TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

YAPAY ZEKA DERSİ FİNAL PROJESİ
GENETİK ALGORİTMA KULLANILARAK 2048 OYUNU

15011079 — Yasemin KESKİN

Danışman
Doç. Dr. Mehmet Fatih AMASYALI

Haziran, 2020

İÇİNDEKİLER

ŞEKİL LİSTESİ

1
GİRİŞ

Gerçekleştirilen projenin temel amacı, son yıllarda popülaritesinin yüksekliği ile bilinen tek oyunculu çevrimiçi bir oyun olan 2048 oyununun genetik algoritma kullanılarak tasarlanmasıdır.

Genetik algoritmanın kodlanması için kullanılan programlama dili Python olarak seçilmiştir. Genetik algoritmanın her jenerasyon ve mutasyon işlemleri sonrasında hangi fitnes değerlerine sahip olduğu ise Tkinter toolkitinin kullanılması ile gösterilmiştir.

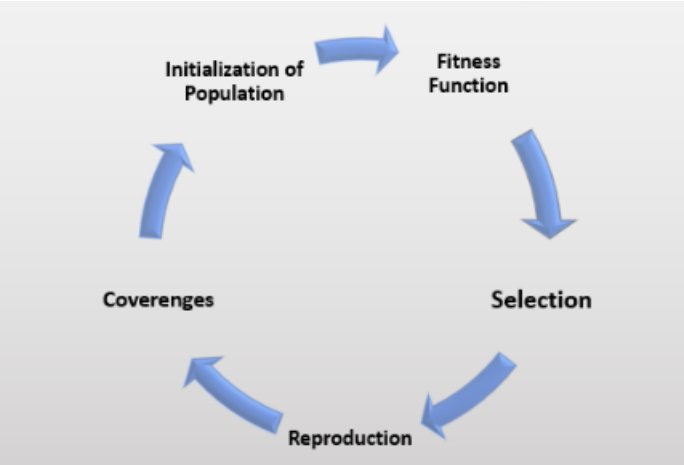
Projenin arayüzü ise oyunun orijinalliğinin de korunması amacıyla Selenium modülünün kullanılması ile gerçekleştirilmiştir.

2
YÖNTEM

2048 oyununun gerçekleştirilmesi için genetik algoritma seçilmiştir. Algoritmanın temel adımları ve temelde nasıl çalıştığıyla alakalı bilgiler aşağıdaki gibi özetlenmiştir.

2.1 GENETİK ALGORİTMA NEDİR?

Genetik algoritmalar[1], evrimsel sürece benzer bir yapıda çalışan optimizasyon algoritmalarına denir. Uygunluk fonksiyonunun optimize edilmesi amaçlanmaktadır.Şekil-2.1'de genetik algoritmanın genel adımları görülmektedir.



Şekil 2.1 Genetik Algoritmanın Temel Adımları

Genetik algoritmada, başlangıç popülasyonu rastgele olarak üretilmektedir. Bu popülasyondaki bireylerin uygunluk fonksiyonuna göre uygunluk değerleri belirlenir. Algoritma üç aşama şeklinde çalışmaktadır: seçim (selection), çaprazlama(crossover) ve mutasyon (mutation). Bu adımlar, yeni nesilde oluşturulan bireylere uygulanmaktadır.

Bu aşamalarda gerçekleştirilen işlemler:

- Seçim: Bireylerin uygunluk değerlerine göre yeni nesli oluşturmak için ebeveyn (parent) seçilmesi.
- Çaprazlama: Ebeveyn kromozomlarındaki çeşitli kısımların karşılıklı olarak yer değiştirmesi.
- Mutasyon: Belirli bir olasılığa göre yeni nesildeki bireylerin kromozomlarındaki bir genin mutasyona uğraması.

Algoritma, istenen optimum çözüm bulunduğunda veya maksimum iterasyon sayısına ulaşıldığında çalışmayı durdurmaktadır.

