

AD SOYAD CENGİZHAN TOPÇU**NUMARA** 2017010225048**GÖRÜNTÜ İŞLEME ÖDEV-2****KARABÜK
ÜNİVERSİTESİ
MÜHENDİSLİK
FAKÜLTESİ
MEKATRONİK
MÜHENDİSLİĞİ**

Tasma Ölçeklendirme Döndürme Egme Kaydırarak Döndürme

X Y

ÖDEV-2

8)
Ölçekleme komutuna tıklandığında resmin köşelerinde küçük kutucuklar oluşsun. Mouse ile bu noktadan basarak sürüklenip bırakıldığında, bırakılan nokta ile köşe arasında resim küçülmüş olsun. Bu esnada orantısı bozulmasın.

Resmi Ölçeklendir

12)
Döndürme komutunu kullanın. Oluşan Alias boğluklarını yok etmek için resim oluştuktan sonra siyah noktaları etrafındaki pikselleri okutun ve renk olarak etrafındaki piksellerin ortalamasını alacak şekilde programlayın. Döndürme işlemini yana ve aşağı kaydırarak programlayın. Resmin köşesinden eğerek döndürme yapacak.

Resmi Döndür

13)
Kırma işlemini yapan kodları yazınız. Mouse takip eden çizgi çizdirerek yapın.

X1 X2
Y1 Y2

Resmi Kırp

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

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

        //Resim Yükle
        private void BtnAc_Click(object sender, EventArgs e)
        {
            try
            {

```

```

        openFileDialog1.DefaultExt = ".jpg";
        openFileDialog1.Filter = "Image
Files(*.BMP;*.JPG;*.GIF;*.PNG)|*.BMP;*.JPG;*.GIF;*.PNG|All files (*.*)|*.*";
        openFileDialog1.ShowDialog();
        String ResminYolu = openFileDialog1.FileName;
        pictureBox1.Image = Image.FromFile(ResminYolu);
    }
    catch
    {
        MessageBox.Show("HATA! Resim yüklenmedi");
    }
}

//Resim Kaydet
private void BtnKaydet_Click(object sender, EventArgs e)
{
    SaveFileDialog saveFileDialog1 = new SaveFileDialog();
    saveFileDialog1.Filter = "Jpeg Resmi|*.jpg|Bitmap Resmi|*.bmp|Gif
Resmi|*.gif";
    saveFileDialog1.Title = "Resmi Kaydet";
    saveFileDialog1.ShowDialog();

    if (saveFileDialog1.FileName != "") //Dosya adı boş değilse kaydedecek.
    {
        // FileStream nesnesi ile kayıtlı gerçekleştirecek.
        FileStream DosyaAkisi = (FileStream)saveFileDialog1.OpenFile();

        switch (saveFileDialog1.FilterIndex)
        {
            case 1:
                pictureBox2.Image.Save(DosyaAkisi,
System.Drawing.Imaging.ImageFormat.Jpeg);
                break;

            case 2:
                pictureBox2.Image.Save(DosyaAkisi,
System.Drawing.Imaging.ImageFormat.Bmp);
                break;

            case 3:
                pictureBox2.Image.Save(DosyaAkisi,
System.Drawing.Imaging.ImageFormat.Gif);
                break;
        }
        DosyaAkisi.Close();
    }
}

//Resim Boyutu Ayarlama
private void btnBoyutAyarla_Click(object sender, EventArgs e)
{
    pictureBox1.SizeMode = PictureBoxSizeMode.Zoom;
    pictureBox2.SizeMode = PictureBoxSizeMode.Zoom;
}

//Resmi Sağa Aktarma
private void btnSagaAktar_Click(object sender, EventArgs e)
{
    //pictureBox2.Image = pictureBox1.Image;

    Color OkunanRenk, DonusenRenk;
    int R = 0, G = 0, B = 0;

    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width; //GirisResmi global tanımlandı.

```

İçerisine görüntü yüklendi.

```
int ResimYuksekligi = GirişResmi.Height;
CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi); //Cikis
resmini oluşturuyor. Boyutları giriş resmi ile aynı olur. Tanımlaması globalde
yapıldı.
```

```
int i = 0, j = 0; //Çıkış resminin x ve y si olacak.
for (int x = 0; x < ResimGenisligi; x++)
{
    for (int y = 0; y < ResimYuksekligi; y++)
    {
        OkunanRenk = GirişResmi.GetPixel(x, y);

        R = OkunanRenk.R;
        G = OkunanRenk.G;
        B = OkunanRenk.B;

        //if (x>100 && x<200)
        DonusenRenk = Color.FromArgb(R, G, B);
        CikisResmi.SetPixel(x, y, DonusenRenk);

    }
}
pictureBox2.Image = CikisResmi;
}
```

//Resmi Sola Aktarma

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

```
    Color OkunanRenk, DonusenRenk;
    int R = 0, G = 0, B = 0;
```

```
    Bitmap GirişResmi, CikisResmi;
    GirişResmi = new Bitmap(pictureBox2.Image);
```

İçerisine görüntü yüklendi.

