

PROJE 1: MAYIN TARLASI OYUNU

Mayın Tarlası, oyuncunun mayınlarla dolu bir grid (ızgara) üzerinde çalıştığı, strateji ve dikkat gerektiren klasik bir bilgisayar oyunudur. Oyun, kare bir grid'den oluşmaktadır. Oyunun amacı bir alanda mayınlara rastlamadan tüm boş kareleri bulmaktır. Karelere tıklayınca çıkan sayılar ise karenin etrafındaki mayın sayısının toplamını gösterir.

Oyun Kuralları:

1. Grid Yapısı:

- Oyun, kare bir hücre grid üzerinde oynanır.
- Her hücre, mayın içerir veya içermez,
- Başta tüm hücreler kapalıdır.

2. Oyuna Başlama:

- Oyuncu oyuna başlamadan mayınların konumları rastgele belirlenir.

3. Hücre Açma:

- Bir hücreye tıklandığında, o hücre açılır.
- Eğer açtığınız hücre bir mayın içeriyorsa, oyun biter ve kaybedersiniz.
- Eğer açtığınız hücrede mayın yoksa, o hücre ya boş olur ya da bir sayı gösterir.

4. Sayılar:

- Sayılar, o hücrenin etrafındaki **sekiz komşu hücrede** kaç tane mayın olduğunu gösterir.
- Örneğin, bir hücrede "1" yazıyorsa, o hücrenin çevresindeki sekiz komşudan birinde mayın var demektir.
- Sayılar, oyuncunun mayınların nerede olabileceğini tahmin etmesine yardımcı olur.

5. Bayrak Koyma (İşaretleme):

- Oyuncu, bir hücrede mayın olduğundan eminse o hücreye sağ tıklayarak bir bayrak koyabilir.
- Bayrak koymak, yanlışlıkla o hücreye tıklamayı engeller.

- Oyuncunun amacı, tüm mayınların bulunduğu hücreleri işaretlemek ve mayınsız olan tüm hücreleri açmaktır.

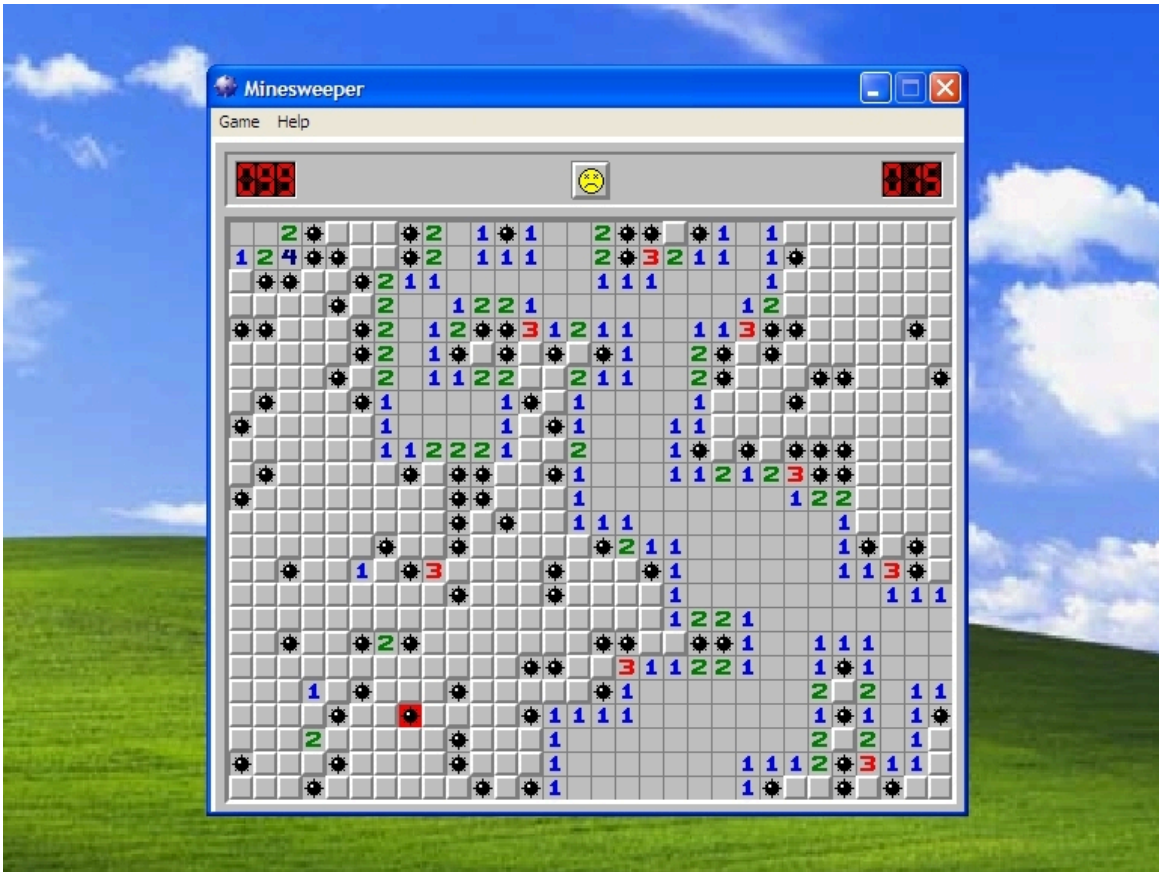
6. Otomatik Açılma:

- Eğer bir hücre açıldıysa ve bu hücrenin komşu hücrelerinde hiç mayın yoksa (yani "0" yazıyorsa), oyun bu hücreyi otomatik olarak açar ve etrafındaki diğer boş hücreleri de otomatik açar.

7. Oyunun Bitmesi:

- Oyun iki şekilde sona erebilir:
 1. Eğer bir mayına tıklarsanız, oyun sona erer ve kaybedersiniz. Oyun alanındaki tüm mayınlar açılır ve gösterilir.
 2. Eğer tüm mayınsız hücreleri açmayı başarırsanız, oyunu kazanırsınız. Kalan tüm mayınlı hücreler otomatik olarak işaretlenir.

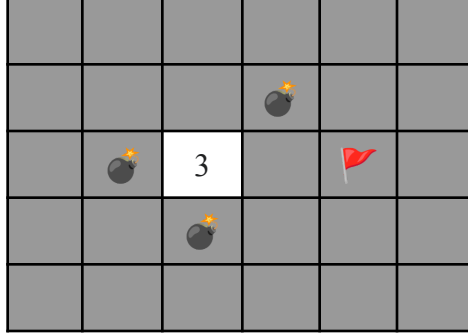
Oyuna ait örnek ekran görüntüsü aşağıdaki şekildedir. Görsel örnek olduğundan sizin uygulamanızda arayüzü farklı şekilde tasarlayabilirsiniz.