2.1.1 Tkinter Araç Kiti

Tkinter (Tk)[2], birçok programlama dili tarafından GUI tasarlamak için kullanılan açık kaynaklı, multi-platform bir toolkittir. Bu projede de kullanıcının, önemli verileri izlemesine yardımcı olmak için web sitesinden alınan HTML' yi gerçek zamanlı olarak güncelleyen Tkinter kullanılmıştır.

2.1.2 Selenium Modülü

Selenium[3], kişisel bilgisayara yüklenen bir driver yardımı ile ekrana chrome, firefox gibi bir tarayıcı açarak, gerçek bir insan gibi istenilen tüm işlemleri programlama dili yardımıyla çalıştırmayı sağlayan bir araçtır. Selenium kullanıldığında bir tarayıcı açılarak bir web sitesine bağlanıp belirli alanlara tıklayarak gezinilebiliyor, bazı formlar doldurulabiliyor veya butonlara tıklanıldığında bazı yerlerin ekrana gelmesi sağlanılabiliyor. Gerçekleştirilen projede de yapılan diğer projeler de incelendikten sonra Selenium modülü kullanılmıştır. Yüklenen driver ile chrome tarayıcısı üzerinden oyun yaratıcısının web sitesine ulaşılmış ve o web sitesi kullanılmıştır.

3
UYGULAMA

3.1 Problemin Çözümü

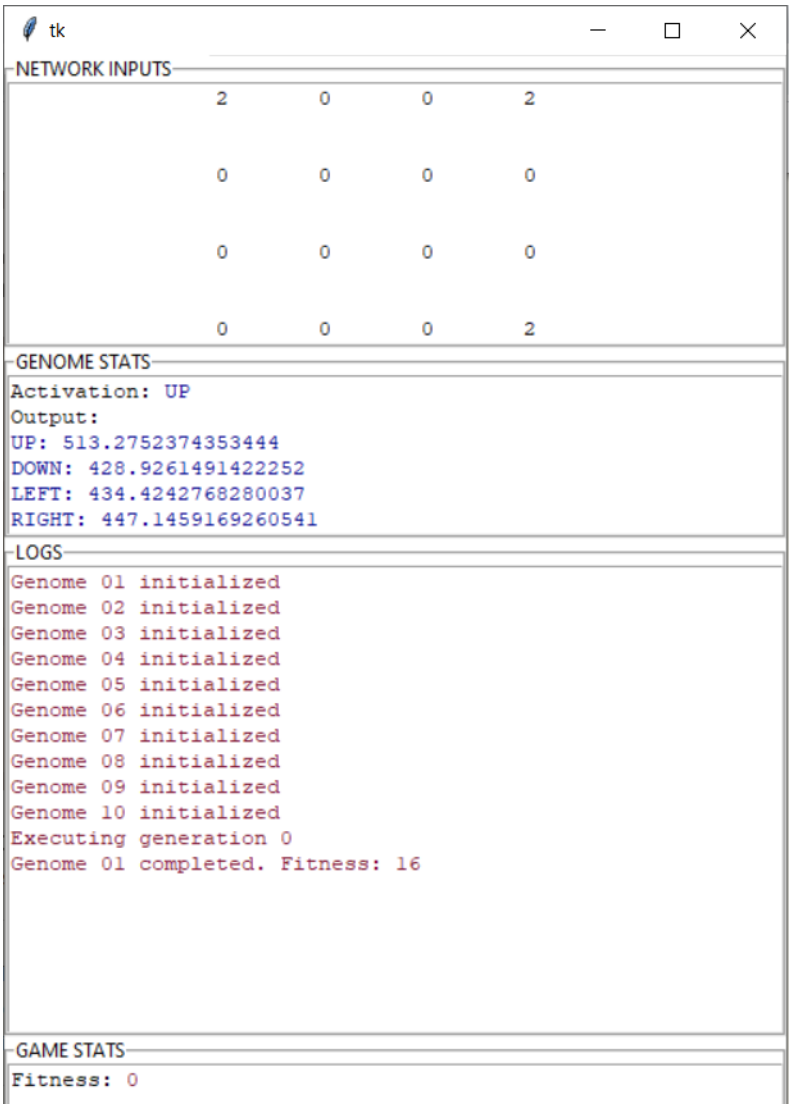
İlk olarak gerçekleştirilecek proje konusu ile ilgili çeşitli kaynaklardan bilgi toplanarak araştırmalar yapılmıştır. Konu hakkında yeterli bilgiye sahip olunduktan sonra genetik algoritmanın implementasyonu aşamasına geçilmiştir.

