

# Programlama Laboratuvarı I. Proje Ödevi

Osman Aldemir  
Bilgisayar Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye  
osmanaldir@gmail.com

Tuna Kömür  
Bilgisayar Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye  
kmrtuna9@gmail.com

**Özetçe**—"Otonom Hazine Avcısı" projesi, nesne yönelimli programlama ve veri yapıları konularında öğrencilerin becerilerini geliştirmeyi amaçlayan bir oyun uygulamasıdır. Proje kapsamında, karakterin rastgele oluşturulan bir harita üzerinde belirlenmiş hazine sandıklarını en kısa sürede toplamasını sağlayacak bir algoritma tasarlanmıştır. Kullanıcıdan alınan harita boyut bilgisine göre rastgele harita oluşturulmakta, karakterin başlangıç ve bitiş noktaları belirlenmekte ve en kısa yol algoritması kullanılarak karakterin hedefe ulaşması sağlanmaktadır.

Projede, java programlama dili kullanılmış ve nesne yönelimli programlama prensipleri takip edilmiştir. Uygulama, öğrencilerin programlama becerilerini geliştirmek ve algoritmik düşünme yeteneklerini artırmak için etkili bir öğrenme aracı olarak tasarlanmıştır.

## I. AMAÇ

Bu proje, "Otonom Hazine Avcısı" adını taşıyan bir oyun uygulamasının geliştirilmesini amaçlamaktadır. Bu oyun, bir karakterin rastgele oluşturulan bir harita üzerindeki hazineleri toplamasını sağlayacak bir algoritmayı içerecektir. Oyunun amacı, program taradından rastgele olarak belirlenen başlangıç noktasından başlayarak tüm hazine sandıklarını en kısa sürede ve en kısa yol ile toplamak olacaktır. Bu proje, nesneye yönelik programlama ve veri yapıları bilgisinin pekiştirilmesi, uygulanması ve problem çözme becerisinin geliştirilmesini hedeflemektedir.

### A. Başlamadan Önce

Projenin başlamadan önceki aşaması, detaylı bir araştırma ve analiz sürecini içermiştir. Bu süreçte, benzer oyun projeleri incelenmiş, kullanılacak programlama dili ve teknolojiler belirlenmiş ve projenin genel stratejisi oluşturulmuştur. Geliştirme ekibi, proje gereksinimlerini ve kullanıcı beklentilerini dikkate alarak başlangıç adımlarını planlamış ve projenin teknik detaylarını ele almıştır.

### B. Başlangıç

Projenin başlangıç aşamasında, temel bileşenlerin tasarımı ve altyapısı hazırlanmıştır. Harita oluşturma, engellerin yerleştirilmesi ve karakterin hareketi gibi temel işlevlerin prototipleri geliştirilmiş ve test edilmiştir. Ayrıca, proje ekibi arasında görev dağılımı yapılarak, ilerleyen aşamalarda uygulanacak olan algoritma ve görsel arayüz tasarımı için hazırlıklar tamamlanmıştır.

## C. İlerleyiş

Projenin ilerleyiş aşamasında, temel bileşenlerin detaylı olarak kodlanması ve entegrasyonu gerçekleştirilmiştir. İşte bu sürecin ana adımları:

**Harita Oluşturma Algoritması Geliştirme:**  
Haritanın rastgele oluşturulması için geliştirilen algoritma, belirlenen boyutlarda ve temaya uygun bir harita üretmek üzere uygulanmıştır. Bu algoritma, her seferinde farklı bir harita oluşturarak oyun deneyimini çeşitlendirmiştir.

**Engel ve Hazine Nesnelerinin Yerleştirilmesi:**  
Harita üzerindeki sabit ve hareketli engellerin yerleştirilmesi, belirlenen kurallara göre gerçekleştirilmiştir. Ayrıca, hazine sandıklarının belirli bir öncelik sırasına göre yerleştirilmesi sağlanmıştır.

