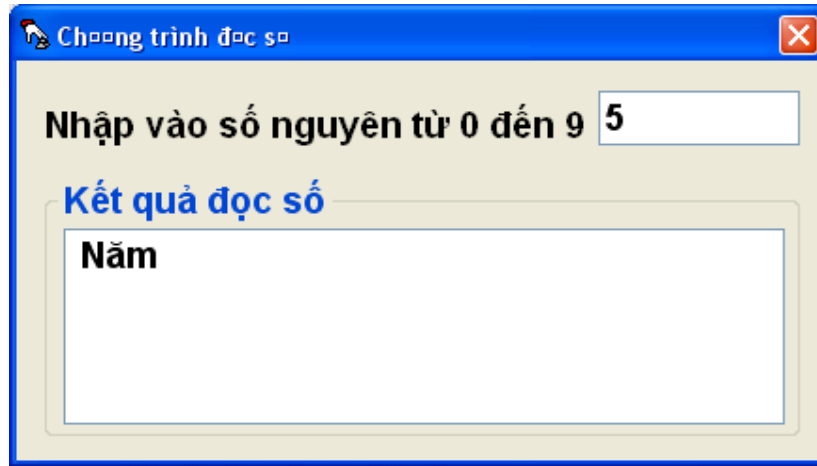
2.2. Bài 02 – Viết chương trình nhập vào 2 số a, b và c cho biết số lớn nhất và nhỏ nhất trong 3 số a, b và c với giao diện như sau



Hình 2: Chương trình tìm Min, Max 3 số

2.3. Bài 03 - Nhập vào một số nguyên từ 0 đến 9, hiển thị bằng “chữ” các ký số trên, ví dụ:

- Nhập 1 → “Một”
- Nhập 2 → “Hai”
- Nhập 3 → “Ba”
- ..
- Nhập 9 → “Chín”

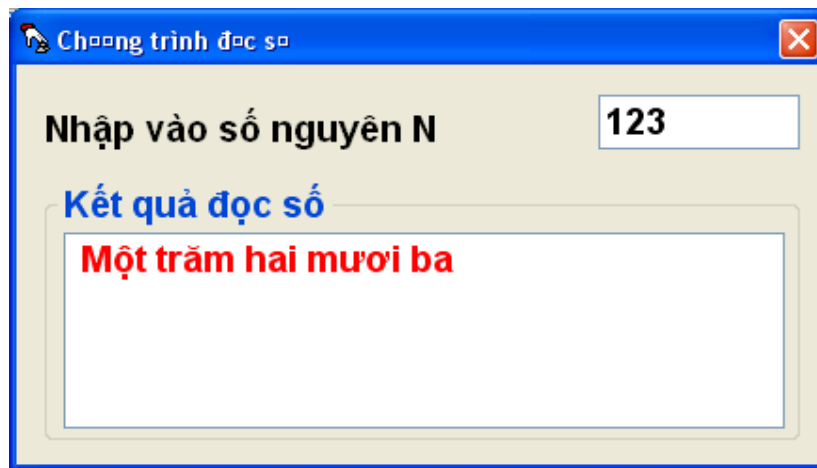


Hình 3: Chương trình đọc số

Hướng dẫn gợi ý: sử dụng cấu trúc **switch ... case**

2.4. Bài 04 - Nhập vào một số nguyên dương N, hiển thị bằng “chữ” số vừa nhập trên, ví dụ:

- Nhập 1 → “Một”
- Nhập 12 → “Mười hai”
- Nhập 123 → “Một trăm hai mươi ba”



