

**HASAN FERDİ TURGUTLU TEKNOLOJİ FAKÜLTESİ**

**YAZILIM MÜHENDİSLİĞİ**

**ÖDEV – 2**

**Mayın Kontrol Oyunu**

YAZILIM SINAMA

**Hazırlayanlar:**

**Kadir Sefa ÜNAL – 132805008**

**Kadir MUTLU - 132805013**

İçindekiler

[1. Amaç 3](#_Toc466232990)

[2. Girdiler 3](#_Toc466232991)

[3. Mantığı 3](#_Toc466232992)

[4. Program Kodu 3](#_Toc466232993)

[5. Ekran Çıktısı 7](#_Toc466232994)

[6. Test Senaryoları 9](#_Toc466232995)

[a)Başarılı Testler 9](#_Toc466232996)

[b)Başarısız Testler 9](#_Toc466232997)

# 1. Amaç

Mayın Kontrol Oyununda amaç mayın döşeli bir arazide mayın bulucu tankın mayınları bulup imha etmesini sağlayan algoritma geliştirilerek yazılımın değişik durumlarda verdiği sonuçları karşılaştırarak test işlemine tabi tutmak amaçlanmıştır.

# 2. Girdiler

* En: Mayın döşeli arazinin eninin alındığı sayısal girdidir.
* Boy: Mayın döşeli arazinin boyunun alındığı sayısal girdidir.
* Engel: Mayın taraması yaparken kullanıcının koyduğu engellerdir.

# 3. Mantığı

Programın mantığı kullanıcıdan en, boy ve engel durumları alınarak mayın taramasının yapılmasıdır.

Tarayıcı etrafında gidebileceği kareleri kontrol eder. Duvar olmayan yönler dışındaki karelere yönelir. Bu kareler arasında seçim yapmak gerektiğinde öncelik sırasına göre alt, sağ, üst ve sol olmak üzere tarama işlemini gerçekleştirir. Bu işlemi gerçekleştirirken bir karenin üzerinden kaç kere geçtiğinin de etkisi vardır. Son mayını bulana kadar tarama işlemi devam eder.

# 4. Program Kodu

public partial class frmMayinKontrol : Form

{

public frmMayinKontrol()

{

InitializeComponent();

}

Button[,] butonlar;

int en, boy;

private void btnOlustur\_Click(object sender, EventArgs e)

{

if (txtEn.Text != "" && txtBoy.Text != "" &&

SayiMi(txtEn.Text) && SayiMi(txtBoy.Text))

{

en = Convert.ToInt32(txtEn.Text);

boy = Convert.ToInt32(txtBoy.Text);

if (en > 4 && boy > 4 && en < 46 && boy < 21)

{

Temizle();

Olustur(en, boy);

btnBasla.Enabled = true;

}

else

MessageBox.Show("Lütfen ekrana sığmayacak boyut girmeyin.", "UYARI",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

MessageBox.Show("Lütfen sayı girin!" , "UYARI",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

private void btnBasla\_Click(object sender, EventArgs e)

{

if (btnBasla.Text == "Başla")

{

butonlar[1, 1].Tag = Convert.ToInt32(butonlar[1, 1].Tag) + 1;

timer.Start();

btnBasla.Text = "Dur";

txtBoy.Clear();

txtEn.Clear();

btnOlustur.Enabled = false;

}

else

{

timer.Stop();

btnBasla.Text = "Başla";

btnOlustur.Enabled = true;

}

}

private void Btn\_Click(object sender, EventArgs e)

{

Button btn = (sender as Button);

if (btn.BackColor == Color.Red)

{

btn.BackColor = Color.Gray;

btn.Tag = 999;

}

else

{

btn.BackColor = Color.Red;

btn.Tag = 0;

}

}

int i = 1, j = 1;

private void timer\_Tick(object sender, EventArgs e)

{

butonlar[i, j].BackColor = Color.LightGreen;

int yon = YonBul(i, j);

switch (yon)

{

case 0:

i++;

butonlar[i, j].Tag = Convert.ToInt32(butonlar[i, j].Tag) + 1;

butonlar[i, j].BackColor = Color.White;

break;

case 1:

j++;

butonlar[i, j].Tag = Convert.ToInt32(butonlar[i, j].Tag) + 1;

butonlar[i, j].BackColor = Color.White;

break;

case 2:

i--;

butonlar[i, j].Tag = Convert.ToInt32(butonlar[i, j].Tag) + 1;

butonlar[i, j].BackColor = Color.White;

break;

case 3:

j--;

butonlar[i, j].Tag = Convert.ToInt32(butonlar[i, j].Tag) + 1;

butonlar[i, j].BackColor = Color.White;

break;

default:

timer.Stop();

btnBasla.Text = "Başla";

btnOlustur.Enabled = true;

