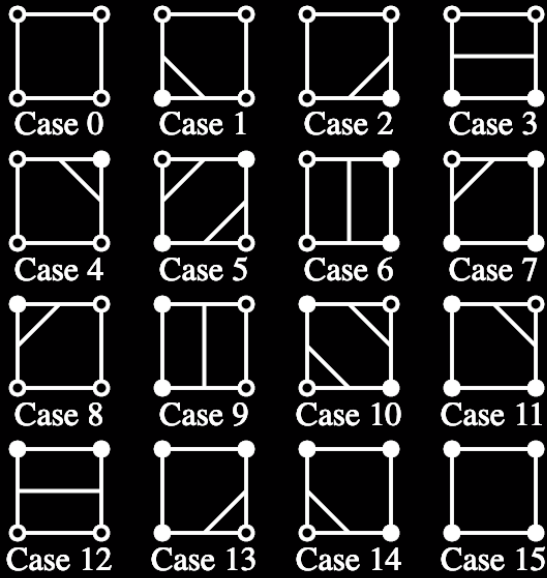
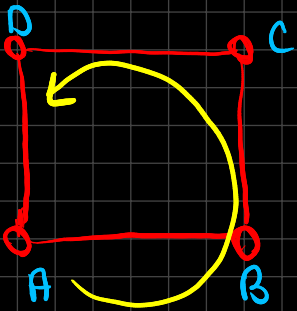


Marching Squares Algoritması



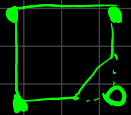
4 tane noktanın 1 veya 0 olma durumuna bağlı olarak soldaki durumlar oluşur.

Peki Nasıl?



Her bir karenin durumunu soldaki şekildaki gibi yorumlarız!

Örnek olarak case 13'e bakalım



D C B A
1 1 0 1

→ içi dolu nokta 1'i, içi boş nokta ise 0'i temsil ediyor.

$$\begin{array}{ccccccc} 1 & & 1 & & 0 & & 1 \\ 2^3 & + & 2^2 & + & 2^1 & + & 2^0 \\ 8 & + & 4 & + & 0 & + & 1 = 13 \end{array}$$

Görüldüğü gibi durumlar Binary olarak toplanıyor.

Şimdi bu durumu şekil çizmek için kullanalım.

ilk olarak bu noktaları oluşturmali ve olusturduğumuz bu noktalara rastgele olarak degerlerini vermeliyiz

int[][] Terrain;

kullanıcıdan alınan bilgiler göre dizinin boyutu belirlenir

```
Terrain = new int[yEksenNoktaSayisi, xEksenNoktaSayisi];
for (int i = 0; i < yEksenNoktaSayisi; i++)
{
    for (int j = 0; j < xEksenNoktaSayisi; j++)
    {
        Random rnd = new Random();
        if (rnd.Next(0, 100) < Oran)
        {
            Terrain[i, j] = 1;
        }
        else
        {
            Terrain[i, j] = 0;
        }
    }
}

listBox1.Items.Add(Terrain[i, j]);
listBox1.Items.Add("-----");
}
```

ilk olarak y eksen

daha sonra x eksenı dısturulur.

Bu bilgiler göre y ve x ekseninde ne kadar nokta oklugu belirlenir

kullanıcıdan alınan Oran sayısı burada görüklüğü gibi noktaların degerini 1 olma oranını olusturuyor!

