

```
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 Desene
```

```
{
    public partial class Form1 : Form
    {
        Bitmap Im1, Im2, Im3; //
        private Image loadedImage;
        //class punct { public double x,y; }
        class muchie { public int st, dr; }
        class varf { public double x,y,z;
                    public varf(int X, int Y, int Z) { x = X; y = Y; z = Z; }
        }
        int u1,v1, u2,v2; // ViewPort
        double a, b, c, d; // Window
        double Raza, Alfa; // Pr. Par / Persp.!

        int u(double x) { return (int) ((x - a) / (b - a) * (u2 - u1) + u1); }
        int v(double y) { return (int) ((y - d) / (c - d) * (v2 - v1) + v1); }

        void ViewPort(int x1,int y1,int x2,int y2) { u1=x1; v1=y1; u2=x2; v2=y2; }
        void Window (double x1,double y1,double x2,double y2) { a=x1; d=y1; b=x2; c=y2; }
    }
}
```

```
void RotOy(varf P, double Alfa)
{
    double xp;
    xp = P.x*Math.Cos(Alfa)-P.z*Math.Sin(Alfa);
    P.z=P.x*Math.Sin(Alfa)+P.z*Math.Cos(Alfa);    P.x=xp;
}

void DefPr (double r, double a)  { Raza=r; Alfa=a; } // r=1; a=0.8;

//double PrX (double x, double z) { return x+Raza*z*cos(Alfa); }
//double PrY (double y, double z) { return y+Raza*z*sin(Alfa); }

double PrX (double x, double z) { double d=Raza, q=Alfa; return x*(d-q)/(d-z); }
double PrY (double y, double z) { double d=Raza, q=Alfa; return y*(d-q)/(d-z); }

double Px(varf P) { return PrX(P.x, P.z); }
double Py(varf P) { return PrY(P.y, P.z); }

public Form1()
{
    InitializeComponent();
    Im1 = new Bitmap(pictureBox1.Image);
    Im2 = new Bitmap(pictureBox2.Image);
    Im3 = new Bitmap(pictureBox3.Image);
    // Paralela: DefPr(1, 0.8);
    // DefPr(1000, 10); // Perspectiva(d,q):
}
}
```

```
private void redBlueToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Drawing.Pen myPen;
    myPen = new System.Drawing.Pen(System.Drawing.Color.Chocolate);
    System.Drawing.Graphics formGraphics = this.CreateGraphics();

    System.IO.StreamReader Fc = new System.IO.StreamReader("Piramida.Txt");
    String Line = Fc.ReadLine();
    String[] Split = Line.Split(new Char[] { '|', ',', '\t' });
    int n = Convert.ToInt32(Split[0]);

    varf [] V=new varf[n+1];
    for (int i = 1; i <= n; i++)
    {
        Line = Fc.ReadLine();
        Split = Line.Split(new Char[] { '|', ',', '\t' });
        int X = Convert.ToInt32(Split[0]);
        int Z = Convert.ToInt32(Split[1]);
        int Y = Convert.ToInt32(Split[2]) - 100; // y <---> z
        V[i] = new varf(X, Y, Z); // V V V !!!
    }

    Line = Fc.ReadLine();
    Split = Line.Split(new Char[] { '|', ',', '\t' });
    int m = Convert.ToInt32(Split[0]);
    muchie[] M = new muchie[m+1];
```

```
for (int j = 1; j <= m; j++)
{
    Line = Fc.ReadLine();
    Split = Line.Split(new Char[] { ',', '}', '\t' });
    M[j] = new muchie();
    M[j].st = Convert.ToInt32(Split[0]);
    M[j].dr = Convert.ToInt32(Split[1]);
}
Fc.Close();
```

```
ViewPort(300, 100, 800, 700);
```

```
for (int i = 1; i <= n; i++)
{
    RotOy(V[i], 0.25);
}
```

```
DefPr(500, 100); // Persp.(d,q);
```

```
a = b=Px(V[1]); c = d=Py(V[1]);
```

```
for (int i = 2; i <= n; i++)
{
    double px= Px(V[i]);
    if (px < a) a=px; else if (px>b) b=px;
    double py= Py(V[i]);
    if (py < c) c=py; else if (py>d) d=py;
}
Window (a,d, b, c);
```

```
for (int j = 1; j <= m; j++)
{
    myPen.Color = System.Drawing.Color.Aqua;
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st]),v(Py(V[M[j].st))),u(Px(V[M[j].dr]),v(Py(V[M[j].dr))));
}

for (int i = 1; i <= n; i++)
{
    RotOy(V[i], 0.1);
}

for (int j = 1; j <= m; j++)
{
    myPen.Color = System.Drawing.Color.Tomato;
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st]), v(Py(V[M[j].st))), u(Px(V[M[j].dr]), v(Py(V[M[j].dr))));
}

// a = b = Px(V[100]); // Debug

// myPen.Color = System.Drawing.Color.Blue;
// formGraphics.DrawLine(myPen, 400, 500, 900, 400);

myPen.Dispose();
formGraphics.Dispose();
}
```

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

// ...

double x_la_y (double x, double y)
{
    return Math.Pow(x,y);
}

double f(double t)
{
    double e=0.008856;
    if (t>e) return x_la_y(t,1/3.0); else return 7.787*t+16.0/116.0;
}

double Tr(double C)
{
    C/=255.0;
    if (C>0.04045) C=x_la_y((C+0.055)/1.055,2.4); else C/=12.92;
    return C*100;
}

int Dm(int x)
{
    return Min(x, 400 - x);
}

int Min(int a, int b)
{
    if (a < b) return a; else return b;
}
```

```
void Swap(ref int a, ref int b)
{
    int Aux = a; a = b; b = Aux;
}
```

```
private void labToolStripMenuItem_Click(object sender, EventArgs e)
{
```

```
    pictureBox1.Visible = true;
    ViewPort(0, 0, 255, 255);
    Window (0, 0, 400, 400);
    for (int x = 0; x < 400; x++)
        for (int y = 0; y < 400; y++)
        {
            // Im1.SetPixel(x, y, Color.White);
            // ViewPort(Dm(x), Dm(y), 255, 255);
            int r = u(x);
            int g = 255 - u(x);
            int b = v(y);
            Im1.SetPixel(x, y, Color.FromArgb(r,g,b));
        }
```