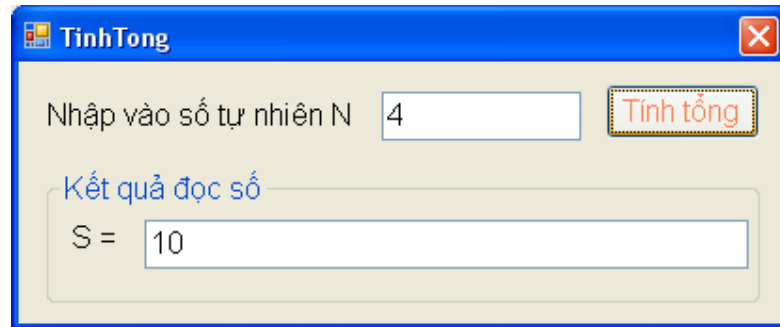
Hình 4: Chương trình đọc số tổng quát

Hướng dẫn gợi ý: sử dụng cấu trúc **switch ... case**

2.5. Bài 05 - Viết chương trình nhập vào giá trị nguyên dương N, tính tổng

$$S = 1 + 2 + 3 + \dots + N$$

với giao diện như sau:



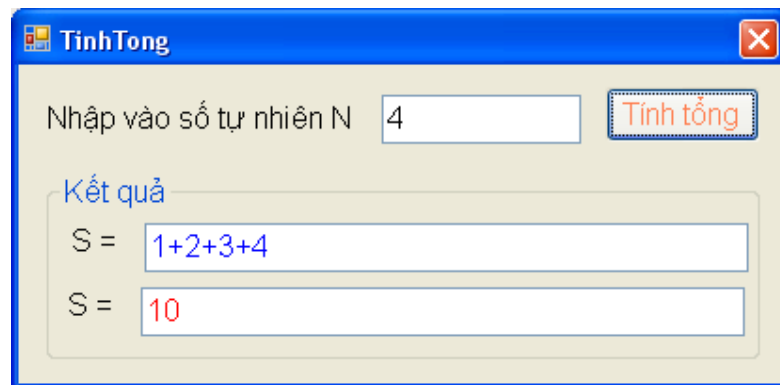
Hình 5: CT tính tổng N số nguyên liên tiếp - 1

Hướng dẫn gợi ý: sử dụng vòng lặp While, For

2.6. Bài 06- Viết chương trình nhập vào giá trị nguyên dương N, tính tổng

$$S = 1 + 2 + 3 + \dots + N$$

với giao diện như sau

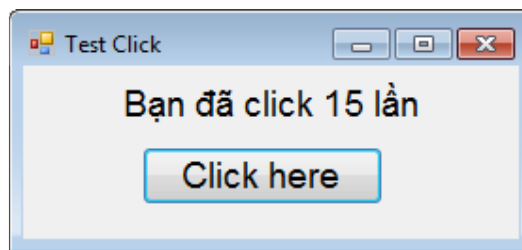


Hình 6: CT tính tổng N số nguyên liên tiếp - 2

Tương tự tính tổng

- $S = 1 + 3 + 5 + 7 + \dots + (2N + 1)$
- $S = 2 + 4 + 6 + 8 + \dots + 2N$.

2.7. Bài 07: Viết chương trình gồm 1 form trong đó có 1 button và 1 label hiển thị số lần nhấn button đó.



2.8. Bài 08- Viết chương trình nhập vào giá trị nguyên dương N, và số thực X, tính tổng

$$S = X + X^2 + X^3 + X^4 + \dots + X^N$$

Tính tổng

Nhập vào số tự nhiên N: 4

Nhập vào số thực X: 2

Tính tổng

Kết quả

S = $X+X^2+X^3+X^4$

S = $2+2^2+2^3+2^4$

S = $2+4+8+16$

S = 30

Hình 7: CT tính tổng lũy thừa

Tương tự tính tổng

- $S = X + X^3 + X^5 + X^7 + \dots + X^{2N+1}$
- $S = X^2 + X^4 + X^6 + \dots + X^{2N}$

2.9. Bài 09 – Thiết kế giao diện và cài đặt chương trình như sau:

Chương trình Máy tính đơn giản

Số hạng thứ nhất:

Số hạng thứ hai:

Tổng:

Phép toán

☒ Cộng

☐ Trừ

☐ Nhân

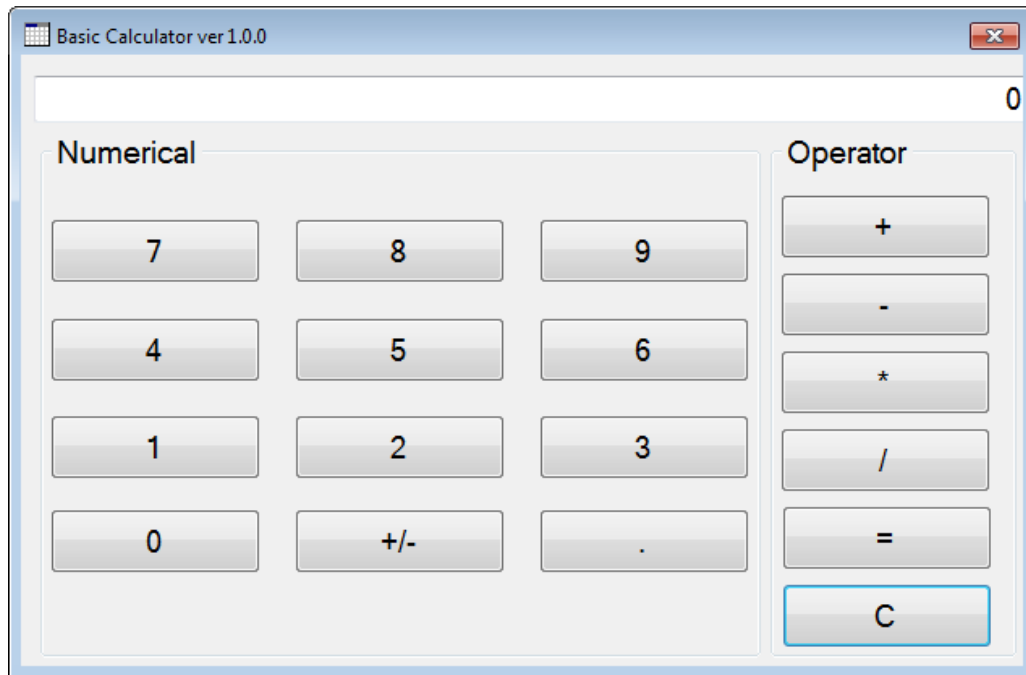
☐ Chia

Thoát **Tính**

Hình 8: Các phép toán cơ bản

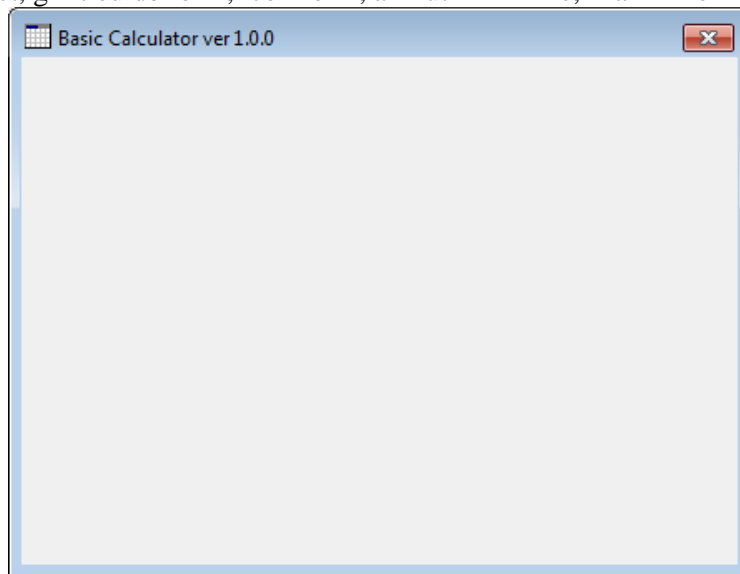
2.10. Bài 10 – (BT có hướng dẫn) Máy tính đơn giản thực hiện các chức năng +, -, *, /.