MessageBox.Show("Yon hatası!");

break;

}

BittiMi();

}

//0 = alt

//1 = sag

//2 = ust

//3 = sol

private int YonBul(int x, int y)

{

bool[] yon = new bool[4]{ false, false, false, false };

int sol = Convert.ToInt32(butonlar[x, y - 1].Tag);

int ust = Convert.ToInt32(butonlar[x - 1, y].Tag);

int alt = Convert.ToInt32(butonlar[x + 1, y].Tag);

int sag = Convert.ToInt32(butonlar[x, y + 1].Tag);

if (alt != 999)

yon[0] = true;

if (sag != 999)

yon[1] = true;

if (ust != 999)

yon[2] = true;

if (sol != 999)

yon[3] = true;

if (yon[0] && alt <= ust && alt <= sol && alt <= sag)

return 0;

else if (yon[1] && sag <= sol && sag <= ust && sag <= alt)

return 1;

else if (yon[2] && ust <= alt && ust <= sag && ust <= sol)

return 2;

else if (yon[3] && sol <= sag && sol <= ust && sol <= alt)

return 3;

else

return -1;

}

private void BittiMi()

{

bool durum = true;

foreach (Button btn in butonlar)

{

if (Convert.ToInt32(btn.Tag) == 0)

{

durum = false;

break;

}

}

if (durum)

{

timer.Stop();

btnBasla.Text = "Başla";

btnBasla.Enabled = false;

btnOlustur.Enabled = true;

MessageBox.Show("Tarama bitti!");

}

}

private void Temizle()

{

i = 1;

j = 1;

if (butonlar != null)

{

foreach (Button btn in butonlar)

{

Controls.Remove(btn);

}

}

}

private void Olustur(int en, int boy)

{

butonlar = new Button[boy, en];

Button btn;

for (int i = 0; i < boy; i++)

{

for (int j = 0; j < en; j++)

{

btn = new Button();

btn.Name = "btn" + i + "\_" + j;

btn.Width = 30;

btn.Height = 30;

btn.Top = 70 + (i \* 30);

btn.Left = 13 + (j \* 30);

btn.FlatStyle = FlatStyle.Flat;

if (i == 0 || j == 0 || i == (boy - 1) || j == (en - 1))

{

btn.BackColor = Color.Gray;

btn.Tag = 999;

}

else

{

if (i != 1 || j != 1)

btn.Click += Btn\_Click;

btn.BackColor = Color.Red;

btn.Tag = 0;

}

butonlar[i, j] = btn;

}

}

foreach (var item in butonlar)

{

Controls.Add(item);

}

}

public bool SayiMi(string ifade)

{

foreach (char chr in ifade)

{

if (!Char.IsNumber(chr)) return false;