```
checkbox1.Visible = true;
```

```
ViewPort(0,0, 400, 400);
Window (-100, 100, 100, -100);
```

```
for (int r = 0; r < 256; r++)
  for (int g = 0; g < 256; g++) // +b
  {
    int b;
    if (checkBox1.Checked) b = 0; else b=255; // -b

    // (R,G,B)->(L,a,b) ...
    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
    double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
    double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
    double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;
    double W = 255;
    double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
    double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
    double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;

    double E = 0.008856;
    double L, a_, b_;

    X = X / Xn; Y = Y / Yn; Z = Z / Zn; if (Y > E)
      L = 116.0 * x_la_y(Y, 1 / 3.0) - 16;
    else L = 903.3 * Y;
    a_ = 500.0 * (f(X) - f(Y));
    b_ = 200.0 * (f(Y) - f(Z));

    // ... (R,G,B)->(L,a,b)
    Color c = Color.FromArgb(255, r,g,b); // &b
    Im1.SetPixel(u(a_), v(b_), c); Im1.SetPixel(u(a_)+1, v(b_), c);
  }
```



```
for (int r = 0; r < 256; r++)
  for (int b = 0; b < 256; b++)  // +b
  {
    int g;
    if (checkBox1.Checked) g = 0; else g = 255;

    // (R,G,B)->(L,a,b) ...
    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
    double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
    double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
    double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;

    double W = 255;
    double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
    double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
    double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;

    double E = 0.008856;
    double L, a_, b_;

    X = X / Xn; Y = Y / Yn; Z = Z / Zn; if (Y > E)
      L = 116.0 * x_la_y(Y, 1 / 3.0) - 16;
    else L = 903.3 * Y;
    a_ = 500.0 * (f(X) - f(Y));
    b_ = 200.0 * (f(Y) - f(Z));

    // ... (R,G,B)->(L,a,b)
    Color c = Color.FromArgb(255, r, g, b);  // &b
    Im1.SetPixel(u(a_), v(b_), c); Im1.SetPixel(u(a_) + 1, v(b_), c);
  }
```

```
for (int g = 0; g < 256; g++)
  for (int b = 0; b < 256; b++)  // +b
  {
    int r;
    if (checkBox1.Checked) r = 0; else r = 255;

    // (R,G,B)->(L,a,b) ...
    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
    double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
    double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
    double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;

    double W = 255;
    double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
    double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
    double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;

    double E = 0.008856;
    double L, a_, b_;

    X = X / Xn; Y = Y / Yn; Z = Z / Zn; if (Y > E)
      L = 116.0 * x_la_y(Y, 1 / 3.0) - 16;
    else L = 903.3 * Y;
    a_ = 500.0 * (f(X) - f(Y));
    b_ = 200.0 * (f(Y) - f(Z));

    // ... (R,G,B)->(L,a,b)
    Color c = Color.FromArgb(255, r, g, b);  // &b
    Im1.SetPixel(u(a_), v(b_), c); Im1.SetPixel(u(a_) + 1, v(b_), c);
  }
```

```
pictureBox1.Image = Im1;  
pictureBox1.Refresh();  
hideToolStripMenuItem.Visible = true;  
}
```

```
private void hideToolStripMenuItem_Click(object sender, EventArgs e)  
{  
    pictureBox1.Visible = false;  
    checkBox1.Visible = false;  
}
```

```
private void pictureBox1_Click(object sender, EventArgs e)  
{  
    saveFileDialog1.ShowDialog();  
    Im1.Save(saveFileDialog1.FileName);  
}
```

```
double Dn(int x1, int y1, int x2, int y2)  
{  
    return Math.Abs(x1 - x2) + Math.Abs(y1 - y2);  
}
```

```
private void xYZToolStripMenuItem_Click(object sender, EventArgs e)
{
    pictureBox1.Visible = true;
    ViewPort(0, 0, 255, 255);
    Window(0, 0, 400, 400);
    for (int x = 0; x < 400; x++)
        for (int y = 0; y < 400; y++)
        {
            //Im1.SetPixel(x, y, Color.White);
            /*
            double dG = Dn(x, y, 0, 0);
            double dB = Dn(x, y, 0, 400);
            double dR = Dn(x, y, 400, 400);
            double dY = Dn(x, y, 400, 0);

            double aG = dG / (dG + dB + dR + dY);
            double aB = dB / (dG + dB + dR + dY);
            double aR = dR / (dG + dB + dR + dY);
            double aY = dY / (dG + dB + dR + dY);

            int r = (int)(aG * 255 + aB * 255 + aR * 0 + aY * 0);
            int g = (int)(aG * 0 + aB * 255 + aR * 255 + aY * 0);
            int b = (int)(aG * 255 + aB * 0 + aR * 255 + aY * 255);
            Im1.SetPixel(x, y, Color.FromArgb(r, g, b));
            */
            Im1.SetPixel(x, y, Color.FromArgb(u(x), v(400-y), Math.Abs(v(y)-u(x)/4)));
        }

