

**KOCAELİ ÜNİVERSİTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**BLM210 PROGRAMLAMA LAB. II**  
**PROJE 1**

**GEZGİN ROBOT PROJESİ**

**Proje İlan Tarihi:** 27/02/2023

**Proje Teslim Tarihi:** 24/03/2023

Belirli kurallara göre hareket eden bir robotun önündeki engelleri aşarak istenen hedefe ulaşmasını sağlayan bir oyun tasarlanması beklenmektedir. Oyunda üç adet problemin çözülmesi gerekmektedir. Problemlerin çözümü için nesneye yönelik programlama ve veri yapıları bilgilerinin kullanılması beklenmektedir.

**Amaç:** Proje gerçekleştirimi ile öğrencilerin nesneye yönelik programlama ve veri yapıları bilgisinin pekiştirilmesi ve problem çözme becerisinin gelişimi amaçlamaktadır.

**Programlama Dili:** Proje C++, C#, Java ve Python dili kullanılarak gerçekleştirilecektir.

Projede aşağıda tanımlanan üç probleme çözüm bulmanız beklenmektedir.

**PROBLEM 1:**

Bu problemde sizden robotu ızgara (grid) üzerinde verilen hedefe engellere takılmadan en kısa sürede ve en kısa yoldan ulaştırmanız beklenmektedir. Robotu tüm ızgarayı değil, yalnızca gerekli yolları gezerek hedefe ulaşmasını sağlamalısınız.

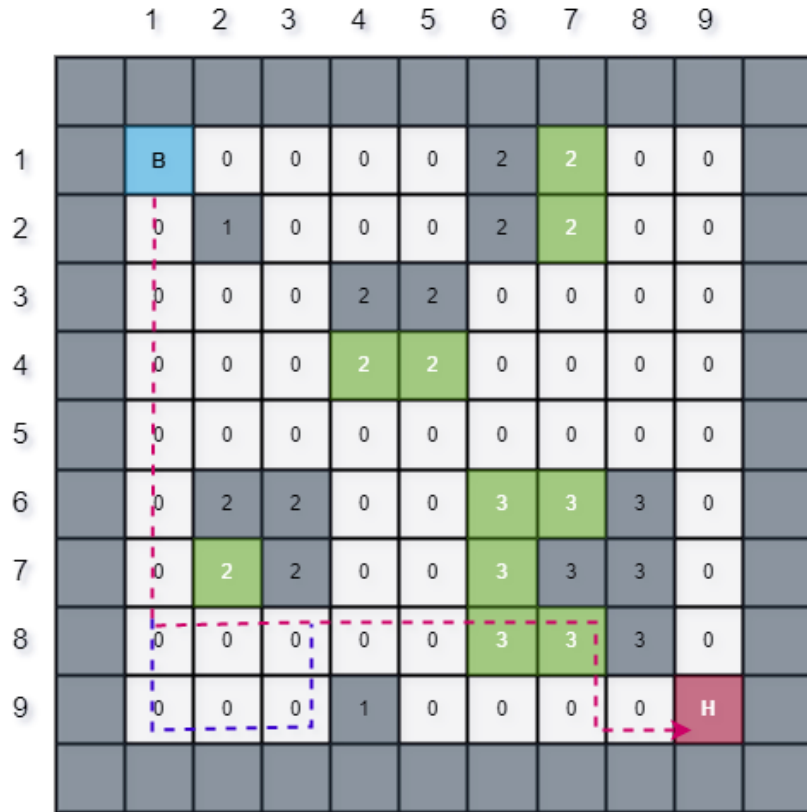
**Adım 1:** İstenilen boyutlarda karesel bir ızgara alanı oluşturmanız gerekmektedir.