**Karakterin Hareket Algoritması Geliştirme:**  
Karakterin otomatik hareketini sağlayacak algoritma, en kısa yol hesaplamasını ve engelleri aşmayı içerecek şekilde kodlanmıştır. Karakterin yolu üzerinde bulunan ve henüz keşfedilmemiş kareler sisli olarak gösterilmiştir.

**En Kısa Yol Algoritması ve Görselleştirme:**  
Tüm hazine sandıklarını en kısa sürede toplamak için geliştirilen en kısa yol algoritması, karakterin hareket güzergahını belirlemiştir. Bu yol, harita üzerinde yeşil renkle gösterilmiş ve karakterin hareketini takip etmek için kullanıcıya sunulmuştur.

**Arayüz Tasarımı ve Butonlar:**  
Kullanıcı arayüzünde "Yeni Harita Oluştur" ve "Başlat" butonları yer almıştır. "Yeni Harita Oluştur" butonuna basıldığında yeni bir harita oluşturulmuş ve başlangıç noktası belirlenmiştir. "Başlat" butonuna tıklandığında ise karakterin hareketi başlamış ve oyun sonuçları kullanıcıya gösterilmiştir.

## II. GİRİŞ

Otonom Hazine Avcısı projesi, nesneye yönelik programlama ve veri yapıları konularının pratiğe dökülmesini sağlayan bir uygulamadır. Bu proje kapsamında, oyun karakterinin bir harita üzerinde rastgele yerleştirilmiş hazine sandıklarını en kısa sürede toplamasını sağlayacak bir algoritma geliştirilmesi amaçlanmıştır. Bu hedefe ulaşmak için, karakterin harita üzerinde otomatik olarak hareket etmesi, engelleri aşması ve hazine sandıklarını toplaması gerekmektedir.

Projenin temel amacı, öğrencilerin nesneye yönelik programlama ve veri yapıları konularında edindikleri teorik bilgileri pratiğe dökerek derinleştirmelerini sağlamaktır. Ayrıca,

bu proje sayesinde problem çözme becerileri ve algoritma geliştirme yetenekleri de geliştirilmektedir.

Oyunun genel adımları arasında haritanın oluşturulması, engellerin ve hazine sandıklarının yerleştirilmesi, karakterin hareket etmesi ve en kısa yolun hesaplanması gibi adımlar bulunmaktadır. Proje sürecinde, nesneye yönelik programlama prensipleri (Encapsulation, Inheritance, Polymorphism, Abstraction) ve veri yapıları (kuyruklar, ağaçlar) kullanılarak kodlar geliştirilecektir.

Bu projenin raporunda, proje gereksinimleri, kullanılan programlama dili ve sınıf tanımları gibi detaylar ele alınacak, ardından projenin gerçekleştirilmesiyle elde edilen sonuçlar ve deneyimler paylaşılacaktır. Son olarak, projenin gelecekteki gelişim potansiyeli ve uygulanabilir alanları hakkında değerlendirmeler yapılacaktır.

### III. YÖNTEM

#### Harita Oluşturma ve İçerik Tasarımı:

İlk olarak, kullanıcıdan haritanın boyut bilgisi alınmış ve bu bilgiye göre bir ızgara oluşturulmuştur. Harita oluşturulurken, sabit ve hareketli engellerin yanı sıra hazine sandıkları da rastgele yerleştirilmiştir. Sabit engeller (ağaçlar, kayalar, duvarlar ve dağlar) yaz ve kış temalarına göre farklılık göstermektedir. Hareketli engeller (kuşlar ve arılar) belirli bir alanda rastgele hareket etmektedir. Hazine sandıkları ise farklı türlerde (altın, gümüş, zümrüt ve bakır) ve öncelik sırasına göre yerleştirilmiştir.