    ViewPort(0, 0, 400, 400);
    Window(0.1, 0.7, 0.7, 0);
}
```

```
for (int r = 0; r < 256; r++)
  for (int g = 0; g < 256; g++)  // +b
  {

    int b;
    if (checkBox1.Checked) b = 100; else b = 255;    //-b

    // (R,G,B)->(X,Y,Z) ...
    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
    //double X = (R * 0.49 + G * 0.31 + B * 0.2) / 0.17697;
    //double Y = (R * 0.17697 + G * 0.8124 + B * 0.01063) / 0.17697;
    //double Z = (R * 0.0 + G * 0.01 + B * 0.99) / 0.17697;
    double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
    double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
    double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;
    R = X / (X + Y + Z);
    G = Y / (X + Y + Z);
    Color c = Color.FromArgb(255, r, g, b);  // &b
    Im1.SetPixel(u(R), v(G), c); Im1.SetPixel(u(R) + 1, v(G), c);
  }
for (int r = 0; r < 256; r++)
  for (int b = 0; b < 256; b++)  // +b
  {
    int g;
    if (checkBox1.Checked) g = 100; else g = 255;

    // (R,G,B)->(X,Y,Z) ...
    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
```

```
double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;
R = X / (X + Y + Z);
G = Y / (X + Y + Z);
Color c = Color.FromArgb(255, r, g, b); // &b
Im1.SetPixel(u(R), v(G), c); Im1.SetPixel(u(R) + 1, v(G), c);
}
for (int g = 0; g < 256; g++)
for (int b = 0; b < 256; b++) // +b
{
int r;
if (checkBox1.Checked) r = 100; else r = 255;

// (R,G,B)->(X,Y,Z) ...
double R = Tr(r);
double G = Tr(g);
double B = Tr(b);
double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;

R = X / (X + Y + Z);
G = Y / (X + Y + Z);
Color c = Color.FromArgb(255, r, g, b); // &b
Im1.SetPixel(u(R), v(G), c); Im1.SetPixel(u(R) + 1, v(G), c);
}
pictureBox1.Image = Im1;
pictureBox1.Refresh();
hideToolStripMenuItem.Visible = true;
}
```

```
private void dToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Drawing.Pen myPen;
    myPen = new System.Drawing.Pen(System.Drawing.Color.Chocolate);
    System.Drawing.Graphics formGraphics = this.CreateGraphics();
    openFileDialog1.ShowDialog();
    System.IO.StreamReader Fc = new System.IO.StreamReader(openFileDialog1.FileName);

    String Line = Fc.ReadLine();
    String[] Split = Line.Split(new Char[] { ' ', ',', '\t' });
    int n = Convert.ToInt32(Split[0]);

    varf[] V = new varf[n + 10];
    for (int i = 1; i <= n; i++)
    {
        Line = Fc.ReadLine();
        Split = Line.Split(new Char[] { ' ', ',', '\t' });
        int X = Convert.ToInt32(Split[0]);
        int Z = Convert.ToInt32(Split[1]);
        int Y = Convert.ToInt32(Split[2]) - 100; // y <---> z
        V[i] = new varf(X, Y, Z); // V V V !!!
    }

    Line = Fc.ReadLine();
    Split = Line.Split(new Char[] { ' ', ',', '\t' });
    int m = Convert.ToInt32(Split[0]);
    muchie[] M = new muchie[m + 10];
    for (int j = 1; j <= m; j++)
    {
        Line = Fc.ReadLine();
        Split = Line.Split(new Char[] { ' ', ',', '\t' });
    }
}
```

```
M[j] = new muchie();
M[j].st = Convert.ToInt32(Split[0]);
M[j].dr = Convert.ToInt32(Split[1]);
}
Fc.Close();
int nn = n; int mm = m;
for (int i = 1; i <= 3; i++)
{
    nn++; V[nn] = new varf((int)V[i].x, -100, (int)V[i].z); mm++; M[mm] = new muchie(); M[mm].st = i; M[mm].dr = nn;
    nn++; V[nn] = new varf(0, -100, (int)V[i].z); mm++; M[mm] = new muchie(); M[mm].st = nn - 1; M[mm].dr = nn;
    nn++; V[nn] = new varf((int)V[i].x, -100, 0); mm++; M[mm] = new muchie(); M[mm].st = nn - 2; M[mm].dr = nn;
}
n = nn; m = mm;
ViewPort(400, 100, 1000, 800);
DefPr(500, 100); // Persp.(d,q);
for (int i = 1; i <= n; i++)
{
    RotOy(V[i], 0.25);
}

a = b = Px(V[1]); c = d = Py(V[1]);
for (int i = 2; i <= n; i++)
{
    double px = Px(V[i]);
    if (px < a) a = px; else if (px > b) b = px;
    double py = Py(V[i]);
    if (py < c) c = py; else if (py > d) d = py;
}
Window(a, d, b, c);
//DefPr(500, 100); // Persp.(d,q);
```



```
for (int j = 1; j <= m; j++)
{
    myPen.Color = System.Drawing.Color.Aqua;
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st])), v(Py(V[M[j].st])), u(Px(V[M[j].dr])), v(Py(V[M[j].dr]]));
}

for (int i = 1; i <= n; i++)
{
    RotOy(V[i], 0.1);
}

for (int j = 1; j <= m; j++)
{
    myPen.Color = System.Drawing.Color.Tomato;
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st])), v(Py(V[M[j].st])), u(Px(V[M[j].dr])), v(Py(V[M[j].dr]]));
}

// a = b = Px(V[100]); // Debug

// myPen.Color = System.Drawing.Color.Blue;
// formGraphics.DrawLine(myPen, 400, 500, 900, 400-nn*500);

myPen.Dispose();
formGraphics.Dispose();

}

int Pz(int a, int b)
{
    if (b > 0) return b; else return a;
}
```

```
private void luvToolStripMenuItem_Click(object sender, EventArgs e)
{
    pictureBox1.Visible = true;
    ViewPort(0, 0, 255, 255);
    Window(0, 0, 200, 400);
    for (int x = 0; x < 400; x++)
        for (int y = 0; y < 400; y++)
            {
                //Im1.SetPixel(x, y, Color.White);
                //ViewPort(Dm(x), Dm(y), 255, 255);
                int r; if (x < 200) r = u(x); else r = 255;
                int g; if (x < 200) g = 255; else g = 255 - u(x - 200);
                int b = v(y);
                r = Pz(r, r - r * y / 900);
                g = Pz(g, g - g * y / 900);
                Im1.SetPixel(x, y, Color.FromArgb(7 * r / 8, 7 * g / 8, 7 * b / 8));
            }

    checkBox1.Visible = true;

