

BURSA ULUDAĞ ÜNİVERİSTESİ

BİLGİSAYAR MÜHENDİSLİĞİ

PROJE RAPORU

HÜSEYİN ÖZKAN

**İÇİNDEKİLER**

**1. Otomatik Pilot Projesi**

1.1 Projenin Kodlanma Mantığı

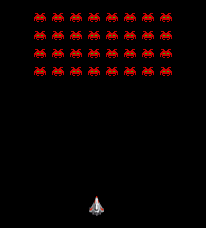
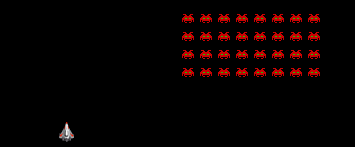
1.2 Proje Kodları ve Algoritmalar

ÖZSÖZ

Bu raporda yapılmış olan bir yapay zeka projesinin kodlama mantığı ve algoritmaları anlatılmıştır.Yapay zeka projesindeki amaç el ile kontrol edilebilen bir uzay aracını otomatik pilot moduna alarak en kısa zamanda ve en kısa adımda bütün hedefleri vurmak.

**1.1 Projenin Kodlanma Mantığı**

Proje iki temel mantığa göre kodlanmıştır. İlk olarak uzay aracı hedefe odaklanmaktadır. Hedefe odaklıysa ateş etmektedir. Hedefe odaklanırken en sağdaki hedefe mi odaklanmalı yoksa en soldaki hedefe mi odaklanmalı sorusunu uzay aracının konumu belirlemektedir. Eğer uzay aracı hedeflerin dışında ve en solunda ise (Resim 1.1) en soldaki hedefe odaklanır, en sağında ise (Resim 1.2) en sağdaki hedefe odaklanır ve ateş edilir. Eğer uzay aracı hedeflerin içinde ise (Resim 1.3) hangi yöne yakın olduğuna bakılır ve o yöndeki hedefe odaklanır. Bu mantıkla gittiğimizde Alienlerin sayısı ve uzay aracının başlama yeri değişken olabiliyor. Her türlü bütün hedefler temizleniyor.

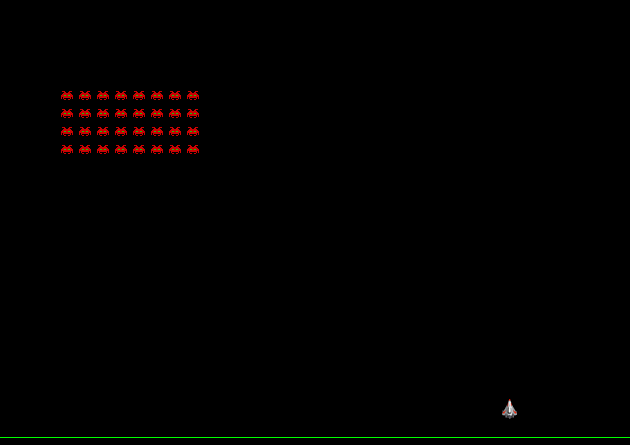


Resim 1.1 Resim 1.2 Resim 1.3

Proje bu mantıkla kodlandığında uzay aracı en iyi durumda başlarsa (Resim 1.4) 118 adım ve yaklaşık 37 saniyede tüm hedefleri vurabiliyor. Eğer uzay aracı en kötü durumda başlarsa (Resim 1.5) 150 adım ve yaklaşık 47 saniyede tüm hedefleri vurabiliyor. Ortalamayı düşünürsek 135 adım ve yaklaşık 43 saniyede tüm hedefler vuruluyor.



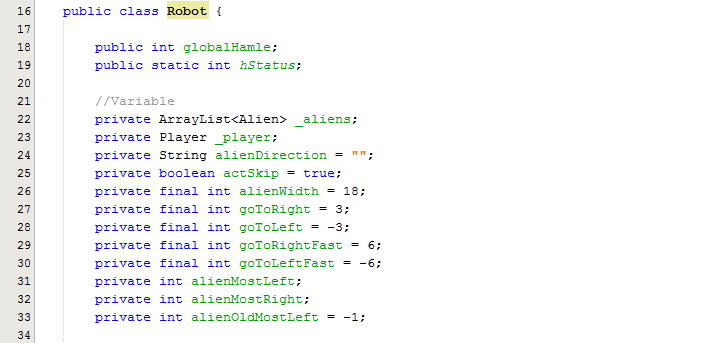
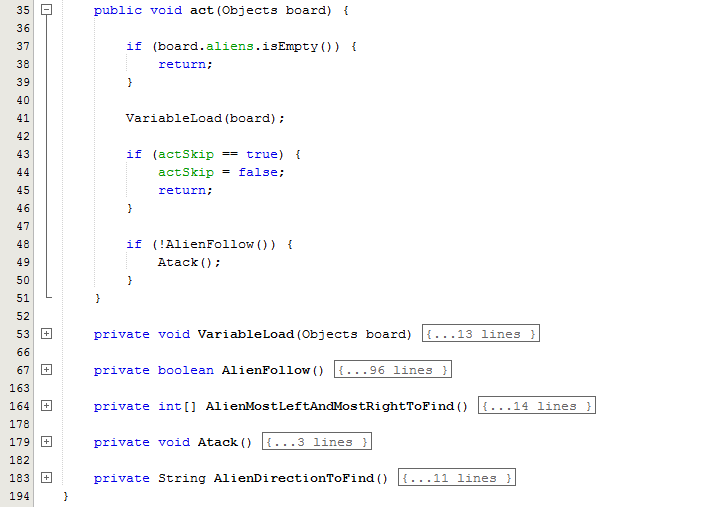
Resim 1.4 En iyi durum



