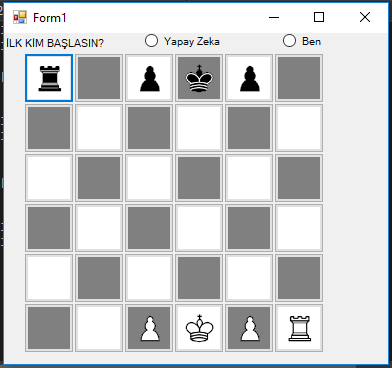
# **YAPAY ZEKA PROJE ÖDEVİ**

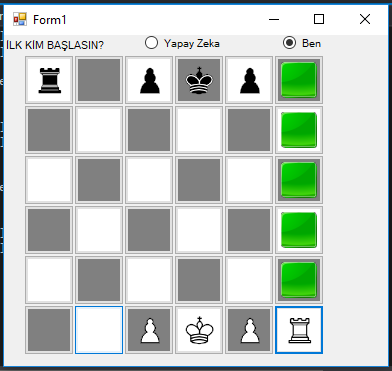
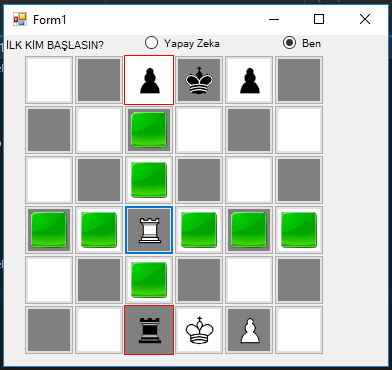
Bu projede yapay zeka ile satranç oyunu yaptım. Algoritmamı min-max ile tasarlayıp kullandım. Projemi C# kodları ile yazdım. Tasarım kısmında C# Form Uygulamasını kullanarak satranç tahtasını butonlar ile yaptım.

**BAŞLANGIÇ TAHTASI**

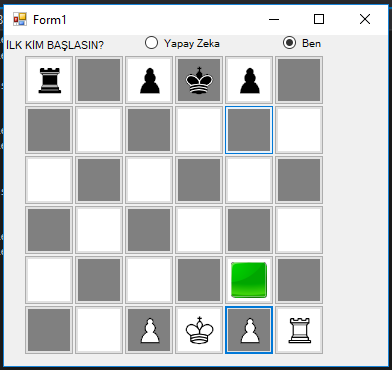
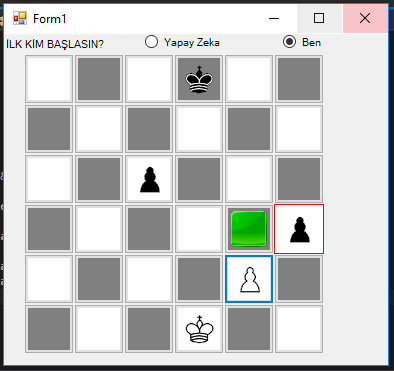
****

Satranç taşlarının kurallara göre hareket edebilecekleri yerleri ve rakip taşı yemesinin gösterimini yaptım.

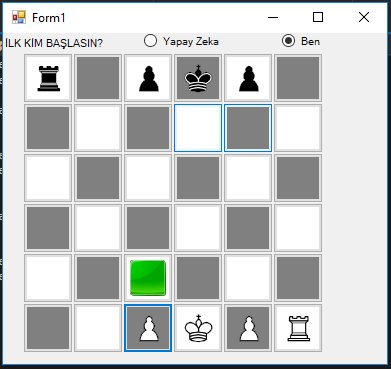
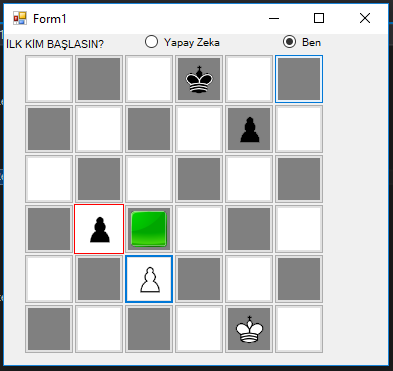
**KALE HAREKETİ**

** **

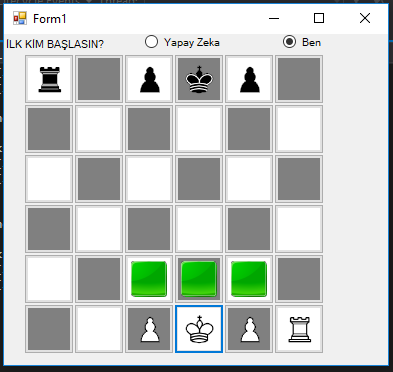
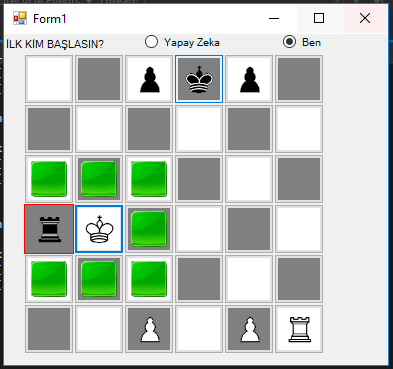
**PİYON HAREKETİ**