    ViewPort(0, 0, 400, 400);
    Window(-90, 110, 130, -110);

    for (int r = 0; r < 256; r++)
        for (int g = 0; g < 256; g++) // +b
            {

                int b;
                if (checkBox1.Checked) b = 200; else b = 255; // -b

                // (R,G,B)->(L,u,v) ...
            }
}
```

```
double R = Tr(r);
double G = Tr(g);
double B = Tr(b);
double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;

double W = 255;
double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;

double E = 0.008856;

double L, u_, v_, U_, V_, Un, Vn;

//X = X / Xn; Y = Y / Yn; Z = Z / Zn;

if (Y/Yn > E)
    L = 116.0 * x_la_y(Y/Yn, 1 / 3.0) - 16;
else L = 903.3 * Y/Yn;

//X = X / Xn; Y = Y / Yn; Z = Z / Zn;

U_ = 4 * X / (X + 15 * Y + 3 * Z);
V_ = 9 * Y / (X + 15 * Y + 3 * Z);

Un = 4 * Xn / (-2 * Xn + 12 * Yn + 3);
Vn = 9 * Yn / (-2 * Xn + 12 * Yn + 3);

Un=0.2009;
Vn=0.4610;
```

```
u_ = 13*L*(U_-Un);  
v_ = 13 * L * (V_ - Vn);
```

```
//double x=27*;
```

```
// ... (R,G,B)->(L,u,v)
```

```
Color c = Color.FromArgb(255, r, g, b); // &b  
Im1.SetPixel(u(u_), v(v_), c); Im1.SetPixel(u(u_) + 1, v(v_), c);
```

```
}
```

```
for (int r = 0; r < 256; r++)
```

```
for (int b = 0; b < 256; b++) // +b
```

```
{
```

```
int g;
```

```
if (checkBox1.Checked) g = 200; else g = 255;
```

```
// ... (R,G,B)->(L,u,v)
```

```
double R = Tr(r);
```

```
double G = Tr(g);
```

```
double B = Tr(b);
```

```
double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
```

```
double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
```

```
double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;
```

```
double W = 255;
```

```
double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
```

```
double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
```

```
double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;
```

```
double E = 0.008856;
```

```
double L, u_, v_, U_, V_, Un, Vn;
```

```
//X = X / Xn; Y = Y / Yn; Z = Z / Zn;
```

```
if (Y / Yn > E)
```

```
    L = 116.0 * x_la_y(Y / Yn, 1 / 3.0) - 16;
```

```
else L = 903.3 * Y / Yn;
```

```
//X = X / Xn; Y = Y / Yn; Z = Z / Zn;
```

```
U_ = 4 * X / (X + 15 * Y + 3 * Z);
```

```
V_ = 9 * Y / (X + 15 * Y + 3 * Z);
```

```
Un = 4 * Xn / (-2 * Xn + 12 * Yn + 3);
```

```
Vn = 9 * Yn / (-2 * Xn + 12 * Yn + 3);
```

```
Un = 0.2009;
```

```
Vn = 0.4610;
```

```
u_ = 13 * L * (U_ - Un);
```

```
v_ = 13 * L * (V_ - Vn); ;
```

```
// ... (R,G,B)->(L,u,v)
```

```
Color c = Color.FromArgb(255, r, g, b); // &b
```

```
Im1.SetPixel(u(u_), v(v_), c); Im1.SetPixel(u(u_) + 1, v(v_), c);
```

```
}
```

```
for (int g = 0; g < 256; g++)
  for (int b = 0; b < 256; b++)  // +b
  {
    int r;
    if (checkBox1.Checked) r = 200; else r = 255;

    // (R,G,B)->(L,u,v) ...

    double R = Tr(r);
    double G = Tr(g);
    double B = Tr(b);
    double X = R * 0.4124 + G * 0.3576 + B * 0.1805;
    double Y = R * 0.2126 + G * 0.7152 + B * 0.0722;
    double Z = R * 0.0193 + G * 0.1192 + B * 0.9505;

    double W = 255;
    double Xn = W * 0.4124 + W * 0.3576 + W * 0.1805;
    double Yn = W * 0.2126 + W * 0.7152 + W * 0.0722;
    double Zn = W * 0.0193 + W * 0.1192 + W * 0.9505;

    double E = 0.008856;

    double L, u_, v_, U_, V_, Un, Vn;

    //X = X / Xn; Y = Y / Yn; Z = Z / Zn;

    if (Y / Yn > E)
      L = 116.0 * x_la_y(Y / Yn, 1 / 3.0) - 16;
    else L = 903.3 * Y / Yn;

    //X = X / Xn; Y = Y / Yn; Z = Z / Zn;
```

```
U_ = 4 * X / (X + 15 * Y + 3 * Z);  
V_ = 9 * Y / (X + 15 * Y + 3 * Z);
```

```
Un = 4 * Xn / (-2 * Xn + 12 * Yn + 3);  
Vn = 9 * Yn / (-2 * Xn + 12 * Yn + 3);
```