Genetik algoritmadaki önemli bilgilere ulaşılmayı kolaylaştırmak için Tkinter araç kiti kullanılmıştır. Burada, oyunun her genomdaki ve her jenerasyondaki durumu güncellenerek network lobunda, her genomun 4 farklı durum için de aldığı değerler ve seçilen değer genom lobunda, hangi genom ve hangi jenerasyonda olunduğu logs lobunda, o andaki fitnes değeri ise game lobunda kullanıcıya gösterilmiştir.

Daha sonrasında genetik algoritmanın 2048 oyununda nasıl çalıştığını gözlemlemek için https://play2048.co/ [4] web sitesi kullanılarak projenin arayüzü tamamlanmıştır.

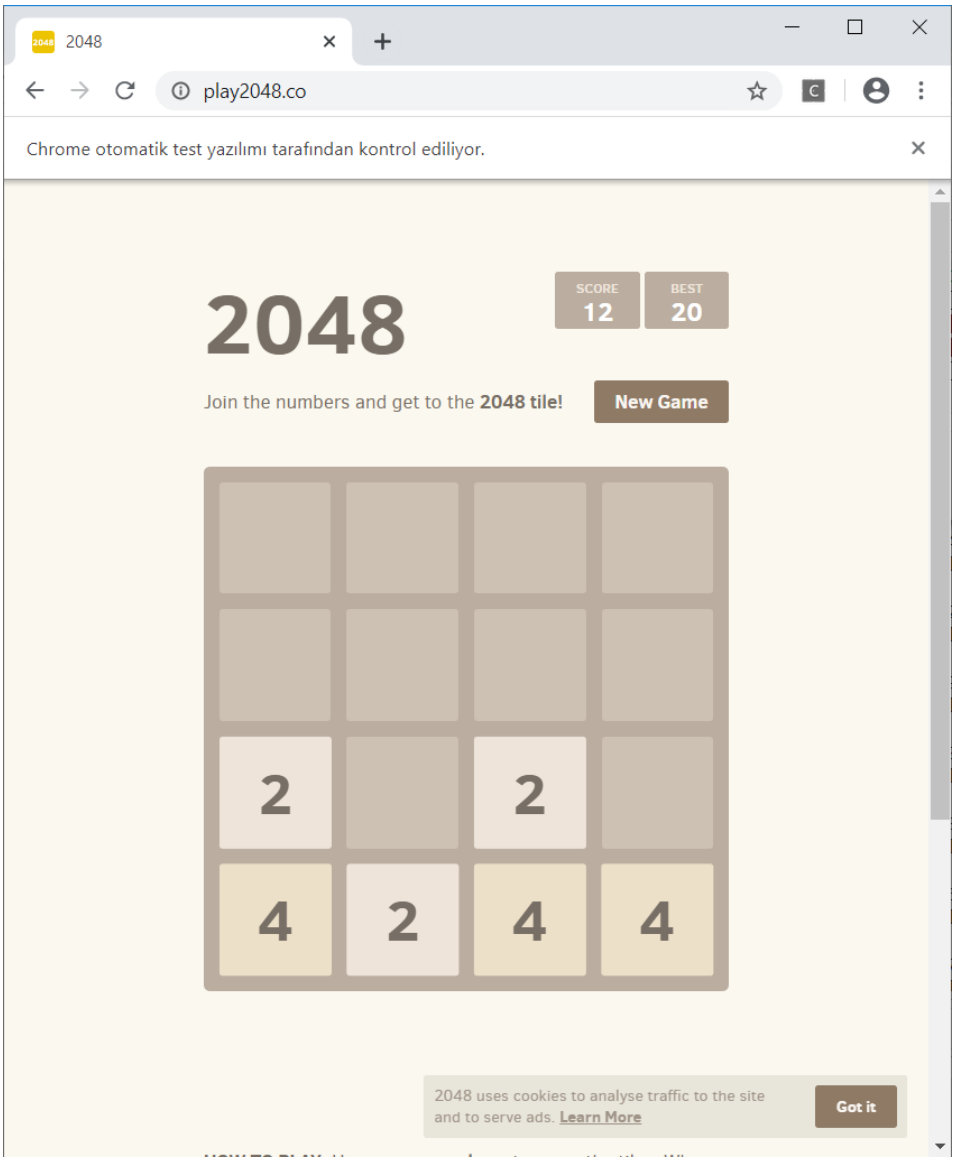
Genetik algoritma ile 2048 oyununun gerçeklenmesi 10 genom ve 50 jenerasyon kullanılarak tamamlanmıştır.