** **

**PİYON HAREKETİ**

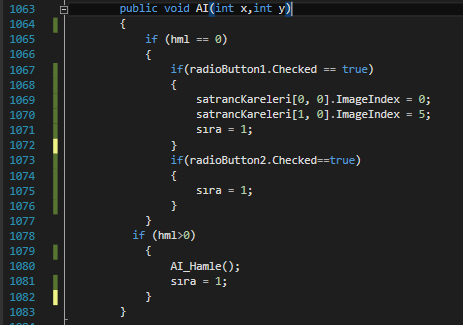
** **

**ŞAH HAREKETİ**

** **

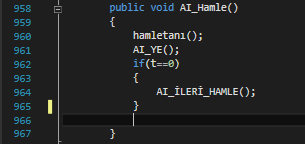
**YAPAY ZEKA KODLARI**

AI Fonksiyonu çalıştığında başlangıcın kim yapacağını kontrol ederek yapay zekada olup olmadığını belirtiyorum ve AI\_Hamle Fonksiyonunu çağırarak hamle yapmasını istiyorum.



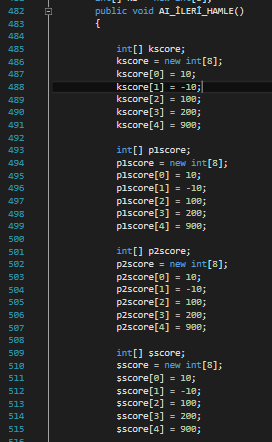
**AI\_Hamle() Fonksiyonu**

AI\_Hamle Fonksiyonu içerisinde ilk olarak yapay zekanın hamletanı(); Fonksiyonu ile taşlarının konumlarını algılamasını istiyorum. Sonrasında AI\_YE(); Fonksiyonunu çağırarak yapay zeka taşlarının, rakip taşları yeme durumu olup olmadığını kontrol ediyor. Eğer rakip taşı yeme gibi bir durum söz konusu değilse AI\_İLERİ\_HAMLE(); Fonksiyonunu çağırarak yapay zeka en iyi hamlesini bulup taşını ona göre hareket ettirecektir.

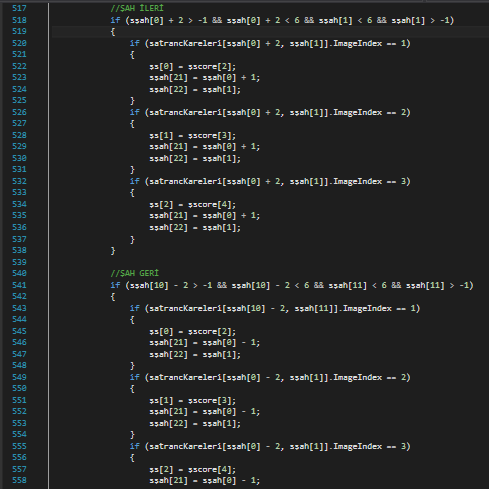


**AI\_İLERİ\_HAMLE() Fonksiyonu**

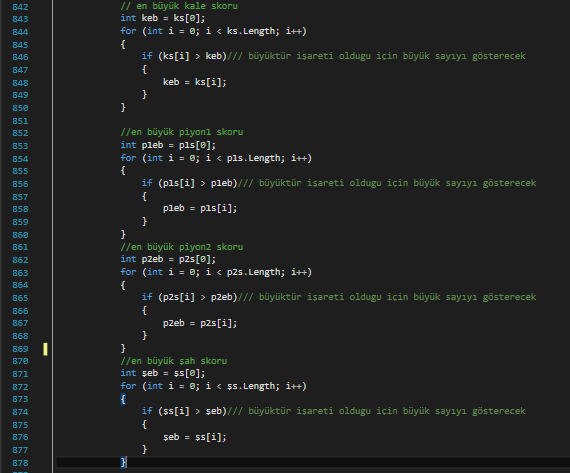
İlk olarak her bir taşın skor tablosunu oluşturdum.



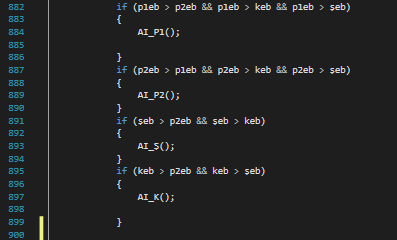
Örnek olarak şah taşının bir sonraki adımına göre puanlama yapıyor.



Sonrasında puan bilgileri alındıktan sonra bu puanlar içinde en büyük puanı seçerek yapay zekanın hamle yapmasına 1 adım daha yaklaşıyoruz.



Taşların en yüksek skorları alındıktan sonra taşlar arasında seçim yaparak hamlemizi tamamlayacağız.



Son olarak en yüksek skora sahip taşın fonksiyonu çalıştırılarak yapay zekanın hamle yapmasını sağlıyoruz.