```
Un = 0.2009;
```

```
Vn = 0.4610;
```

```
u_ = 13 * L * (U_ - Un);
```

```
v_ = 13 * L * (V_ - Vn); ;
```

```
// ... (R,G,B)->(L,u,v)
```

```
Color c = Color.FromArgb(255, r, g, b); // &b
```

```
Im1.SetPixel(u(u_), v(v_), c); Im1.SetPixel(u(u_) + 1, v(v_), c);
```

```
}
```

```
pictureBox1.Image = Im1;
```

```
pictureBox1.Refresh();
```

```
hideToolStripMenuItem.Visible = true;
```

```
}
```

```
private void button2_Click(object sender, EventArgs e)
```

```
{
```

```
openFileDialog1.ShowDialog();
```

```
loadedImage = Image.FromFile(openFileDialog1.FileName);
```

```
Im1 = new Bitmap(loadedImage);
```

```
pictureBox1.Image = Im1;
```

```
pictureBox1.Refresh();
```

```
}
```

```
private void button1_Click(object sender, EventArgs e)
{
    checkBox1.Visible = true;
    for (int i = 0; i < 400; i++)
        for (int j = 0; j < 400; j++)
            {
                Color c = Im1.GetPixel(i, j);
                if (checkBox1.Checked)
                {
                    int cR = c.R; int cB = c.B;
                    Swap(ref cR, ref cB);
                    Im1.SetPixel(i, j, Color.FromArgb(255, cR, c.G, cB));
                }
                else
                {
                    int m = (11 * c.R + 16 * c.G + 5 * c.B) / 32;
                    Im1.SetPixel(i, j, Color.FromArgb(255, m, m, m));
                }
            }
    pictureBox1.Image = Im1;
    pictureBox1.Refresh();
}

private void grayToolStripMenuItem_Click(object sender, EventArgs e)
{
    pictureBox1.Visible = true;
    button1.Visible = true;
    button2.Visible = true;
}

private void hideToolStripMenuItem1_Click(object sender, EventArgs e)
```



```
{  
    pictureBox1.Visible = false;  
    pictureBox2.Visible = false;  
    checkBox1.Visible = false;  
    button1.Visible = false;  
    button2.Visible = false;  
    button3.Visible = false;  
    eraseToolStripMenuItem_Click(sender, e);  
}
```

```
private void blackWhiteToolStripMenuItem_Click(object sender, EventArgs e)  
{  
    pictureBox1.Visible = true;  
    button2.Visible = true;  
    button3.Visible = true;  
}
```

```
private void button3_Click(object sender, EventArgs e)  
{  
    for (int i = 0; i < 400; i++)  
        for (int j = 0; j < 400; j++)  
            {  
                Color c = Im1.GetPixel(i, j);  
                int m = (11 * c.R + 16 * c.G + 5 * c.B) / 32;  
                if (m < 128) Im1.SetPixel(i, j, Color.Black); else Im1.SetPixel(i, j, Color.White);  
                // Im3.SetPixel(i, j, c);  
            }  
    pictureBox1.Image = Im1;  
    pictureBox1.Refresh();  
}
```

```
private void erosionToolStripMenuItem_Click(object sender, EventArgs e)
{
    Bitmap Aux = new Bitmap(Im1);

    Im2 = Aux; pictureBox2.Visible = true;

    int[,] b = new int[3, 3];
    b[0, 0] = 0; b[0, 1] = 0; b[0, 2] = 1;
    b[1, 0] = 0; b[1, 1] = 1; b[1, 2] = 1;
    b[2, 0] = 1; b[2, 1] = 1; b[2, 2] = 1;
    for (int i = 1; i < 400-1; i++)
        for (int j = 1; j < 400-1; j++) {

            int Min=255;
            for (int s = 0; s <= 2; s++)
                for (int t = 0; t <= 2; t++)
                {
                    if (b[s, t] > 0)
                    {
                        Color c = Im1.GetPixel(i + s - 1, j + t - 1);
                        int m = (c.R + c.G + c.B) / 3;
                        if (m < Min) Min = m;
                    }
                }
            Im2.SetPixel(i, j, Color.FromArgb(255, Min, Min, Min));
        }
    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
```

```
private void pictureBox2_Click(object sender, EventArgs e)
{
    saveFileDialog1.ShowDialog();
    Im2.Save(saveFileDialog1.FileName);
}
private void dilationToolStripMenuItem_Click(object sender, EventArgs e)
{
    Bitmap Aux = new Bitmap(Im1);
    Im2 = Aux; pictureBox2.Visible = true;

    b[0, 0] = 0; b[0, 1] = 0; b[0, 2] = 1;
    b[1, 0] = 0; b[1, 1] = 1; b[1, 2] = 1;
    b[2, 0] = 1; b[2, 1] = 1; b[2, 2] = 1;
    for (int i = 1; i < 400 - 1; i++)
        for (int j = 1; j < 400 - 1; j++)
            {
                int Max = 0;
                for (int s = 0; s <= 2; s++)
                    for (int t = 0; t <= 2; t++)
                        {
                            if (b[s, t] > 0)
                                {
                                    Color c = Im1.GetPixel(i + s - 1, j + t - 1);
                                    int m = (c.R + c.G + c.B) / 3;
                                    if (m > Max) Max = m;
                                }
                        }
                Im2.SetPixel(i, j, Color.FromArgb(255, Max, Max, Max));
            }
    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
```

```
private void difToolStripMenuItem_Click(object sender, EventArgs e)
{
    //Bitmap Aux = new Bitmap(Im1);          //Im2 = Aux; pictureBox2.Visible = true;
    for (int i = 0; i < 400 ; i++)
        for (int j = 0; j < 400 ; j++)
            {
                Color c1 = Im1.GetPixel(i , j);
                int m1 = (c1.R + c1.G + c1.B) / 3;
                Color c2 = Im2.GetPixel(i , j);
                int m2 = (c2.R + c2.G + c2.B) / 3;
                int m=m1-m2;
                if (m < 0) m = 0; // m = 256 + m;          //int m;          //if (m1 < m2) m = m2-m1; else m = m1-m2;
                Im2.SetPixel(i, j, Color.FromArgb(255, m, m, m));
            }
    pictureBox2.Image = Im2;          pictureBox2.Refresh();
}
```

```
private void rightToolStripMenuItem_Click(object sender, EventArgs e)
{
    Bitmap Aux = new Bitmap(Im1);
    Im2 = Aux; pictureBox2.Visible = true;

