

BÀI THỰC HÀNH SỐ 8:
CÀI ĐẶT MỘT SỐ KỸ THUẬT
LẬP TRÌNH HĐT TRONG C#.NET_{5/10}

I. Mục tiêu bài học

- Kiến thức:

- Củng cố kiến thức và kỹ năng sử dụng kỹ thuật lập trình hướng đối tượng trong lập trình :
Chồng hàm, chồng toán tử, kế thừa

- Kỹ năng

- Vận dụng cài đặt kỹ thuật kế thừa, chồng hàm, chồng toán tử trong bài toán quản lý sinh viên.

- Thái độ

- Khả năng tư duy, cẩn thận trong công việc

II. Bài tập có gợi ý

1. Windows Form: Chồng toán tử

STT	LOẠI ĐK	TÊN ĐK	TEXT	CHÚ THÍCH
1	Label		CHỒNG TOÁN TỬ	Font: size: 16
2	Label		Tử số	
3	Label		Mẫu số	
4	Label		Kết quả	
5	Label	lblKQ		Borderstyle: fixed single, Autosize: false
6	TextBox	txtTS		
7	TextBox	txtMS		
8	Button	btnPS1	Phân số 1	
9	Button	btnPS2	Phân số 2	
10	Button	btnCong	Cộng	
11	Button	btnTru	Trừ	
12	Button	btnNhan	Nhân	
13	Button	btnChia	Chia	

a. Lớp Phân số

```
public class Phanso
{
    private int tuso;
    private int mauso;
    //Thuộc tính
    public int TUSO
    {
        get { return tuso; }
        set { tuso = value; }
    }
    public int MAUSO
    {
        get { return mauso; }
        set { mauso = value; }
    }
    public Phanso() {}
    public Phanso(int a, int b)
    {
        TUSO = a;
        MAUSO = b;
    }
    public override string ToString()
    {
        return TUSO.ToString() + "/" + MAUSO.ToString();
    }
    public static Phanso operator +(Phanso x, Phanso y)
    {
        return new Phanso(x.TUSO * y.MAUSO + y.TUSO * x.MAUSO, x.MAUSO * y.MAUSO);
    }
    //Cung cap thay the cho chong toan tu +
    public static Phanso Cong(Phanso x, Phanso y)
    {
        return x + y;
    }
    public static Phanso operator -(Phanso x, Phanso y)
    {
        return new Phanso(x.TUSO * y.MAUSO - y.TUSO * x.MAUSO, x.MAUSO * y.MAUSO);
    }
    //Cung cap thay the cho chong toan tu -
    public static Phanso Tru(Phanso x, Phanso y)
    {
        return x - y;
    }
    public static Phanso operator *(Phanso x, Phanso y)
    {
        return new Phanso(x.TUSO * y.TUSO, x.MAUSO * y.MAUSO);
    }
    public static Phanso Nhan(Phanso x, Phanso y)
    {
        return x * y;
    }
    public static Phanso operator /(Phanso x, Phanso y)
    {
        return new Phanso(x.TUSO * y.MAUSO, x.MAUSO * y.TUSO);
    }
    public static Phanso Chia(Phanso x, Phanso y)
    {
        return x / y;
    }
}
```

```
}
```