#### Başlangıç ve Bitiş Noktalarının Belirlenmesi:

Karakterin başlangıç noktası, uygun ve engel içermeyen bir karede rastgele belirlenmiştir. Bitiş noktası ise en son toplanacak hazine sandığının konumu olarak belirlenmiştir.

#### Karakterin Hareketi ve En Kısa Yol Algoritması:

Karakter, hareket ederken çapraz yönde değil, sadece sağ, sol, yukarı veya aşağı yönde ilerlemektedir. Oyun boyunca karakterin hareketi otomatiktir ve karakterin toplamak için en kısa yolu hesaplamak için A\* algoritması kullanılmıştır. Bu algoritma, karakterin tüm hazine sandıklarını en kısa sürede toplamasını sağlayacak en optimum yolu belirlemektedir. Karakterin ilerlediği yol, yeşil renkle gösterilmekte ve kaç adım atıldığı bilgisi ekranda yazdırılmaktadır.

#### Arayüz ve Görsellik:

Proje arayüzünde, "Yeni Harita Oluştur" ve "Başlat" olmak üzere iki buton bulunmaktadır. "Yeni Harita Oluştur" butonuna basıldığında, rastgele bir harita oluşturulmaktadır ve harita üzerindeki sis kaldırılmaktadır. "Başlat" butonuna basıldığında ise karakter, belirlenen en kısa yol üzerinde hareket etmeye başlamaktadır. Ayrıca, ekranda karakterin gezdiği kareler ve bulunduğu en kısa yolun görsel olarak takibi sağlanmaktadır.

### IV. KATKILAR

#### Tuna Kömür:

Arayüz tasarımları, paneldeki yeni harita oluştur ve oyunu başlatma işlemleri, kullanıcı hatalı bir giriş yaptığında gösterilecek hatalar ve koddaki bug çözümlemelerinde daha baskın çalışmıştır.

Osman Aldemir:

Class'ları tanımlama ve düzenleme, class'lardaki implement ve extends uygulamaları, override işlemleri ve kod mimarisinde çalışmıştır.

### V. SONUÇ

Otonom Hazine Avcısı projesi kapsamında gerçekleştirdiğimiz çalışmalarla, nesneye yönelik programlama ve veri yapıları konularında önemli deneyimler kazandık. Projeyi tamamlarken Java programlama dilini kullanarak karakterin bir harita üzerindeki hazine sandıklarını en kısa sürede toplamasını sağlayacak bir algoritma geliştirdik.

Projenin genel adımları arasında haritanın dinamik olarak oluşturulması, engellerin ve hazine sandıklarının yerleştirilmesi, başlangıç ve bitiş noktalarının belirlenmesi, karakterin hareketi ve en kısa yolun hesaplanması gibi adımlar bulunmaktadır.

Haritanın oluşturulması aşamasında, kullanıcıdan alınan boyut bilgisi doğrultusunda harita oluşturuldu. Bu süreçte, her uygulama başladığında yeni bir harita üretilmesi sağlandı ve haritanın sol tarafı kış, sağ tarafı yaz temasında oluşturuldu. Harita içeriğinde yol, engeller ve hazine sandıkları bulunmakta olup, engeller sabit (ağaçlar, kayalar, duvarlar, dağlar) ve hareketli (kuşlar, arılar) olarak tasarlandı.

Karakterin başlangıç noktası uygun bir kareye rastgele olarak yerleştirilirken, bitiş noktası ise toplanacak en son hazine sandığının konumu olarak belirlendi. Karakterin hareketi otomatik olarak gerçekleştirildi ve en kısa yol algoritması kullanılarak hazine sandıklarını toplamak için en kısa yol hesaplandı. Bu hesaplanan en kısa yol harita üzerinde yeşil renkle gösterilirken, toplam adım sayısı ve toplanan nesnelerin bilgisi ekranda yazdırıldı.