    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
```

```
private void colorToolStripMenuItem_Click(object sender, EventArgs e)
{
    pictureBox1.Visible = true;
    //button1.Visible = true;
    button2.Visible = true;
}
```

```
private void erosionToolStripMenuItem1_Click(object sender, EventArgs e)
{
    Bitmap Aux = new Bitmap(Im1);

    Im2 = Aux; pictureBox2.Visible = true;

    int[,] b = new int[3, 3];
    b[0, 0] = 0; b[0, 1] = 1; b[0, 2] = 0;
    b[1, 0] = 1; b[1, 1] = 1; b[1, 2] = 1;
    b[2, 0] = 0; b[2, 1] = 1; b[2, 2] = 0;
    for (int i = 1; i < 400 - 1; i++)
        for (int j = 1; j < 400 - 1; j++)
            {
                int mr = 255; int mg = 255; int mb = 255;
                for (int s = 0; s <= 2; s++)
                    for (int t = 0; t <= 2; t++)
                        {
                            if (b[s, t] > 0)
                                {
                                    Color c = Im1.GetPixel(i + s - 1, j + t - 1);
                                    if (mr > c.R) mr = c.R;
                                    if (mg > c.G) mg = c.G;
                                    if (mb > c.B) mb = c.B;
                                }
                        }
                Im2.SetPixel(i, j, Color.FromArgb(255, mr,mg,mb));
            }
    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
```

```
private void dilationToolStripMenuItem1_Click(object sender, EventArgs e)
{
    Bitmap Aux = new Bitmap(Im1);
    Im2 = Aux; pictureBox2.Visible = true;
    int[,] b = new int[3, 3];
    b[0, 0] = 0; b[0, 1] = 1; b[0, 2] = 0;
    b[1, 0] = 1; b[1, 1] = 1; b[1, 2] = 1;
    b[2, 0] = 0; b[2, 1] = 1; b[2, 2] = 0;
    for (int i = 1; i < 400 - 1; i++)
        for (int j = 1; j < 400 - 1; j++)
        {

            int mr = 0; int mg = 0; int mb = 0;
            for (int s = 0; s <= 2; s++)
                for (int t = 0; t <= 2; t++)
                {
                    if (b[s, t] > 0)
                    {
                        Color c = Im1.GetPixel(i + s - 1, j + t - 1);
                        if (mr < c.R) mr = c.R;
                        if (mg < c.G) mg = c.G;
                        if (mb < c.B) mb = c.B;
                    }
                }
            Im2.SetPixel(i, j, Color.FromArgb(255, mr, mg, mb));
        }
    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
/*
int fs(int x)    {    return (int) (255*Math.Sin(x/255.0*3.1415/2.0));    }
*/
```

```
private void differenceToolStripMenuItem_Click(object sender, EventArgs e)
{
    for (int i = 0; i < 400; i++)
        for (int j = 0; j < 400; j++)
            {
                Color c1 = Im1.GetPixel(i, j);
                Color c2 = Im2.GetPixel(i, j);
                int mr = c1.R - c2.R; if (mr < 0) mr = 0;
                int mg = c1.G - c2.G; if (mg < 0) mg = 0;
                int mb = c1.B - c2.B; if (mb < 0) mb = 0;
                // Im2.SetPixel(i, j, Color.FromArgb(255, fs(mr), fs(mg), fs(mb)));
                Im2.SetPixel(i, j, Color.FromArgb(255, mr, mg, mb));
            }
    pictureBox2.Image = Im2;
    pictureBox2.Refresh();
}
```

```
private void eraseToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Drawing.Pen myPen;
    myPen = new System.Drawing.Pen(System.Drawing.Color.Chocolate);
    System.Drawing.Graphics formGraphics = this.CreateGraphics();
    myPen.Color = System.Drawing.Color.White;

    for (int i = 0; i < 1500; i++)
    {
        formGraphics.DrawLine(myPen, i, 0, i, 900);
    }
    myPen.Dispose();
    formGraphics.Dispose();
}
```

```
private void drawToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Drawing.Pen myPen;
    myPen = new System.Drawing.Pen(System.Drawing.Color.Chocolate);
    System.Drawing.Graphics formGraphics = this.CreateGraphics();
    System.IO.StreamReader Fc = new System.IO.StreamReader("Piramida.Txt");
    String Line = Fc.ReadLine();
    String[] Split = Line.Split(new Char[] { ' ', ',', '\t' });
    int n = Convert.ToInt32(Split[0]);
    varf[] V = new varf[n + 1];
    for (int i = 1; i <= n; i++)
    {
        Line = Fc.ReadLine();
        Split = Line.Split(new Char[] { ' ', ',', '\t' });
        int X = Convert.ToInt32(Split[0]);
        int Z = Convert.ToInt32(Split[1]);
        int Y = Convert.ToInt32(Split[2]) - 100; // y <--->z
        V[i] = new varf(X, Y, Z);           // V V V !!!
    }
    Line = Fc.ReadLine();
    Split = Line.Split(new Char[] { ' ', ',', '\t' });
    int m = Convert.ToInt32(Split[0]);
    muchie[] M = new muchie[m + 1];
    for (int j = 1; j <= m; j++)
    {
        Line = Fc.ReadLine();
        Split = Line.Split(new Char[] { ' ', ',', '\t' });
        M[j] = new muchie();
        M[j].st = Convert.ToInt32(Split[0]);
        M[j].dr = Convert.ToInt32(Split[1]);
    }
}
```



```
Fc.Close();
```

```
for (int i = 1; i <= n; i++)  
{  
    RotOy(V[i], 0.25);  
}
```

```
DefPr(500, 100); // Persp.(d,q);
```

```
a = b = Px(V[1]); c = d = Py(V[1]);
```

```
for (int i = 2; i <= n; i++)  
{  
    double px = Px(V[i]);  
    if (px < a) a = px; else if (px > b) b = px;  
    double py = Py(V[i]);  
    if (py < c) c = py; else if (py > d) d = py;  
}
```