b. Lớp Form: frmPhanso

```
public partial class frmPhanso : Form
```

```
{
```

```
    public frmPhanso()
```

```
    {
```

```
        InitializeComponent();
```

```
    }
```

```
    Phanso ps1 = new Phanso();
```

```
    Phanso ps2 = new Phanso();
```

```
    private void btnPS1_Click(object sender, EventArgs e)
```

```
    {
```

```
        ps1.TUSO = Int32.Parse(txtTS.Text);
```

```
        ps1.MAUSO = Int32.Parse(txtMS.Text);
```

```
        txtTS.Clear();
```

```
        txtMS.Clear();
```

```
        txtTS.Focus();
```

```
    }
```

```
    private void btnPS2_Click(object sender, EventArgs e)
```

```
    {
```

```
        ps2.TUSO = Int32.Parse(txtTS.Text);
```

```
        ps2.MAUSO = Int32.Parse(txtMS.Text);
```

```
        txtTS.Clear();
```

```
        txtMS.Clear();
```

```
        txtTS.Focus();
```

```
    }
```

```
    public int UCLN(int a, int b)
```

```
    {
```

```
        if (b == 0)
```

```
            return a;
```

```
        else
```

```
            return UCLN(b, a % b);
```

```
    }
```

```
    private void btnCong_Click(object sender, EventArgs e)
```

```
    {
```

```
        Phanso kq = new Phanso();
```

```
        kq = ps1 + ps2;
```

```
        int ucln = UCLN(kq.TUSO, kq.MAUSO);
```

```
        kq.TUSO = kq.TUSO / ucln;
```

```
        kq.MAUSO = kq.MAUSO / ucln;
```

```
        if (kq.TUSO % kq.MAUSO == 0)
```

```
            lblKQ.Text = ps1.ToString() + " + " + ps2.ToString() + " = " +  
(kq.TUSO / kq.MAUSO).ToString();
```

```
        else
```

```
            lblKQ.Text = ps1.ToString() + " + " + ps2.ToString() + " = " +  
kq.ToString();
```

```
    }
```

```
    private void btnTru_Click(object sender, EventArgs e)
```

```
    {
```

```
        Phanso kq = new Phanso();
```

```
        kq = ps1 - ps2;
```

```
        int ucln = UCLN(kq.TUSO, kq.MAUSO);
```

```
        kq.TUSO = kq.TUSO / ucln;
```

```
        kq.MAUSO = kq.MAUSO / ucln;
```

```

        if (kq.TUSO % kq.MAUSO == 0)
            lblKQ.Text = ps1.ToString() + " - " + ps2.ToString() + " = " +
+ (kq.TUSO / kq.MAUSO).ToString();
        else
            lblKQ.Text = ps1.ToString() + " - " + ps2.ToString() + " = " +
kq.ToString();
    }

    private void btnNhan_Click(object sender, EventArgs e)
    {
        Phanso kq = new Phanso();
        kq = ps1 * ps2;
        int ucln = UCLN(kq.TUSO, kq.MAUSO);
        kq.TUSO = kq.TUSO / ucln;
        kq.MAUSO = kq.MAUSO / ucln;
        if (kq.TUSO % kq.MAUSO == 0)
            lblKQ.Text = ps1.ToString() + " * " + ps2.ToString() + " = " +
(kq.TUSO / kq.MAUSO).ToString();
        else
            lblKQ.Text = ps1.ToString() + " * " + ps2.ToString() + " = " +
kq.ToString();
    }

    private void btnChia_Click(object sender, EventArgs e)
    {
        Phanso kq = new Phanso();
        kq = ps1 / ps2;
        int ucln = UCLN(kq.TUSO, kq.MAUSO);
        kq.TUSO = kq.TUSO / ucln;
        kq.MAUSO = kq.MAUSO / ucln;
        if (kq.TUSO % kq.MAUSO == 0)
            lblKQ.Text = ps1.ToString() + " / " + ps2.ToString() + " = " +
(kq.TUSO / kq.MAUSO).ToString();
        else
            lblKQ.Text = ps1.ToString() + " / " + ps2.ToString() + " = " +
kq.ToString();
    }
}

```

III. Bài tập tự làm

1. Tạo phương thức rút gọn phân số trong lớp Phanso, sử dụng phương thức đó trong ứng dụng windows form với các chức năng tính Cộng, trừ, nhân, chia 2 phân số.
2. Tạo lớp Ma trận, thực hiện chồng toán tử với phương thức cộng 2 ma trận

```

class Program
{
    static void Main(string[] args)
    {
        matran mt1 = new matran();
        matran mt2 = new matran();
        Console.WriteLine("Nhap ma tran 1:");
        mt1.nhap(mt1);
        Console.WriteLine("Ma tran 1: ");
        mt1.hienthi(mt1);
    }
}

```

```

        Console.WriteLine("Nhap ma tran 1:");
        mt2.nhap(mt2);
        Console.WriteLine("Ma tran 2: ");
        mt1.hienthi(mt2);
        matran mt3 = mt1 + mt2;
        Console.WriteLine("Tong 2 ma tran:");
        mt3.hienthi(mt3);
        Console.ReadLine();
    }
}
public class matran
{
    public const int size = 3;
    private int[,] mangMT = new int[size, size];
    public int this[int x, int y]
    {
        get { return mangMT[x, y]; }
        set { mangMT[x, y] = value; }
    }
    public void nhap(matran mt)
    {
        for (int i = 0; i < size; i++)
            for (int j = 0; j < size; j++)
                mt[i, j] = Int32.Parse(Console.ReadLine());
    }
    public void hienthi(matran mt)
    {
        for (int i = 0; i < size; i++)
        {
            for (int j = 0; j < size; j++)
                Console.Write(mt[i, j] + " ");
            Console.WriteLine();
        }
    }
    public static matran operator +(matran mt1, matran mt2)
    {
        matran kq = new matran();
        for (int i = 0; i < size; i++)
            for (int j = 0; j < size; j++)
                kq[i, j] = mt1[i, j] + mt2[i, j];
        return kq;
    }
}

```