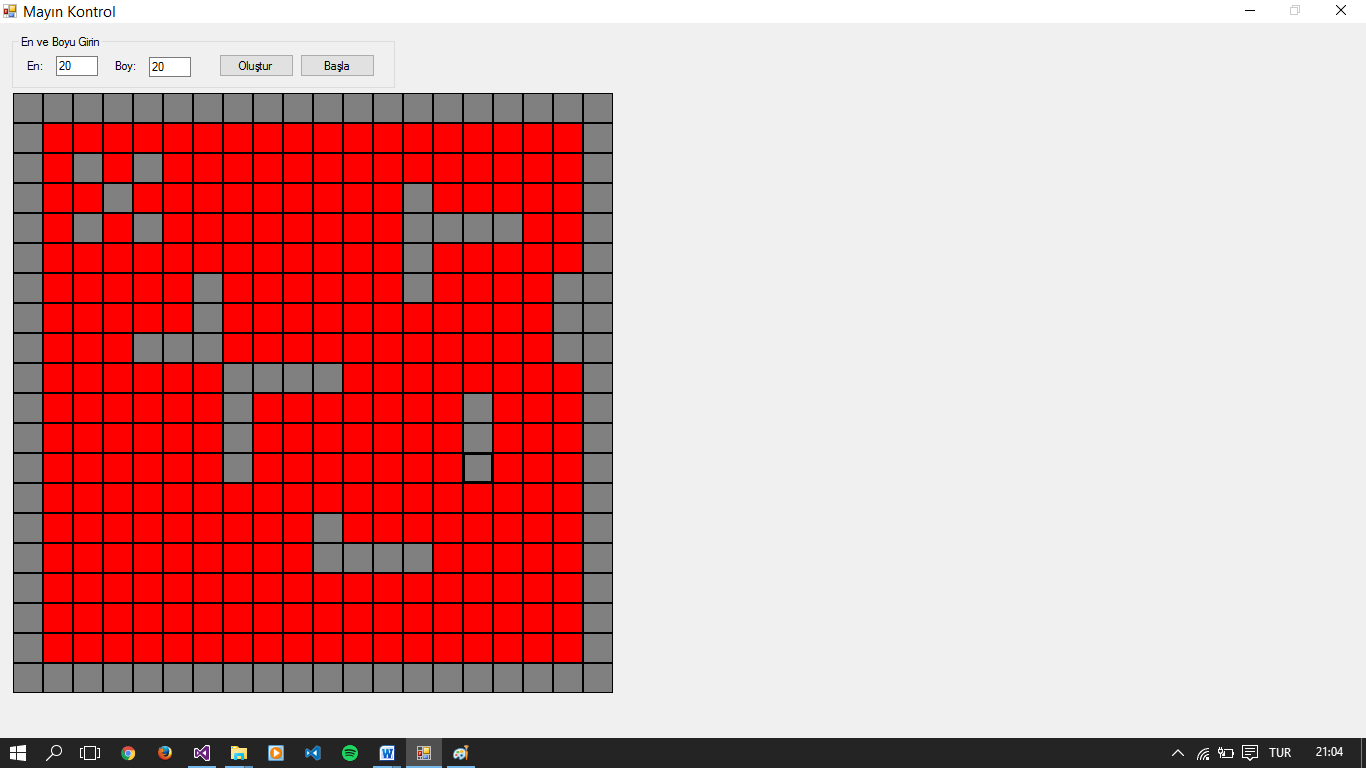
}

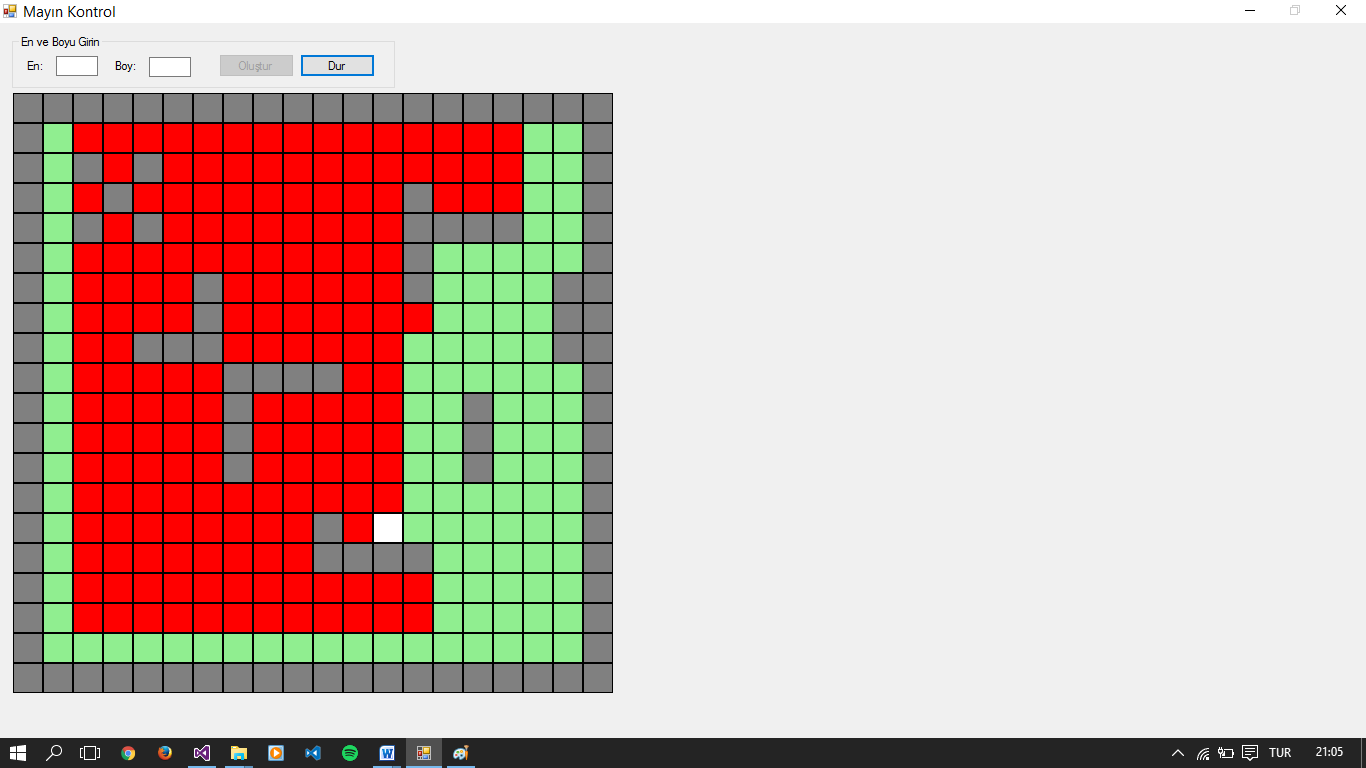
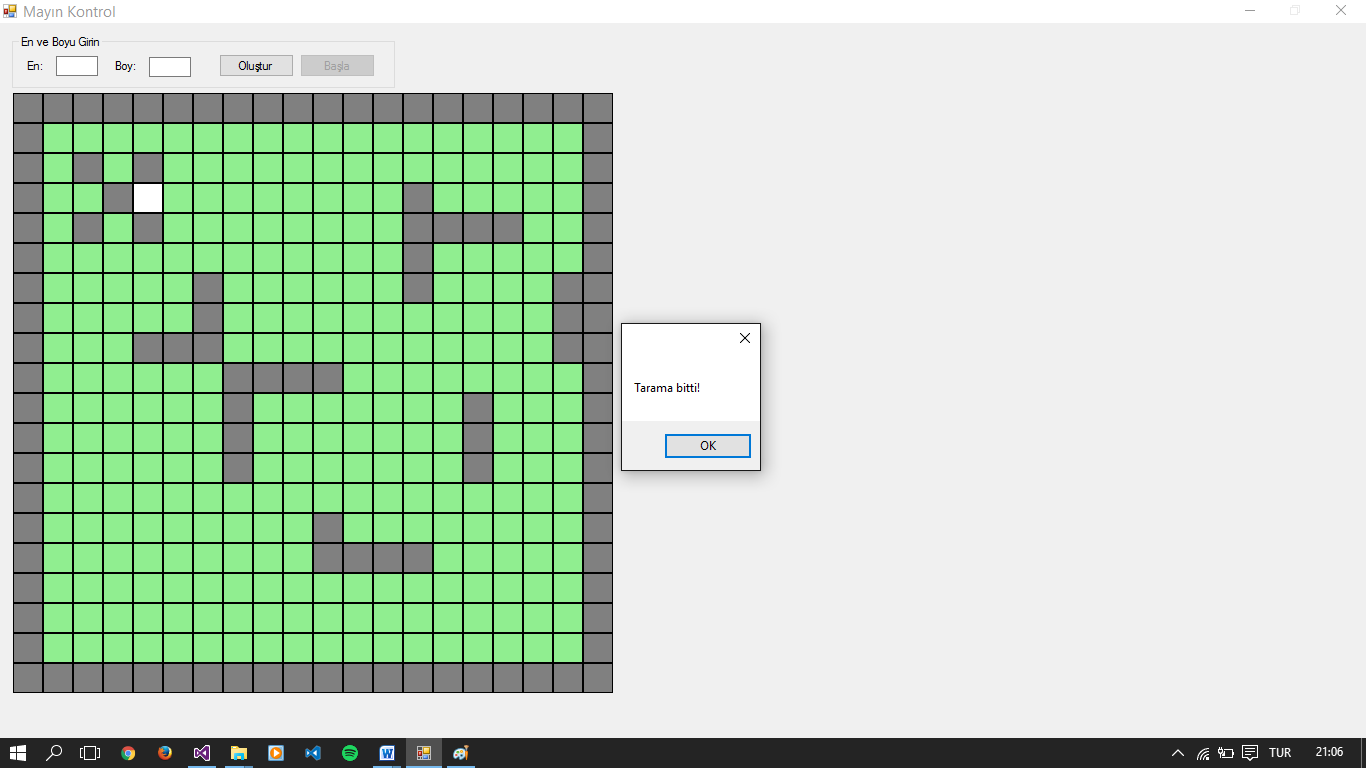
return true;

}

}

# 5. Ekran Çıktısı





# 6. Test Senaryoları

## a)Başarılı Testler

* Girdilerin sayı olup olmadığı kontrolü başarılı
* Arazi oluşturma başarılı
* Belirtilen arazinin ekran boyutunu aşmaması kontrolü başarılı
* Engel koyma başarılı
* Tarayıcının başlayacağı nokta duvarla çevriliyse yön hatası verir.
* Tarayıcı tarama işlemini sonlandırdığında işlemin tamamladığına dair bilgi verir.
* Tarayıcının başlangıç noktasına duvar konulma engeli konulmuştur.

## b)Başarısız Testler

* Tarayıcının ulaşamayacağı şekilde engeller konulursa tarayıcı sonsuz döngüye girer ve program sonlanmaz.