```
Window(a, d, b, c);
```

```
ViewPort(200, 100, 700, 700);
```

```
for (int j = 1; j <= m; j++)  
{  
    myPen.Color = System.Drawing.Color.Blue;  
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st])), v(Py(V[M[j].st])), u(Px(V[M[j].dr])), v(Py(V[M[j].dr))));  
}
```

```
for (int i = 1; i <= n; i++)  
{  
    RotOy(V[i], -0.05);  
}
```

```
ViewPort(700, 100, 1200, 700);
for (int j = 1; j <= m; j++)
{
    myPen.Color = System.Drawing.Color.Blue;
    formGraphics.DrawLine(myPen, u(Px(V[M[j].st])), v(Py(V[M[j].st])), u(Px(V[M[j].dr])), v(Py(V[M[j].dr]]));
}

// a = b = Px(V[100]); // Debug

// myPen.Color = System.Drawing.Color.Blue;
// formGraphics.DrawLine(myPen, 400, 500, 900, 400);

myPen.Dispose();
formGraphics.Dispose();
}

private void hideToolStripMenuItem2_Click(object sender, EventArgs e)
{
    pictureBox3.Visible = false;
    eraseToolStripMenuItem_Click(sender, e);
}

int St(int j, int i)
{
    if (j < 500)
    {
        j += 50; i += 100;
        double x = (i - 225) / 100.0;
        double y = (j - 400) / 100.0;
        if (x * x + y * y > 1) return 0;
    }
}
```

```
else
{
    x *= 1.0 * 3.1415272;
    y *= 1.0 * 3.1415272;
    return (int)((Math.Cos(x * x + y * y)) * 3 + 3);
}
}
else
{
    j -= 250; i -= 100;
    double x = (i - 225) / 100.0;
    double y = (j - 400) / 100.0;
    if (x * x + y * y > 1) return 0;
    else
    {
        x *= 1.0 * 3.1415272;
        y *= 1.0 * 3.1415272;
        return (int)((Math.Sin(x * x + y * y)) * 3 + 3);
    }
}
}
int Pa(int j, int i)
{
    double x = (i - 225) / 100.0;
    double y = (j - 400) / 100.0;
    if (Math.Abs(x) < 0.25 && Math.Abs(y) < 0.25) return 5;
    if (Math.Abs(x) < 0.5 && Math.Abs(y) < 0.5) return 3;
    else
        if (Math.Abs(x) < 0.75 && Math.Abs(y) < 0.75) return 1;
        else
            return 0;
}
```

```
private void randomDotToolStripMenuItem_Click(object sender, EventArgs e)
{
    pictureBox3.Visible = true;

    for (int i = 0; i < 450; i++)
        for (int j = 0; j < 1200; j++)
            {
                int d = St(j, i); //St(j, i);
                Im3.SetPixel(j, i, Color.FromArgb(255, 255 - d * 10, 255 - d * 10, 255 - d * 10));
            }
    pictureBox3.Image = Im3;
    pictureBox3.Refresh();
}

private void pictureBox3_Click(object sender, EventArgs e)
{
    saveFileDialog1.ShowDialog();
    Im3.Save(saveFileDialog1.FileName);
}

private void oneImageToolStripMenuItem_Click(object sender, EventArgs e)
{
    Random ra = new Random();
    pictureBox3.Visible = true;
    for (int i = 0; i < 450; i++)
        for (int j = 0; j < 200; j++)
            {
                int r = ra.Next(255); int g = ra.Next(255); int b = ra.Next(255);
                for (int k = 0; k < 6; k++)
                    {
                        Im3.SetPixel(j + k * 200, i, Color.FromArgb(255, r, g, b));
                    }
            }
}
```

```
int[] dx = new int[1201];

for (int i = 0; i < 450; i++)
{
    for (int j = 0; j <= 1200; j++) dx[j] = 0;
    for (int j = 100; j < 1000; j++)
    {
        int dX=St(j, i) + dx[j];
        if (dX > 0)
        {
            Im3.SetPixel(j, i, Im3.GetPixel(j+dX, i));
            dx[j + 200] = dX;
        }
    }
}

/*

for (int i = 0; i < 450; i++)
{
    int k = 0;
    for (int j = 100; j < 700; j++)
    {
        int st = Pa(j, i)/5;
        if (st > 0)
        {
            //for (int y = 0; y < st; y++)
            {
                for (int x = j; x < 700 - st; x++)
                {
                    Color c = Im3.GetPixel(x + st, i);
                    Im3.SetPixel(x, i, c);
                }
            }
        }
    }
}
```

```
    }
    Im3.SetPixel(700-st, i, Im3.GetPixel(k, i));
    k=k+st; k %= 100;
  }
}
pictureBox3.Image = Im3;
pictureBox3.Refresh();
}

    */
/*
for (int i = 0; i < 450; i++)
{
    int Aij = 0; int k = 0;
    for (int j = 100; k < 700; j++)
    {
        //int aij=St(j,i)+Aij;

        if (Pa(j,i)+Aij> 0) Aij =Aij+Pa(j,i) - 1;
        else
            Im3.SetPixel(100+k++, i, Im3.GetPixel(j % 100, i));
        // Aij = dj;
    }
}

*/
/*
```

```
* Dacă  $A_{ij} > 0$  atunci șterge punctul  $C_{i,j}$ ;  $A_{i,j+1} := A_{i,j+1} + A_{i,j} - 1$   
    altfel pune punctul  $C_{i,j}$  în culoarea  $(j-1) \bmod d+1$ 
```