Şekil-3.1'de oluşturulmuş olan Tkinter'ın bir örneği gösterilmektedir.



Şekil 3.1 Tkinter

Şekil-3.2 ’de kullanılan arayüz gösterilmektedir.



Şekil 3.2 Projenin Arayüzü

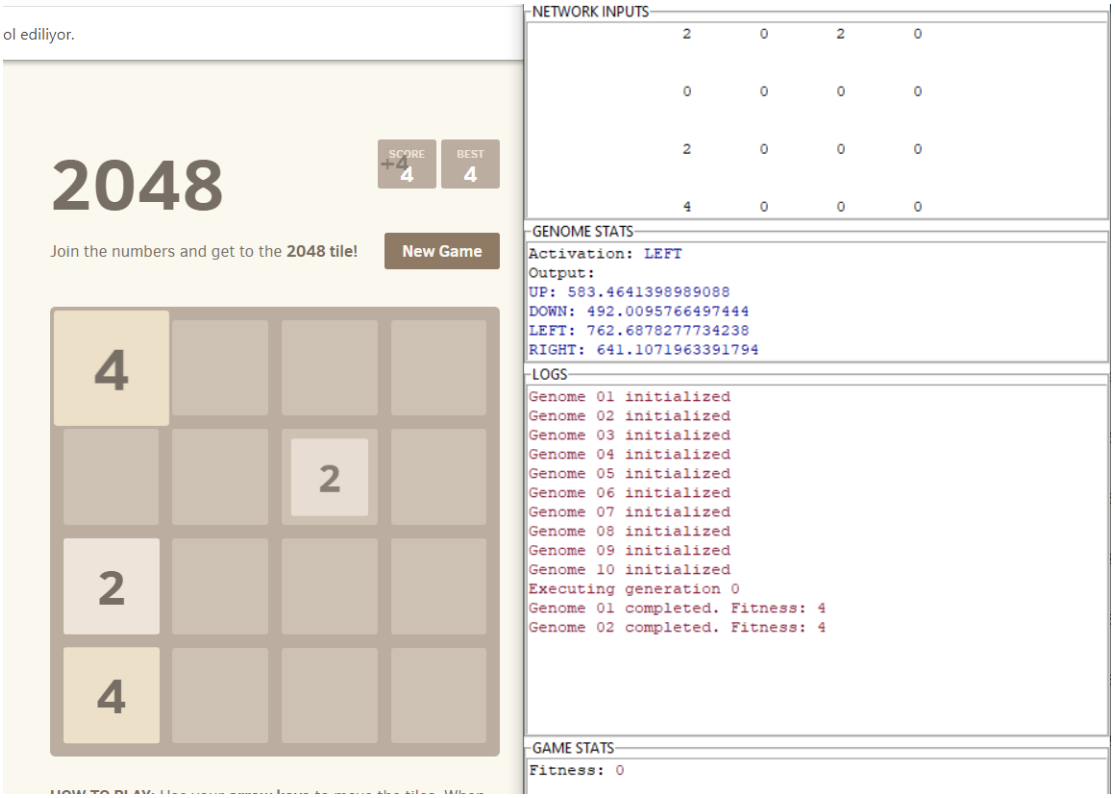
3.2 Algoritmanın Başarı Performansı

Program, rastgele jenerasyon başına 10 "Genom" üretir ve genlerin bir geçiş algoritması aracılığıyla bir sonraki jenerasyona aktarmak için her jenerasyondan en iyi iki performansı seçer. Her Genom'un, bazı özelliklerini değiştiren bir mutasyon algoritmasından etkilenme olasılığı % 4 olarak seçilmiştir. Sonunda yeterli nesiller ile, sonuçta en iyi performans gösteren Genom 2048' in üzerinde bir puana sürekli olarak ulaşabilir.