**Adım 2:** Izgara üzerine engeller ve duvarlar yerleştirilmelidir. Izgara boyutu, engel sayısı ve engellerin konum bilgileri içeriği matris biçimindeki bir text dosyasından alınacaktır. Bu text dosyasına önceden verilecek bir url adresinden (e-destek üzerinden paylaşılacaktır) uygulama çalıştırıldığında otomatik olarak erişilerek dosyadaki tasarıma göre ızgara ve engel yapısı oluşturulacaktır. Engeller birbirinden farklı tipteki nesnelerden oluşabilir. (Verilecek text dosyasındaki 0 değeri engelsiz yollara; 1, 2, 3 değerleri ise üç farklı tipteki nesne için engelleri temsil edecektir. Birbirinden farklı sayıda karesel alan işgal eden bu üç engel nesnesinden 1 değerli nesne yalnızca 1 karelik alan 2 değerine sahip nesneler yanyana 2 kare içeren maksimum 2x2 lik; 3 değerine sahip nesneler ise yan yana 3 kare içeren maksimum 3x3 lük kare alana Şekil 1’deki gibi yerleştirilecektir. )

**Adım 3:** Robotun başlangıç ve hedef noktaları ızgara üzerindeki uygun (engel veya duvar içermeyen) karelere rastgele belirlenmelidir. Robot başlangıçta tüm ızgara dünyasını bilmemelidir, sadece bir adım sonraki kareleri görebilmelidir. Her adımda robotun

öğrenmediği kareler bulutlu (kapalı) olarak gösterilmeli, öğrenilen kareler ise açılarak ilgili karelerde bulunan nesneye göre (engel, duvar, yol, vs.) belirtilmelidir.

**Adım 4:** Tüm bu bilgiler doğrultusunda, robotun hedefe en kısa sürede ulaşabileceği en kısa yol, adım adım ızgara üzerinde gösterilmelidir. Robotun daha önce geçtiği yerler belli olacak şekilde her adımda yol üzerinde iz bırakması gerekmektedir. Hedefe ulaşıldığında ise başlangıç noktasından hedef konuma giden robota göre en kısa yol ızgara üzerinde ayrıca çizdirilmelidir. Geçen toplam süre (sn cinsinden) ve kaç kare üzerinden geçildiği bilgileri ekranda gösterilmelidir.



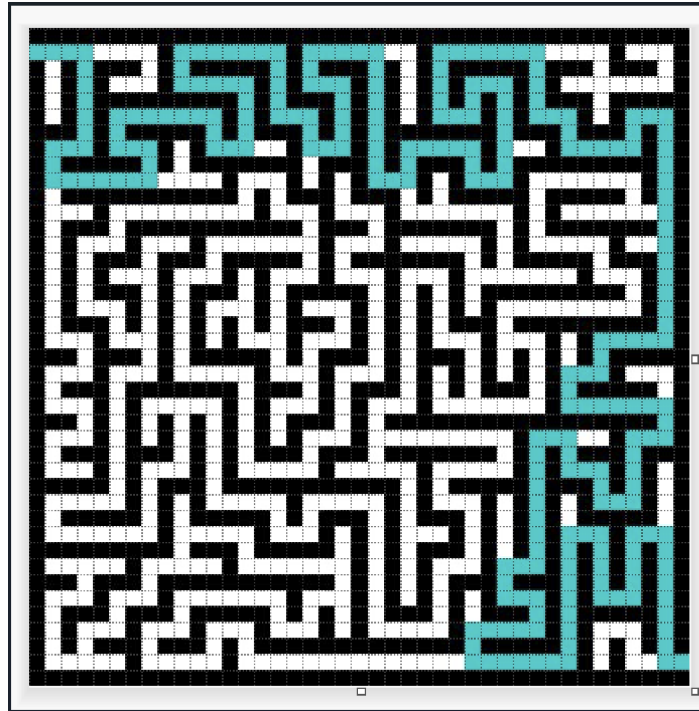
Şekil 1. Problem 1 için örnek ızgara ve engel görünümü

## PROBLEM 2:

Bu problemde sizden robotu labirentteki çıkış noktasına ulaştırmanız beklenmektedir.

**Adım 1:** Kullanıcı tarafından istenilen boyutlarda bir ızgara oluşturmanız gerekmektedir.

**Adım 2:** ızgara üzerine 1 nolu tipte engeller yerleştirilerek labirent oluşturulmalıdır. Labirent içerisinde mutlaka çıkışa ulaşamayan yollar bulunmalıdır.



### Engel Sınıfı:

- Engel sınıfında problem 1 için üç farklı tipteki nesneler kullanarak engellerin oluşturulması url adresindeki text dosyasına göre yapılırken problem 2 için labirent oluşumu 1 nolu tek bir nesne türü kullanılarak uygulama içerisinde rastgele oluşturulacaktır.