```
for (int i = 0; i < 450; i++)  
    for (int j = 0; j < 100; j++)  
    {  
        int r = ra.Next(255); int g = ra.Next(255); int b = ra.Next(255);  
        for (int k = 0; k < 8; k++)  
        {  
            Im3.SetPixel(j + k * 100 - St(j + k * 100,i), i, Color.FromArgb(255, r, g, b));  
            //Im3.SetPixel(j + k * 100, i, Color.FromArgb(255, St(j + k * 100, i)*10, 255-St(j + k * 100, i) * 10, 255-St(j + k * 100, i)*20));  
        }  
    }  
*/  
pictureBox3.Image = Im3;  
pictureBox3.Refresh();  
}
```

```
private void onScreenToolStripMenuItem_Click(object sender, EventArgs e)  
{  
    openFileDialog1.ShowDialog();  
    loadedImage = Image.FromFile(openFileDialog1.FileName);  
    Im3 = new Bitmap(loadedImage);  
    pictureBox3.Image = Im3;  
    pictureBox3.Refresh();  
    pictureBox3.Visible = true;  
  
    // Simetrie |100  
  
    /*  
    for (int i = 0; i < 450; i++)  
        for (int j = 0; j < 100; j++)
```

```
    Im3.SetPixel(200-j, i, Im3.GetPixel(j, i));
*/
/*
for (int i = 0; i < 450; i++)
    for (int j = 0; j < 200; j++)
    {
        Color c = Im3.GetPixel(j, i);
        int r = c.R; int g = c.G; int b = c.B;
        for (int k = 1; k < 6; k++)
        {
            Im3.SetPixel(j + k * 200, i, Color.FromArgb(255, r+200, g, b));
        }
    }
*/
/*
Random ra = new Random();
pictureBox3.Visible = true;

for (int i = 0; i < 450; i++)
    for (int j = 0; j < 200; j++)
    {
        int r = ra.Next(255); int g = ra.Next(255); int b = ra.Next(255);
        for (int k = 0; k < 1; k++)
        {
            Im3.SetPixel(j + k * 200, i, Color.FromArgb(255, r, g, b));
        }
    }
*/
```



```
int[] dx = new int[1201];

Color Cf = Im3.GetPixel(1000, 10);

for (int i = 0; i < 450; i++)
{
    for (int j = 0; j < 200; j++)
    {
        Color c = Im3.GetPixel(j, i);
        int r = c.R; int g = c.G; int b = c.B;
        for (int k = 1; k < 1; k++)
        {
            Im3.SetPixel(j + k * 200, i, Color.FromArgb(255, r, g, b));
        }
    }
    if (i < 450 - 1)
    {
        //for (int j = 0; j <= 1200; j++) dx[j] = 0;
        for (int j = 200; j < 1200; j++)
        {
            int St = 0; if (Im3.GetPixel(j, i + 1) == Cf) St = 0; else St = 1 + 2*(Im3.GetPixel(j, i + 1).R) / 64;
            int dX = St;
            // if (j > 700) if (dX>0) dX = -9+St;
            {
                Im3.SetPixel(j, i, Im3.GetPixel(j -200 + dX, i));
                //for (int k=0; k<=dX; k++) dx[j + 200 +k] = dX;
            }
            // if (Im3.GetPixel(j, i+1) == Cf) Im3.SetPixel(j, i, Color.Blue); else Im3.SetPixel(j, i, Color.Red);
        }
    }
}
```

```
/*
 * ===== @
int[] dx = new int[1201];
Color Cf = Im3.GetPixel(1000, 10);
for (int i = 0; i < 450; i++)
{
    for (int j = 0; j < 200; j++)
    {
        Color c = Im3.GetPixel(j, i);
        int r = c.R; int g = c.G; int b = c.B;
        for (int k = 1; k < 6; k++)
        {
            Im3.SetPixel(j + k * 200, i, Color.FromArgb(255, r, g, b));
        }
    }
    if (i < 450 - 1)
    {
        for (int j = 0; j <= 1200; j++) dx[j] = 0;
        for (int j = 200; j < 1000-50; j++)
        {
            int St = 0; if (Im3.GetPixel(j, i + 1) == Cf) St = 0; else St = 1 + (Im3.GetPixel(j, i + 1).R) / 64;
            int dX = St+dx[j];
            //if (dX > 0)
            {
                Im3.SetPixel(j, i, Im3.GetPixel(j + dX, i));
                for (int k=0; k<=dX; k++) dx[j + 200 +k] = dX;
            }
            // if (Im3.GetPixel(j, i+1) == Cf) Im3.SetPixel(j, i, Color.Blue); else Im3.SetPixel(j, i, Color.Red);
        }
    }
}
}
```

```
* ===== @
*/

//==

/*
int[] dx = new int[1201];

//Color Cf = Im3.GetPixel(1000, 10);
for (int i = 0; i < 450; i++)
{
    for (int j = 0; j <= 1200; j++) dx[j] = 0;
    for (int j = 210; j < 1000; j++)
    {
        int St;
        if (Im3.GetPixel(j, i) == Cf) St = 0; else St = 1;
        int dX = St; // +dx[j];
        //if (dX > 0)
        {
            Im3.SetPixel(j, i, Im3.GetPixel(j-200-dX, i));
            dx[j + 200] = dX;
        }
    }
}

*/
pictureBox3.Image = Im3;
pictureBox3.Refresh();
}
```

```
private void invColorsToolStripMenuItem_Click(object sender, EventArgs e)
{
    openFileDialog1.ShowDialog();
    loadedImage = Image.FromFile(openFileDialog1.FileName);
    Im3 = new Bitmap(loadedImage);
    pictureBox3.Image = Im3;
    pictureBox3.Refresh();
    pictureBox3.Visible = true;

    // Swap Red <---> Green

    for (int i = 0; i < 450; i++)
        for (int j = 0; j < 1200; j++)
        {
            Color c = Im3.GetPixel(j, i);
            Im3.SetPixel(j, i, Color.FromArgb(255, c.G, c.R, c.B));
        }
    pictureBox3.Image = Im3;
    pictureBox3.Refresh();
}
}
```