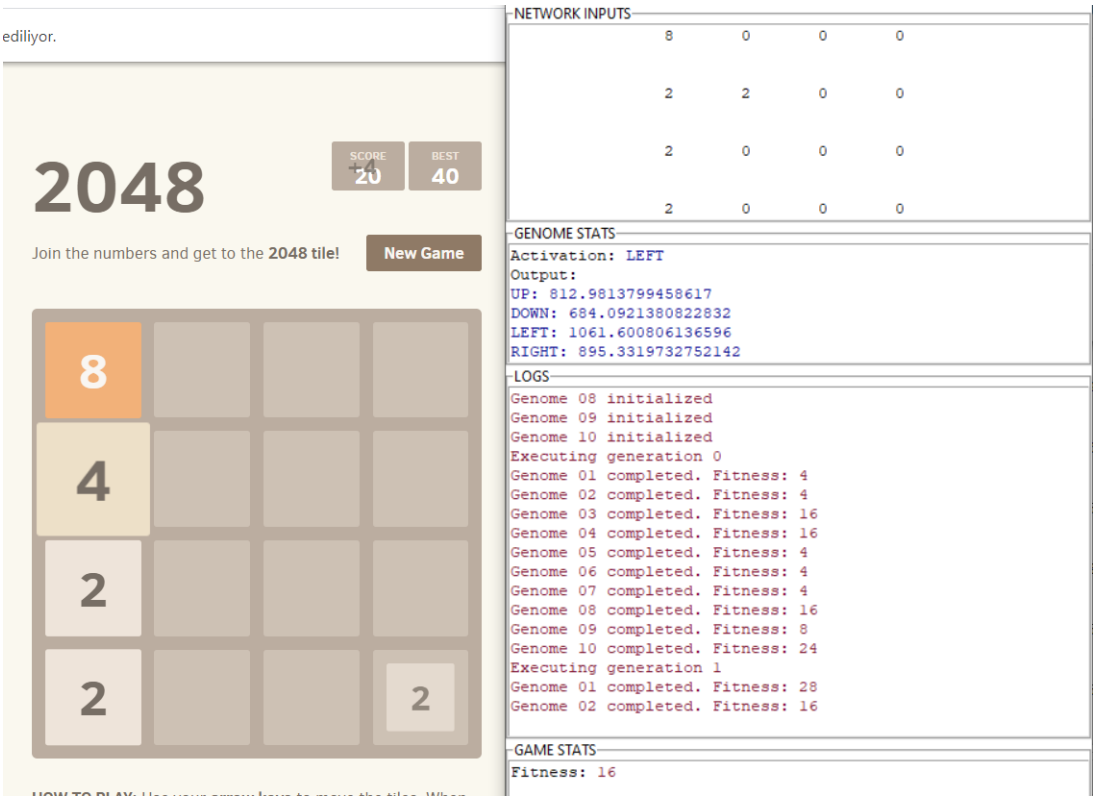
Yapılan testler sonucunda mutasyon oranının artırılması ya da jenerasyon sayısının çoğaltılması Genom' un 2048 sayısına ulaşma ihtimalini yükselttiği gözlemlenmiştir.