PROJE KAPSAMINDA YAPILMASI İSTENENLER

- Proje kapsamında kullanılacak programlama dili **C#**'dir. C# haricinde kullanılan programlama dilleri ile yazılan projeler **kabul edilmeyecektir.**
- Proje kapsamında oyun için bir grafiksel kullanıcı arayüzü (GUI) oluşturulacaktır. **(5 p)**
- Oyun açılır açılmaz oyuncu adı alınacak ve ardından alınacak inputa göre bir grid oluşturulacaktır. Kullanıcıya oluşturulacak alanın boyutu sorulacaktır. Örneğin kullanıcının input olarak 10 girdiğini varsayarsak 10×10 boyutunda bir oyun alanı oluşturulacaktır. En fazla 30 girilebilecektir. Oyun alanının boyutu, bilgisayar ekranından taşmayacak şekilde sınırlandırılmalıdır. **(5 p)**
- Mayın sayısı yine kullanıcıdan input olarak alınacaktır; ancak en az 10 adet mayın olabilir. 10'dan az olduğu durumlarda kullanıcıya uyarı verilmelidir. **(5 p)**
- “Oyun” adında bir sınıf oluşturulmalıdır ve oyun için gerekli kontroller ile arayüzdeki gösterilen bilgiler bu sınıf içinde tanımlanmalıdır. Örneğin; skor, sayaçlar, grid, vb. **(5 p)**
- “Skorboard” adında bir sınıf oluşturulmalıdır. En iyi 10 oyuncuya ait skor bilgileri skora göre sıralı bir şekilde bu sınıfta tutulmalıdır. Ayrıca skor ile ilgili işlemler de bu sınıfta yapılmalıdır. Skorboard sınıfına Oyun sınıfından erişim sağlanmalıdır. **(5 p)**
- Skorboard gösterme işi yeni bir form ekranında yapılmalıdır. **(5 p)**
- Skor, oyun sonlandığında bayrak ile doğru işaretlenmiş olan bombaların oyunda geçen süreye (saniye) oranının 1000 katıdır. **(10 p)**
- Oyun başlangıcında, mayınlar oyun alanına rastgele olacak şekilde yerleştirilmelidir. Bu işlem için bir metot oluşturulmalıdır. **(10 p)**
- Hücrelere sol tıklandığında hücre açma işlemi, sağ tıklandığında ise bayrak ile işaretleme işlemi yapılmalıdır. **(10 p)**
- Eğer hücre içerisinde mayın varsa oyun bitirilmelidir ve tüm mayınların yeri gösterilmelidir. Bu işlem için “Oyun” sınıfı kullanılmalıdır. **(10 p)**
- Oyunda bir grid hücresi sol tıklandığı zaman, hücrenin içinde mayın yer almıyorsa çevresindeki 8 komşu hücredeki (sağ, sol, alt, üst, alt sağ, köşeler) mayın sayısının toplamı hücre içinde sayıyla gösterilmelidir. Ek olarak, hücrenin rengi değiştirilmelidir ve hücre sağ veya sol tıklanamaz duruma getirilmelidir. Aşağıdaki şekilde gri ile gösterilen

hücreler tıklanmamıştır. Beyaz ile gösterilen sol tıklanmıştır. Örnek gösterim aşağıda yer almaktadır. **(10 p)**



- Eğer bir hücre tıklandıysa ve bu hücrenin komşu hücrelerinde hiç mayın yoksa (yani hücrede "0" yazıyorsa), oyun bu hücreyi otomatik olarak açmalıdır ve etrafındaki diğer boş hücreleri de otomatik olarak açmalıdır. **(5 p)**
- Arayüze yapılan hamle sayısını gösteren bir sayaç eklenmelidir. Oyunda hamle sınırı istenmemektedir. Sayaç, yapılan her hamle sonrasında güncellenmelidir. **(5 p)**
- Arayüzün bir bölgesinde programı geliştiren kişiye ait bilgiler yer almalıdır (Ad soyad, öğrenci numarası, vb.). **(5 p)**
- Proje kapsamında yapılan işlemler ile ilgili bir proje raporu oluşturulmalı ve proje sunumu öncesinde rapor ve projeye ait kodlar e-destek sistemi üzerinden teslim edilmelidir. Raporun içeriğinde yazılan kodlar, kodlar ile ilgili açıklamalar, oluşturulan sınıf hiyerarşisine ait Class diyagramları yer almalıdır. Rapor pdf formatında hazırlanmalıdır. Raporun adı “ogrencino_AdSoyad.pdf” şeklinde olmalıdır. Örneğin; 123123123_SevvalSolpan.pdf **(5 p)**
- Projesini sunmayanların projesi “0” (sıfır) olarak değerlendirilecektir.
- Projede ve raporda benzerlik bulunması ve yapay zeka araçlarının kullanıldığı tespit edilmesi halinde proje “0” (sıfır) olarak değerlendirilecektir.