Resim 1.5 En kötü durum

**1.2 Proje Kodları ve Algoritmalar**

Proje kodlarında genel olarak yerel değişkenler ve yerel fonksiyonlar mevcuttur. Uzay aracının hareket edebilmesi için act(); fonksiyonu 300 ms 'de bir tetikleniyor. Burada globalHamle ye belirli değerler vererek uzay aracını sağa, sola veya hedeflere ateş ettirebiliyoruz. Resim 1.6 ' daki kodları incelersek act fonksiyonunda 3 temel kod var. Birincisi VariableLoad() fonksiyonudur. Bu fonsiyon yerel değişkenlerimizin değerlerini doldurmaktadır. İkinci kod bloğunda actSkip işlemi yapılıyor. actSkip işlemi act fonksiyonunun ilk tetiklenmesinde hiçbir işlem yapmadan return yapıyor ve ikinci tetiklenmeyi bekliyor ve diğer tetiklenmelerde çalışmıyor. Bu işlemin sebebi projede Alienlerin gittiği yöne ihtiyacımızın olmasıdır. İkinci tetiklenmeden sonra Alienlerin yeni ve eski konumunu bildiğimiz için yönünü rahatlıkla bulabiliyoruz. Son olarak üçüncü kod bloğunda eğer AlienFollow() fonksiyonu false değer dönüyorsa uzay aracı takip yapmadı yani hedefte demektir ve atack yapması gerekir, true değer dönüyorsa takip yapıldı demektir. Hedefe gitmek için sağa veya sola haraket ettirildi.

****

Resim 1.6 Robot sınıfı

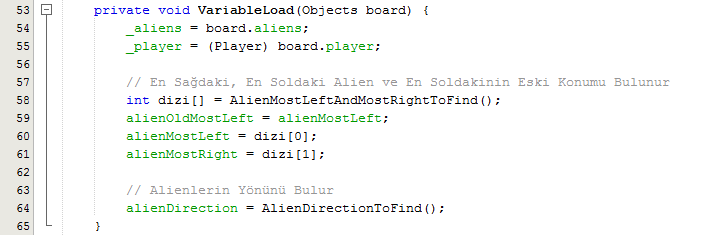
**Değişken Açıklamaları;**

Değişkenler VariableLoad(); fonksiyonunda yüklenmektedir.

* **\_aliens :** Vurulacak Alienler atanır.
* **\_player :** Uzay gemisi atanır.
* **alienDirection :** Alienlerin gittiği yön atanır.
* **actSkip:** Varsayılan değeri true dir. Act() fonksiyonunun ilk çalışmasını atlamak için kullanıldı.
* **alienWidth :** alien genişliği atandı.
* **goToRight :** Varsayılan değeri 3 olarak atandı. Uzay aracını 3 piksel sağa haraket ettirmek için kullanıldı.
* **goToLeft :** Varsayılan deperi -3 olarak atandı. Uzay aracını 3 piksel sola haraket ettirmek için kullanıldı.
* **goToRightFast :** Varsayılan değeri 6 olarak atandı. Uzay aracını daha hızlı bir şekilde sağa hareket ettirmek için kullanıldı.
* **goToLeftFast :** Varsayılan değeri -6 olarak atandı. uzay aracını daha hızlı bir şekilde sola hareket ettirmek için kullanıldı.
* **alienMostLeft :** En soldaki alienin x değeri atandı.
* **alienOldMostLeft :** En soldaki alienin eski x değeri atandı. Bu sayede alienlerin yönü bulunuyor.
* **alienMostRight :** En sağdaki alienin x değeri atandı.