programımızda kayıtl etmek için listBox'a eleman olarak ekleniyor

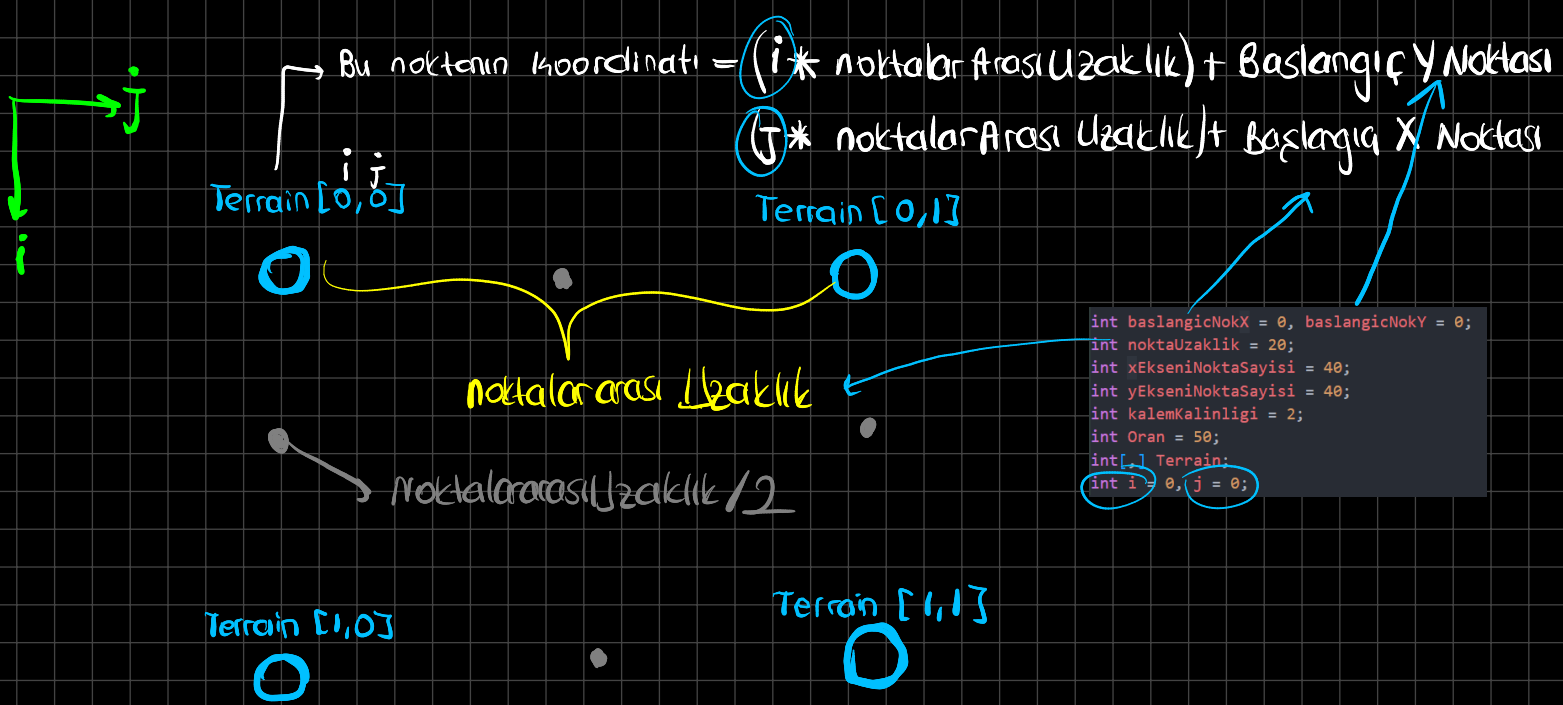
Yukarıdaki koda örnek olarak bir 2x2 boyutlu diziye şekillendirelim!

y eksenNoktası = 3
x eksenNoktası = 2 } olsun.

Terrain[0,0]=1 Terrain[0,1]=0
Terrain[1,0]=0 Terrain[1,1]=1
Terrain[2,0]=1 Terrain[2,1]=1

Temsili olarak böyle bir noktalar dizisi yapılır.

Bu noktaların arasındaki uzaklığı da ayarlayabiliriz. Zaten bu deger girilirken işimize hayli yarayarak.



Marching algoritmasında kullandığımız 16 durum için iki noktanın orta noktasının koordinat bilgileri gereklidir.

→ Ayrıca alınan noktalar arası Uzaklık bilgisi çift sayı olmalıdır ki ikiye bölünebilsin!

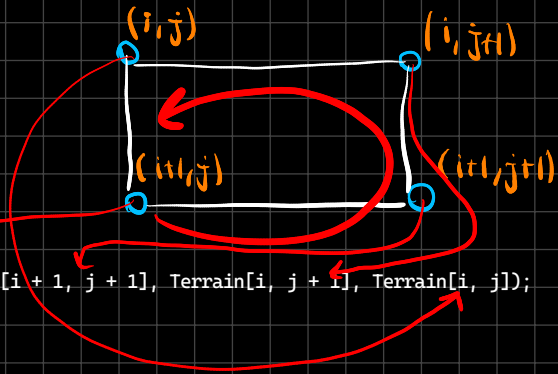
tek mi kontrolü

```
if (Convert.ToInt32(textBox3.Text) % 2 != 0)
{
    noktaUzaklik = Convert.ToInt32(textBox3.Text) + 1;
    textBox3.Text = noktaUzaklik.ToString();
}
else
{
    noktaUzaklik = Convert.ToInt32(textBox3.Text);
}
```

tek + 1 eklenir

```
for (i = 0; i < yEksenNoktaSayisi - 1; i++)
{
    for (j = 0; j < xEksenNoktaSayisi - 1; j++)
    {
        int durum = DurumHesapla(Terrain[i + 1, j], Terrain[i + 1, j + 1], Terrain[i, j + 1], Terrain[i, j]);
        CizimYap(durum);
    }
}

int DurumHesapla(int a, int b, int c, int d)
{
    return a * 1 + b * 2 + c * 4 + d * 8;
}
```



Yukarıdaki kodda 16 durumdan hangisi onu bulmak için metod oluşturuldu.