```
int ResimGenisligi = GirişResmi.Width; //GirişResmi global tanımlandı.
int ResimYuksekligi = GirişResmi.Height;
CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi); //Cikis
resmini oluşturuyor. Boyutları giriş resmi ile aynı olur. Tanımlaması globalde
yapıldı.
```

```
int i = 0, j = 0; //Çıkış resminin x ve y si olacak.
for (int x = 0; x < ResimGenisligi; x++)
{
    for (int y = 0; y < ResimYuksekligi; y++)
    {
        OkunanRenk = GirişResmi.GetPixel(x, y);

        R = OkunanRenk.R;
        G = OkunanRenk.G;
        B = OkunanRenk.B;

        DonusenRenk = Color.FromArgb(R, G, B);
        CikisResmi.SetPixel(x, y, DonusenRenk);

    }
}
pictureBox1.Image = CikisResmi;
}
```

//Tasima

```
private void btnTasima_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
```

```

        Bitmap GirisResmi, CikisResmi;
        GirisResmi = new Bitmap(pictureBox1.Image);
        int ResimGenisligi = GirisResmi.Width;
        int ResimYuksekligi = GirisResmi.Height;
        CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

        double x2 = 0, y2 = 0;

        //Taşıma mesafelerini atıyor.
        int Tx = Convert.ToInt32(textBox1.Text);
        int Ty = Convert.ToInt32(textBox2.Text);

        for (int x1 = 0; x1 < (ResimGenisligi); x1++)
        {
            for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
            {
                OkunanRenk = GirisResmi.GetPixel(x1, y1);

                x2 = x1 + Tx;
                y2 = y1 + Ty;

                if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
                    CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
            }
        }
        pictureBox2.Image = CikisResmi;
    }

    //Aynalama
    private void btnAynalama_Click(object sender, EventArgs e)
    {
        Color OkunanRenk;
        Bitmap GirisResmi, CikisResmi;

        GirisResmi = new Bitmap(pictureBox1.Image);
        int ResimGenisligi = GirisResmi.Width;
        int ResimYuksekligi = GirisResmi.Height;

        CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

        double x2 = 0, y2 = 0;

        //Taşıma mesafelerini atıyor.
        int x0 = ResimGenisligi/2;
        int y0 = ResimYuksekligi/2;

        for (int x1 = 0; x1 < (ResimGenisligi); x1++)
        {
            for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
            {
                OkunanRenk = GirisResmi.GetPixel(x1, y1);

                x2 = x1 + 2*(x0-x1);
                y2 = y1;

                if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
                    CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
            }
        }
        pictureBox2.Image = CikisResmi;
    }

    //Olceklendirme
    private void btnOlceklendirme_Click(object sender, EventArgs e)

```

```

{
    Color OkunanRenk, DonusenRenk;
    Bitmap GirisResmi, CikisResmi;

    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int x2 = 0, y2 = 0; //Çıkış resminin x ve y si olacak.
    //int KucultmeKatsayisi = 2;
    int KucultmeKatsayisi = Convert.ToInt32(textBox3.Text);

    for (int x1 = 0; x1 < ResimGenisligi; x1 = x1 + KucultmeKatsayisi)
    {
        y2 = 0;
        for (int y1 = 0; y1 < ResimYuksekligi; y1 = y1 + KucultmeKatsayisi)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

            CikisResmi.SetPixel(x2, y2, OkunanRenk);
            y2++;
        }
        x2++;
    }
    pictureBox2.Image = CikisResmi;
}

//Dondurma
private void btnDondurma_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int Aci = Convert.ToInt16(textBox4.Text);
    double RadyanAci = Aci * 2 * Math.PI / 360;

    double x2 = 0, y2 = 0;

    //Resim merkezini buluyor. Resim merkezi etrafında döndürecek.
    int x0 = ResimGenisligi / 2;
    int y0 = ResimYuksekligi / 2;

    for (int x1 = 0; x1 < (ResimGenisligi); x1++)
    {
        for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