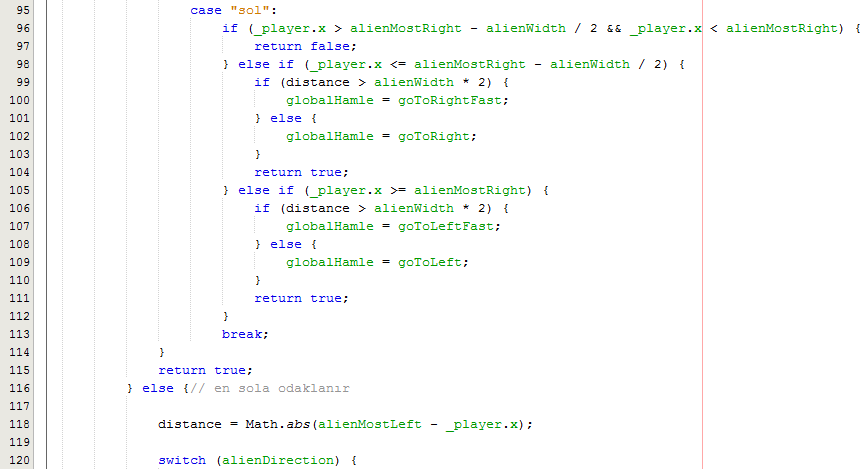
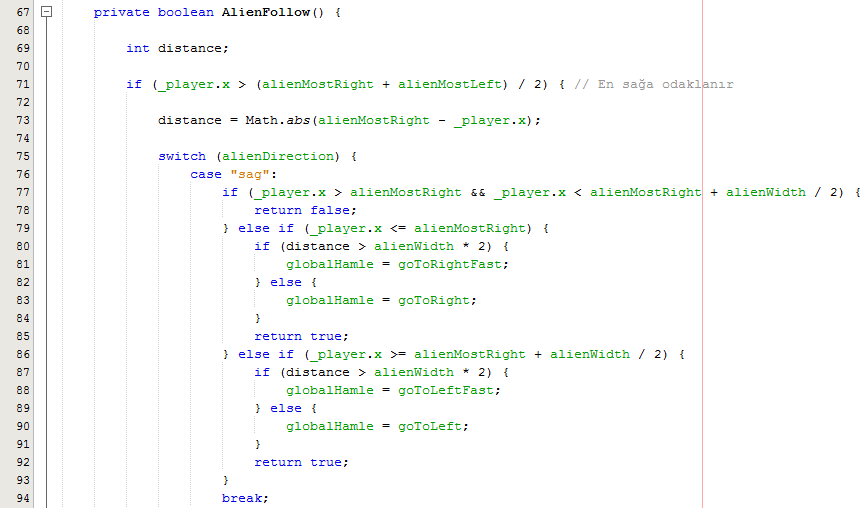
**Kullanılan Fonksiyonlar ;**

**VariableLoad() :** Bu fonksiyonda act() fonksiyonun her tetiklenmesinde yerel değişkenlerin güncel değerleri yükleniyor (Resim 1.7).

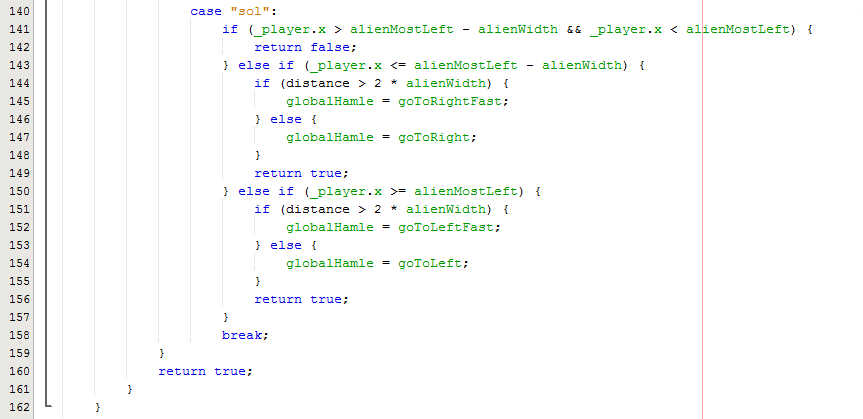
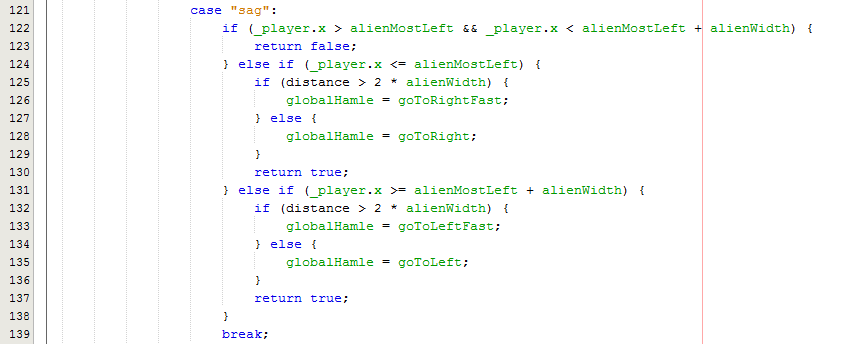