metodun aldığı parametreler sırası binary toplama için çok önemli

hangı durumunda olduğunu parametre alır

```

CizimVap(int durum)
{
    Pen pen1 = new Pen(kalemRengi, kalemKalınlığı);
    Graphics grakik = CreateGraphics();
    switch (durum)
    {
        case 0:
        {
            break;
        }
        case 1:
        {
            grakik.DrawLine(pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY);
            grakik.DrawLine(pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2);
            break;
        }
        case 2:
        {
            grakik.DrawLine(pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2);
            grakik.DrawLine(pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2);
            grakik.DrawLine(pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY + noktaUzaklık / 2);
            break;
        }
    }
}

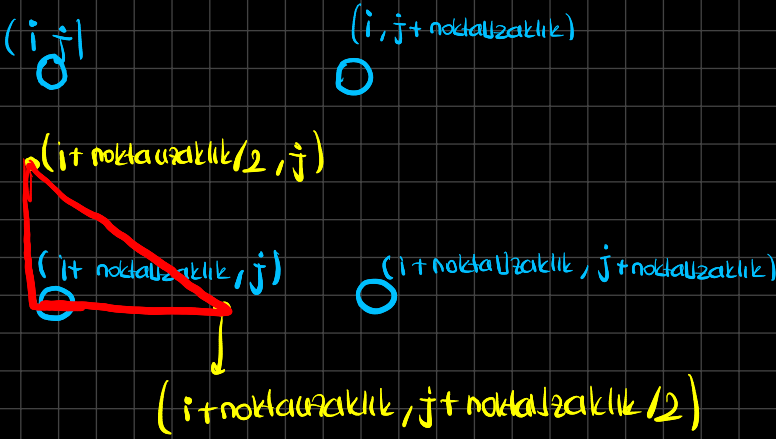
```

→ Çizim yapmak için kalem düstürlüdü.

→ Çizim alanı için Graphics düstürlüdü

System.Drawing
System.Drawing.Printing
kütüphaneleri gaeclıdır.

Case 1 için örnek atalım



Çizmek için Drawline kullanıldı.

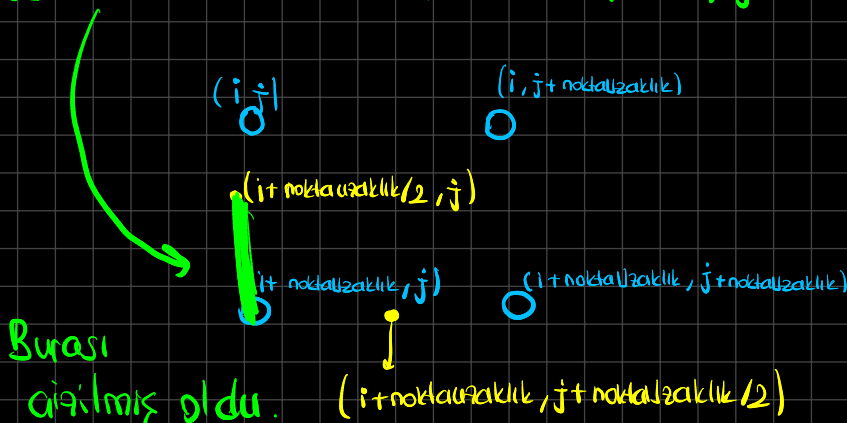
→ Drawline metodu 5 parametre alır.

1. Parametre = kalem
2. Parametre = Çizimin başlangıç noktasının x koordinatı
3. Parametre = Çizimin başlangıç noktasının y koordinatı
4. Parametre = Çizimin bitiş noktasının x koordinatı
5. Parametre = Çizimin bitiş noktasının y koordinatı.

Bu bilgileri biz istediğimiz alırtmada kullanabiliriz.

Case 1 için

• Drawline (pen1, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık / 2, (j * noktaUzaklık) + baslangicNokY, (i * noktaUzaklık) + baslangicNokX + noktaUzaklık, (j * noktaUzaklık) + baslangicNokY)



Başlangıç noktasını da belirleyebiliriz. Ayrıca her adıgı aşımında da başlangıç noktası koordinat bilgisi önemlidir.

Bu aşımı her durum için koordinasyon bilgilerini kullanarak yapıyoruz.

Nokta uzaklık = 10 olsun

Başlangıç X = 5

Başlangıç Y = 10 olsun

i = 0
j = 0



$$X = \text{başlangıç}X + 0 \times \text{nokta uzaklık}$$
$$Y = \text{başlangıç}Y + 0 \times \text{nokta uzaklık}$$

$$X = 5 + 0 \times 10 \quad X = 5$$
$$Y = 10 + 0 \times 10 \quad Y = 10$$

i = 0
j = 1



$$X = \text{başlangıç}X + 1 \times \text{nokta uzaklık}$$
$$Y = \text{başlangıç}Y + 0 \times \text{nokta uzaklık}$$

$$X = 5 + 1 \times 10 = 15$$
$$Y = 10 + 0 \times 10 = 10$$

i = 0
j = 2



$$X = \text{başlangıç}X + 2 \times \text{nokta uzaklık}$$
$$Y = \text{başlangıç}Y + 0 \times \text{nokta uzaklık}$$

$$X = 5 + 2 \times 10 = 25$$
$$Y = 10 + 0 \times 10 = 10$$

Bu şekilde noktaların koordinat sisteminde yerini belirlemiş oluyoruz!

Anlatabildiysem ve yararım dokunduysa NE MUTLU BANA!

İyi Gelişmeler Dilerim.