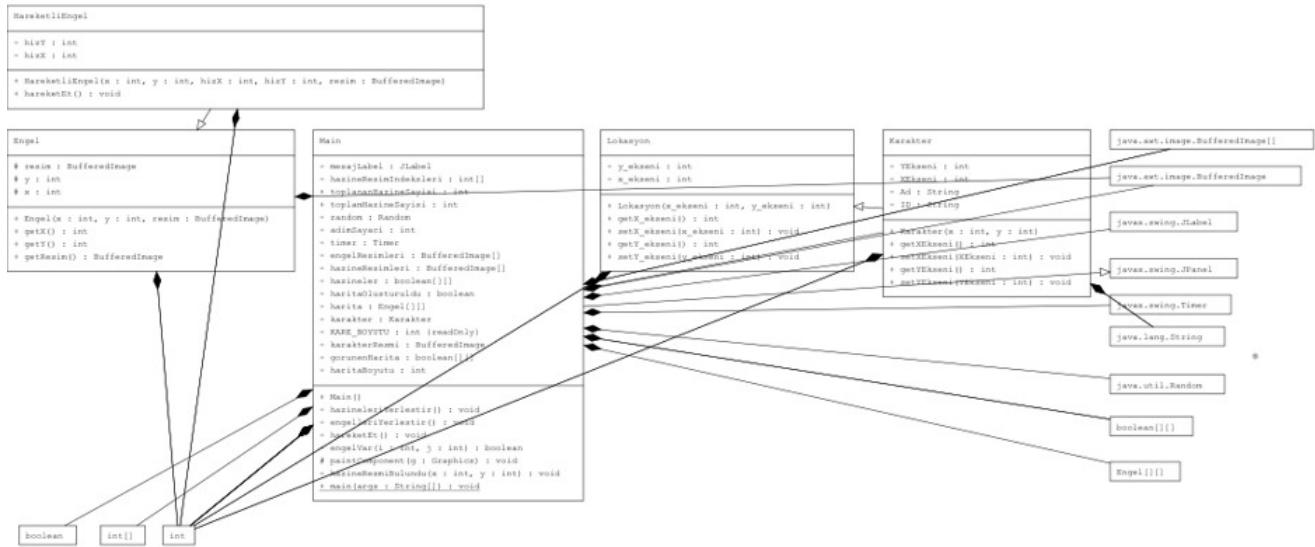
Proje sırasında Encapsulation, Inheritance, Polymorphism, Abstraction gibi nesneye yönelik programlama prensipleri ve veri yapıları kullanılarak kodlar geliştirildi. Ayrıca, projenin arayüzünde yer alan görsellerin net ve anlaşılır olmasına özen gösterildi.

Sonuç olarak, Otonom Hazine Avcısı projesi sayesinde nesneye yönelik programlama becerilerimizi geliştirdik ve veri yapıları konusunda daha fazla deneyim kazandık. Bu proje, problemleri çözmek için etkili algoritmalar geliştirmeyi, kodlama becerilerimizi geliştirmeyi ve kompleks sistemleri yönetmeyi öğrenmemize olanak sağladı.

### KAYNAKLAR

- [1] [https://www.tutorialspoint.com/java\\_dip/java\\_buffered\\_image.htm](https://www.tutorialspoint.com/java_dip/java_buffered_image.htm)
- [2] <https://stackoverflow.com/questions/35638104/how-do-i-properly-create-a-bufferedimage-from-file>
- [3] <https://www.youtube.com/watch?v=QW3iL3zFJSU&list=PL4yfBYtaNjbRRGLQnrTPU2eiAB5rgi9lx>
- [4] [https://www.youtube.com/watch?v=om59cwR7psi&list=PL\\_QPQmz5C6WUF-pOQDsbsKbaBZqXj4qSq](https://www.youtube.com/watch?v=om59cwR7psi&list=PL_QPQmz5C6WUF-pOQDsbsKbaBZqXj4qSq)

## VI. DENEYSEL SONUÇ



## VII. HATALAR