            //Döndürme Formülleri
            x2 = Math.Cos(RadyanAci) * (x1 - x0) - Math.Sin(RadyanAci) *
(y1 - y0) + x0;
            y2 = Math.Sin(RadyanAci) * (x1 - x0) + Math.Cos(RadyanAci) *
(y1 - y0) + y0;

            if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
                CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
        }
    }
}

```

```

    }
    pictureBox2.Image = CikisResmi;
}

//Egme
private void btnEgme_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    //Taşıma mesafelerini atıyor.
    double EgmeKatsayisi = Convert.ToDouble(textBox5.Text);
    double x2 = 0, y2 = 0;

    for (int x1 = 0; x1 < (ResimGenisligi); x1++)
    {
        for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

            // +X eksenini yönünde
            x2 = x1 + EgmeKatsayisi * y1;
            y2 = y1;

            // -X eksenini yönünde
            //x2 = x1 - EgmeKatsayisi * y1;
            //y2 = y1;

            // +Y eksenini yönünde
            //x2 = x1;
            //y2 = EgmeKatsayisi * x1 + y1;

            // -Y eksenini yönünde
            //x2 = x1;
            //y2 = -EgmeKatsayisi * x1 + y1;

            if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
                CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
        }
    }
    pictureBox2.Image = CikisResmi;
}

//Kayıtararak Dondurma
private void btnKaydirarakDondurma_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int Aci = Convert.ToInt16(textBox6.Text);
    double RadyanAci = Aci * 2 * Math.PI / 360;

    double x2 = 0, y2 = 0;

```

```

//Resim merkezini buluyor. Resim merkezi etrafında döndürecek.
int x0 = ResimGenisligi / 2;
int y0 = ResimYuksekligi / 2;

for (int x1 = 0; x1 < (ResimGenisligi); x1++)
{
    for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
    {
        OkunanRenk = GirisResmi.GetPixel(x1, y1);

        //Aliaslı Döndürme -Sağa Kaydırma
        x2 = (x1 - x0) - Math.Tan(RadyanAci / 2) * (y1 - y0) + x0;
        y2 = (y1 - y0) + y0;

        x2 = Convert.ToInt16(x2);
        y2 = Convert.ToInt16(y2);

        //Aliaslı Döndürme -Aşağı kaydırma
        x2 = (x2 - x0) + x0;
        y2 = Math.Sin(RadyanAci) * (x2 - x0) + (y2 - y0) + y0;

        x2 = Convert.ToInt16(x2);
        y2 = Convert.ToInt16(y2);

        //Aliaslı Döndürme -Sağa Kaydırma
        x2 = (x2 - x0) - Math.Tan(RadyanAci / 2) * (y2 - y0) + x0;
        y2 = (y2 - y0) + y0;

        x2 = Convert.ToInt16(x2);
        y2 = Convert.ToInt16(y2);

        if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
            CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
    }
}
pictureBox2.Image = CikisResmi;
}

//Kırpma
private void btnKırpma_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int a = 50;
    int b = 50;
    int c = 150;
    int d = 150;

    for (int x1 = 0; x1 < (ResimGenisligi); x1++)
    {
        for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

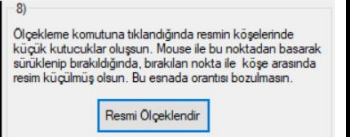
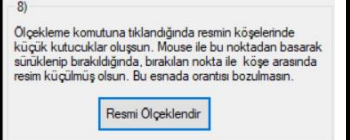
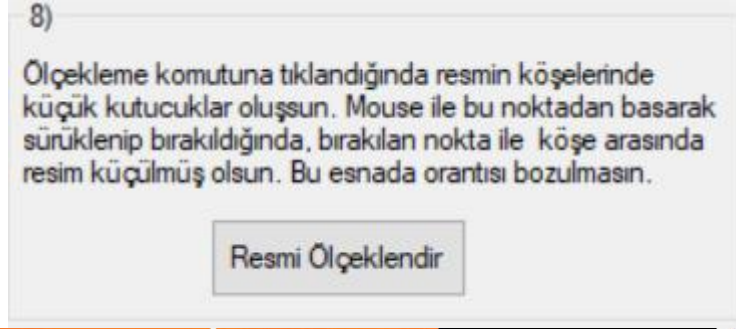
            if (x1 > a && x1 < c && y1 > b && y1 < d)
                CikisResmi.SetPixel((int)x1, (int)y1, OkunanRenk);
        }
    }
}

```