MÁY TÍNH ĐƠN GIẢN¹



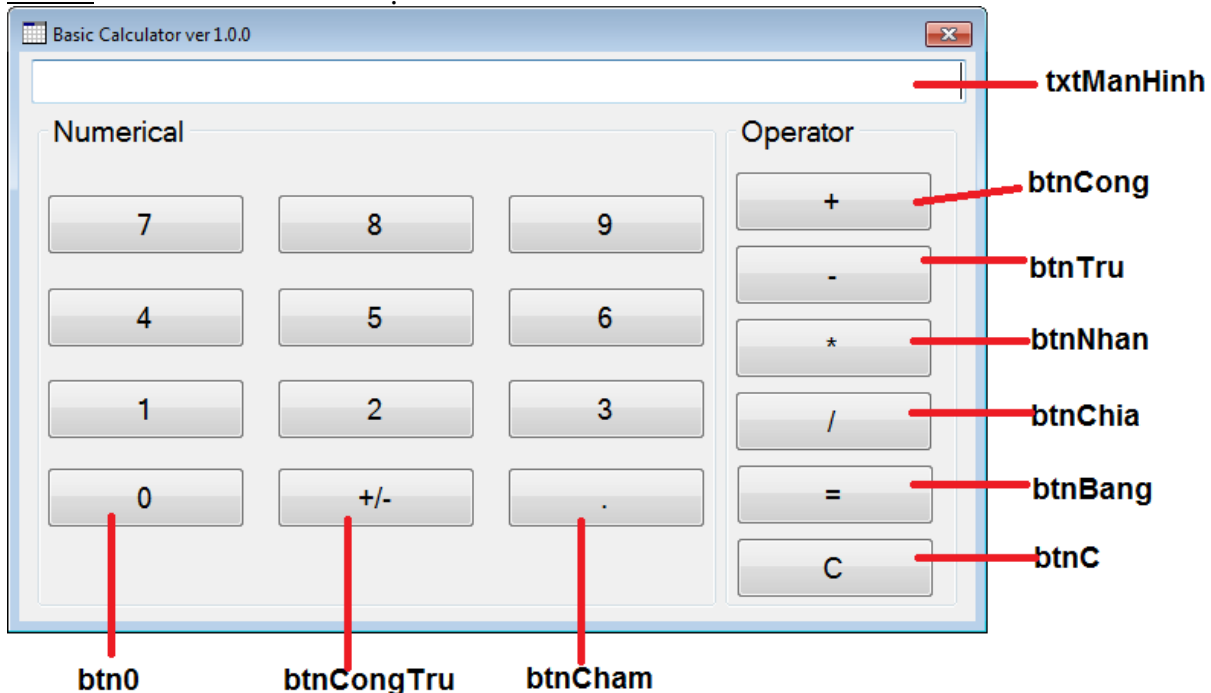
Hình 9: Chương trình máy tính đơn giản

Bước 1: Tạo Project, ghi tiêu đề form, icon form, ấn nút Minimize, Maximize



¹ Xử lý cho nút X: Sự kiện **FormClosing()**

Bước 2: Thiết kế màn hình và đặt tên như hình sau:



Trong đó các phím số đặt tên từ btn0 → btn9.

Bước 3: Khai báo các biến cần thiết

- `string chuoitam = "";` //Dùng để lưu giữ tạm giá trị của số nhập vào
- `char toantu;` //Dùng lưu kí hiệu toán tử của phép toán, vd: +, -, *, /
- `double[] toanhang = new double[2];` //Lưu 2 toán hạng (số hạng) của phép toán
- `double ketqua;` //Dùng lưu kết quả của phép toán
- `int buoc = 1;` //Dùng lưu giữ số bước thực hiện phép toán

//Khai báo biến

```
string chuoitam = "";
char toantu;
double[] toanhang = new double[2];
double ketqua;
int buoc = 1;
```

Bước 4: Viết hàm cho chung cho các phím số (0 → 9)

Đặt tên cho phím vừa bấm là b. Nếu b rỗng, kết thúc hàm. Ngoài ra chúng ta cũng loại trừ trường hợp có rất nhiều số 0 đứng trước một số khác (ví dụ 000000001 chính là 1). Do đó `b.Text == "0" && chuoitam.Length == 0` thì kết thúc hàm.

//Hàm bấm phím số

```
private void NumberButtons(object sender, EventArgs e)
{
    Button b = sender as Button;
    if ((b == null) || (b.Text == "0" && chuoitam.Length == 0)) return;
    chuoitam += b.Text;
    txtManHinh.Text = chuoitam;
}
```

Lưu ý: Cho tất cả các phím số gọi hàm này ứng với sự kiện click.