### Uygulama Sınıfı:

- Uygulama içerisinde robotun problem 1 ve problem 2 deki hedefe ulaşma süresi, kaç kare üzerinden geçildiği gibi bilgilerin tutulduğu ve ekranda gösterilmesi fonksiyonlarını sağlamalıdır.

**NOT:** Yukarıdaki sınıflar işlevsel bakımdan genel olarak tanımlanmış olup her sınıf için kullanılacak özellik (property) ve metotların tanımlaması sizden beklenmektedir. Ayrıca gerektiği durumlarda neden kullanıldığı açıklanması koşuluyla yukarıdaki sınıflardan farklı olarak sınıf tanımlamaları yapılabilecektir.

Her sınıf için ortak olan özellikler:

Projede Encapsulation, Inheritance, Polymorphism, Abstraction yapılarından gerekli olanların kullanılması gerekmektedir. Projede her yapıyı kullanmasanız bile, proje sunumu esnasında bu yapıların ne olduğunu bilip bilmediğinizi ölçecek sorular sorulacaktır.

### **Arayüz ve Görsellik:**

- Problem 1 ve problem 2 için uygulama ilk çalıştırıldığında başta robotun nasıl hareket ettiği simüle edilmeli ancak sonrasında zaman kaybı olmaması için bir buton aracılığıyla sonuç kullanıcıya direk gösterilmelidir. Sonuç olarak robotun sırasıyla gezdiği kareler ve bulunduğu en kısa yol ekranda çizdirilmeli ayrıca bir text dosyasına yazdırılarak kaydedilmelidir.

### **ÖDEV TESLİMİ**

- Proje raporu IEEE formatında (önceki yıllarda verilen formatta) 4 sayfa uzunluğunda olmalıdır. Rapor; akış diyagramı veya yalancı kod içermeli, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır. **Raporda UML sınıf diyagramının çizilmesi beklenmektedir.**
- Dersin takibi projenin teslimi dâhil edestek.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek.kocaeli.edu.tr sitesinde belirtilen tarihten sonra teslim edilen projeler kabul edilmeyecektir.
- Proje ile ilgili sorular edestek.kocaeli.edu.tr sitesindeki **forum üzerinden** Arş. Gör. Gamze Korkmaz Erdem veya Arş. Gör. Abdurrahman Gün'e sorulabilir. **Proje teslimine 2 gün kala sorulan hiçbir soruya cevap verilmeyecektir.**
- Sunum tarihleri daha sonra duyurulacaktır.

- Sunum sırasında;
  - Algoritma, geliřtirdiđiniz kodun eřitli kısımlarının ne amaçla yazıldıđı ve geliřtirme ortamı hakkında sorular sorulabilir.
  - Kullandıđınız herhangi bir satır kodu açıklamanız istenebilir.

## Notlandırma

Ařađıdaki notlandırma bilgisi projeyi yaparken planlama yapabilmeniz aısından paylařılmıştır.

Sunumda sorulara verilen cevaplar, kod hakimiyeti ve dokümanda belirtilen maddeler projenin genel kriterleri ierisinde deđerlendirilecektir.

Uygulama Alanı: Izgara ve engel-labirent tasarımının isterlere uygun olması .....	15
Problem 1: Robotun isterlere uygun řekilde hedefine ulařması .....	20
Problem 2: Robotun isterlere uygun řekilde hedefine ulařması .....	20
NYP: Tanımlanması gereken tüm yapı, icerikler ve NYP'ye uygunluđu .....	20
Görsellik: Uygulamanın görselliđi ve dođru alışması .....	15
Rapor: UML, İERİK, kazanımlar, yorumlar .....	10

Toplam: 100

**Projenin tanıtım toplantısı 1 Mart 2023 arřamba günü saat 15:30'da bölüm duyurularında ve e-destekte duyurulacak toplantı linki üzerinden uzaktan yapılacaktır.**

**Proje grupları en fazla 2 kiřiden oluřmalıdır. Proje grup bilgileri e-destekte paylařılacak link üzerinden en ge 24 Mart Cuma gününe kadar girilmelidir. Bu tarihten sonra gruplarda herhangi bir deđeriklik yapılmayacaktır.**