```

    }
    pictureBox2.Image = CikisResmi;
}

```



```

//Ödev-2 8)
private void btnOlceklendir_Click(object sender, EventArgs e)
{
    Color OkunanRenk, DonusenRenk;
    Bitmap GirisResmi, CikisResmi;

    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int x2 = 0, y2 = 0; //Çıkış resminin x ve y si olacak.
    int KucultmeKatsayisi = 3;

    for (int x1 = 0; x1 < ResimGenisligi; x1 = x1 + KucultmeKatsayisi)
    {
        y2 = 0;
        for (int y1 = 0; y1 < ResimYuksekligi; y1 = y1 + KucultmeKatsayisi)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

            CikisResmi.SetPixel(x2, y2, OkunanRenk);
            y2++;
        }
        x2++;
    }
}

```



```

        pictureBox2.Image = CikisResmi;
    }

```

12)

Döndürme komutunu kullanın. Oluşan Alias boşluklarını yok etmek için resim oluştuktan sonra siyah noktaların etrafındaki pikselleri okutun ve renk olarak etrafındaki piksellerin ortalamasını alacak şekilde programlayın. Döndürme işlemini yana ve aşağı kaydırarak programlayın. Resmin köşesinden eğerek döndürme yapacak.

Resmi Döndür



12)

Döndürme komutunu kullanın. Oluşan Alias boşluklarını yok etmek için resim oluştuktan sonra siyah noktaların etrafındaki pikselleri okutun ve renk olarak etrafındaki piksellerin ortalamasını alacak şekilde programlayın. Döndürme işlemini yana ve aşağı kaydırarak programlayın. Resmin köşesinden eğerek döndürme yapacak.

Resmi Döndür



12)

Döndürme komutunu kullanın. Oluşan Alias boşluklarını yok etmek için resim oluştuktan sonra siyah noktaların etrafındaki pikselleri okutun ve renk olarak etrafındaki piksellerin ortalamasını alacak şekilde programlayın. Döndürme işlemini yana ve aşağı kaydırarak programlayın. Resmin köşesinden eğerek döndürme yapacak.

Resmi Döndür

```

//Ödev-2 12)
private void btnDondur_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int Aci = 60;
    double RadyanAci = Aci * 2 * Math.PI / 360;

    double x2 = 0, y2 = 0;

    //Resim merkezini buluyor. Resim merkezi etrafında döndürecek.
    int x0 = ResimGenisligi / 2;
    int y0 = ResimYuksekligi / 2;

    for (int x1 = 0; x1 < (ResimGenisligi); x1++)
    {
        for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

```

```

//Aliaslı Döndürme -Sağa Kaydırma
x2 = (x1 - x0) - Math.Tan(RadyanAci / 2) * (y1 - y0) + x0;
y2 = (y1 - y0) + y0;

x2 = Convert.ToInt16(x2);
y2 = Convert.ToInt16(y2);

//Aliaslı Döndürme -Aşağı kaydırma
x2 = (x2 - x0) + x0;
y2 = Math.Sin(RadyanAci) * (x2 - x0) + (y2 - y0) + y0;

x2 = Convert.ToInt16(x2);
y2 = Convert.ToInt16(y2);

//Aliaslı Döndürme -Sağa Kaydırma
x2 = (x2 - x0) - Math.Tan(RadyanAci / 2) * (y2 - y0) + x0;
y2 = (y2 - y0) + y0;

x2 = Convert.ToInt16(x2);
y2 = Convert.ToInt16(y2);