Resim 1.7 VariableLoad()

**AlienFollow() :** Bu fonksiyonda uzay aracının yakınlığına göre en sağ Alien 'e yada en sol Alien 'e odaklanır (Resim 1.8). Eğer uzay aracı sağ tarafdaki hedefe yakın ise en sağa odaklanır. Distance değişkenine uzay aracının odaklanması gereken hedefin uzaklığı atılır. Eğer distance değeri Alien ' in genişliğinden iki katından fazla ise hedefe hızlı bir şekilde (6 piksel) gidilir. Değilse hedefe yakınlaşmış demektir ve yavaş (3 piksel) hareket edilir. En sağa odaklanıp distance değerini aldıktan sonra Alienler sağa veya sola gitme durumuna bakılır. Eğer Alienler sağa gidiyorsa uzay aracının odakta olma durumuna bakılır. Odakta ise return false; değeri döner. Bu işlem uzay aracını haraket ettirmeye gerek yok, hedefin içinde bulunuyor ve ateş etmeli anlamına gelir. Odağın dışındaysa hedef uzay aracının sağında mı kalıyor yoksa solunda mı durumuna bakılır. Eğer sağında kalıyorsa distance değerine göre hızlı yada yavaş bir şekilde sağ tarafa hareket ettiriliyor ve return true; dönerek takip işlemi yapıldı anlamına geliyor ve ateş edilmiyor. Benzer işlemler Alienlerin Sola gitme durumunda ve aynı zamanda uzay aracının en sola odaklanması durumunda da yapılır.

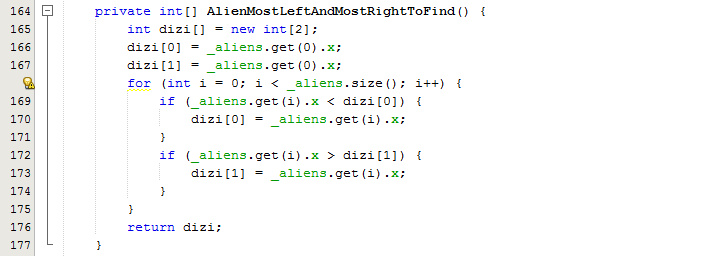


Resim 1.8 AlienFollow()



Resim 1.9 AlienFollow() devamı

**AlienMostLeftAndMostRightToFind() :** Bu fonksiyonda en soldaki Alien ve en sağdaki Alien bulunur (Resim 1.10). İlk olarak dizi[0] ' a ve dizi[1] 'e sıfırıncı indeks deki değerler alınır. Sonrasında bütün Alienler dolaşılarak x değeri en küçük olan dizi[0] ' a, x değeri en büyük olanda dizi[1] 'e atanır. Böylelikler en sağdaki ve en soldaki Alien bulunmuş olur. VariableLoad() fonksiyonunda bu fonksiyon kullanılarak atama işlemleri yapılır.



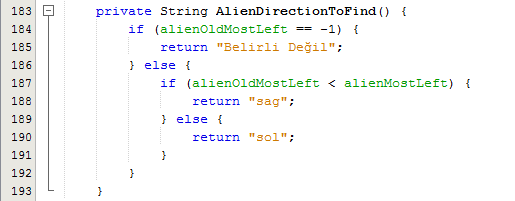
Resim 1.10 AlienMostLeftAndMostRightToFind()

**Atack() :** Bu fonksiyon hedeflere ateş etmek için kullanıldı (Resim 1.11). Eğer uzay aracı takipte ise Atack() fonksiyonu çağırılıyor ve ateş ediliyor.

C:\Users\root\Desktop\atack.PNG

Resim 1.11 Atack()

**AlienDirectionToFind() :** Bu fonksiyonda Alienlerin gittiği yön bulundu. Eğer alienMostLeft -1 ise eski konum bilinmiyor ve yön teyit edilemiyor. Zaten actSkip işleminde bu durumu kontrol edip bir sonraki tetiklemeye geçmişdik. İkinci ve sonrası tetiklemerde en soldaki Alien ' in yeni ve eski konumu biliniyor. Eğer yeni konum eski konumdan büyükse Alienler sağa hareket ediyordur. Büyük değilse solda hareket ediyordur. Geriye string değeri dönerek bu bilgileri AlienFollow() fonksiyonunda kullandık.



Resim 1.12 AlienDirectionToFind()