Bước 5: Viết hàm xử lý cho nút +/- (dương/âm)

//Hàm xử lý cho nút +/- (dương/âm)

```
private void btnCongTru_Click(object sender, EventArgs e)
{
    if (chuoitam.Contains('-'))
        chuoitam = chuoitam.Replace("-", "");
    else
        chuoitam = "-" + chuoitam;
    txtManHinh.Text = chuoitam;
}
```

Bước 6: Viết hàm xử lý cho phím .

//Hàm xử lý phím chấm

```
private void btnCham_Click(object sender, EventArgs e)
{
    if (!chuoitam.Contains('.'))
    {
        chuoitam = chuoitam + ".";
        txtManHinh.Text = chuoitam;
    }
}
```

Bước 7: Viết hàm xử lý cho nút C

//Hàm xử lý nút C

```
private void btnC_Click(object sender, EventArgs e)
{
    if (chuoitam.Length == 0)
    {
        buoc = 1;
        toanhang[0] = toanhang[1] = 0.0;
        toantu = ' ';
        ketqua = 0.0;
    }
    else
        chuoitam = "0";
    txtManHinh.Text = chuoitam;
}
```

Bước 8: Viết hàm xử lý cho phím bằng²

//Hàm bấm phím số

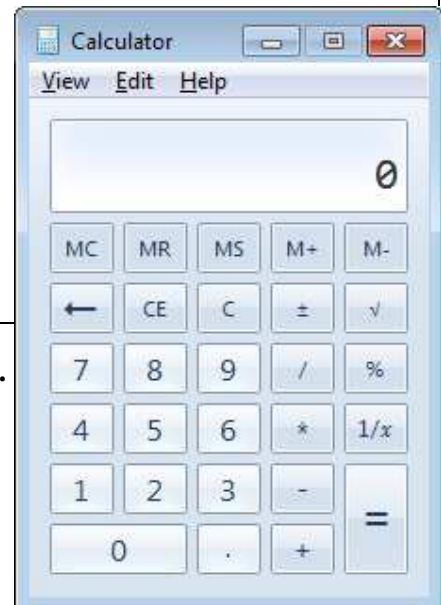
```
private void btnBang_Click(object sender, EventArgs e)
{
    if (chuoitam.Length != 0)
        toanhang[1] = Double.Parse(chuoitam);
    switch (toantu)
    {
        case '+': ketqua = toanhang[0] + toanhang[1]; break;
        case '-': ketqua = toanhang[0] - toanhang[1]; break;
        case '*': ketqua = toanhang[0] * toanhang[1]; break;
        case '/': ketqua = toanhang[0] / toanhang[1]; break;
    }
    txtManHinh.Text = ketqua.ToString();
    //Gán lại giá trị ban đầu
    buoc = 1;
    chuoitam = "";
}
```

² Sinh viên tự hoàn chỉnh thêm cho trường hợp bấm nhiều dấu = liên tiếp.

Bước 9: Viết hàm xử lý các phím thực hiện phép toán³

//Hàm bấm phím số

```
private void Operations(object sender, EventArgs e)
{
    Button b = sender as Button;
    if(buoc == 1)
        toantu = b.Text[0]; //Lấy kí hiệu toán tử
    if (chuoitam.Length == 0)
        toanhang[buoc - 1] = ketqua;
    else
        toanhang[buoc - 1] = Double.Parse(chuoitam);
    if (buoc == 2)
    {
        btnBang_Click(null, null);
        toantu = b.Text[0];
        toanhang[0] = ketqua;
        buoc = 2;
    }
    else
    {
        txtManHinh.Text = toanhang[0].ToString();
        buoc++;
    }
    chuoitam = "";
}
```



2.11. Bài 11: Chương trình máy tính của Windows 7.

Thiết kế máy tính bỏ túi có các chức năng như hình bên.
Không cần thiết kế menu.

³ Sinh viên tự hoàn chỉnh thêm cho trường hợp bấm nhiều lần 1 toán tử.