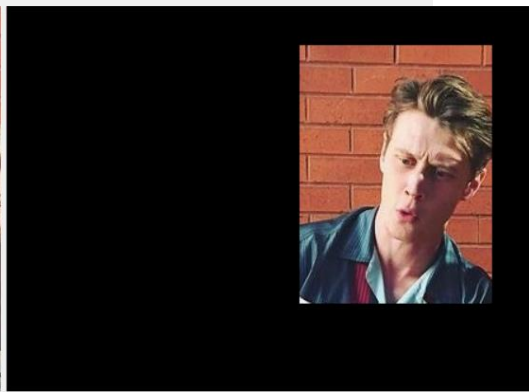
if (x2 > 0 && x2 < ResimGenisligi && y2 > 0 && y2 <
ResimYuksekligi)
    CikisResmi.SetPixel((int)x2, (int)y2, OkunanRenk);
}
}
pictureBox2.Image = CikisResmi;
}

```

13)

Kırpma işlemini yapan kodları yazınız.
Mouse takip eden çizgi çizdirerek yapın.

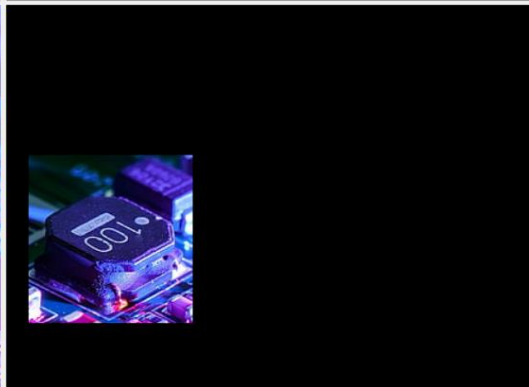
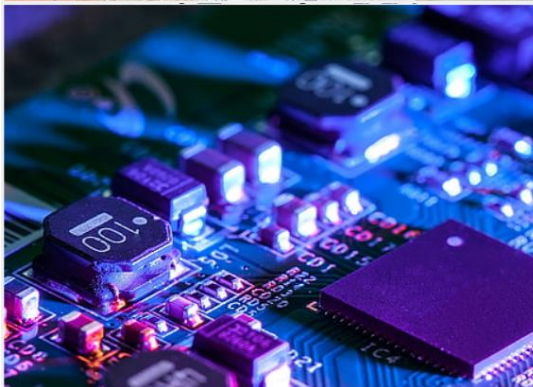
X1	<input type="text"/>	X2	<input type="text"/>
Y1	<input type="text"/>	Y2	<input type="text"/>



13)

Kırpma işlemini yapan kodları yazınız.
Mouse takip eden çizgi çizdirerek yapın.

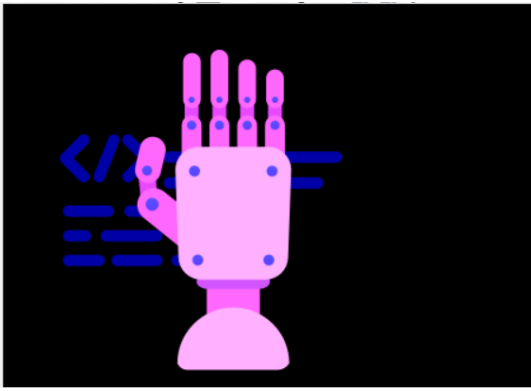
X1	<input type="text" value="247"/>	X2	<input type="text" value="411"/>
Y1	<input type="text" value="32"/>	Y2	<input type="text" value="251"/>



13)

Kırpma işlemini yapan kodları yazınız.
Mouse takip eden çizgi çizdirerek yapın.

X1	<input type="text" value="18"/>	X2	<input type="text" value="158"/>
Y1	<input type="text" value="126"/>	Y2	<input type="text" value="269"/>



13)

Kırma işlemini yapan kodları yazınız.
Mouse takip eden gözü çizdirerek yapın.

X1	139	X2	269
Y1	23	Y2	97

```
//Ödev-2 13)
int tiklanma_sayisi;
private void pictureBox1_MouseUp(object sender, MouseEventArgs e)
{
    tiklanma_sayisi++;

    if (tiklanma_sayisi % 2 == 1)
    {
        textBox_x1.Text = e.X.ToString();
        textBox_y1.Text = e.Y.ToString();
    }

    else if (tiklanma_sayisi % 2 == 0)
    {
        textBox_x2.Text = e.X.ToString();
        textBox_y2.Text = e.Y.ToString();
    }
}

private void btnKırp_Click(object sender, EventArgs e)
{
    Color OkunanRenk;
    Bitmap GirisResmi, CikisResmi;
    GirisResmi = new Bitmap(pictureBox1.Image);

    int ResimGenisligi = GirisResmi.Width;
    int ResimYuksekligi = GirisResmi.Height;

    CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi);

    int a = Convert.ToInt16(textBox_x1.Text);
    int b = Convert.ToInt16(textBox_y1.Text);
    int c = Convert.ToInt16(textBox_x2.Text);
    int d = Convert.ToInt16(textBox_y2.Text);

    for (int x1 = 0; x1 < (ResimGenisligi); x1++)
    {
        for (int y1 = 0; y1 < (ResimYuksekligi); y1++)
        {
            OkunanRenk = GirisResmi.GetPixel(x1, y1);

            if (x1 > a && x1 < c && y1 > b && y1 < d)
                CikisResmi.SetPixel((int)x1, (int)y1, OkunanRenk);
        }
    }
    pictureBox2.Image = CikisResmi;
}
}
```