3.3 Çalıştırma Örnekleri

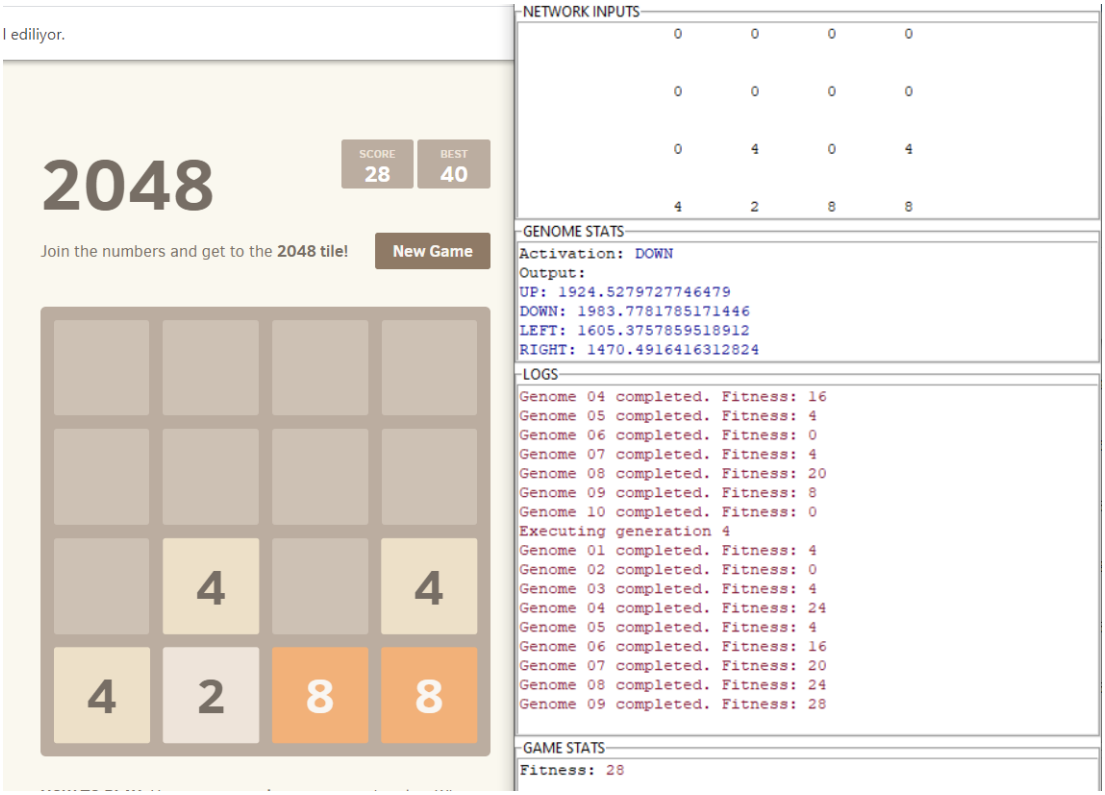
Aşağıdaki figürlerde algoritmanın 2048 oyunu için 50 jenerasyon boyunca mutasyon oranı 0.5 olarak verildiğinde nasıl çalıştığı birkaç örnek ile gösterilmiştir.



Şekil 3.3 2048 Oyununun İlk Jenerasyonda Çalıştığı Bir Örnek



Şekil 3.4 2048 Oyununun 1. Jenerasyonda Çalıştığı Bir Örnek



Şekil 3.5 2048 Oyununun 4. Jenerasyonda Çalıştığı Bir Örnek

4
SONUÇ

Bu projede genetik algoritmanın kullanılması ile 2048 oyunu gerçekleştirilmiştir. Proje boyunca gözlemlenen ilk detay, jenerasyon sayısının arttırılması ile oyunda yapılabilecek en yüksek değer olan 2048 sayısına ulaşmanın daha da mümkün hale gelmesidir. Bu sebeple, oyunun üretmiş olduğu sonucun en optimize sonuç olduğu kesin değildir. Çünkü oyun algoritmaya girilen jenerasyon sayısı kadar ilerleyebilmekte daha sonra da sonlandırılmaktadır.

Algoritmada kullanılan mutasyon oranı da oyunun başarısını büyük derecede etkilemektedir. İki farklı mutasyon oranı ile yapılan testlerde 0.7 mutasyon oranının 0.5 mutasyon oranına göre daha başarılı olduğu gözlemlenmiştir.

Referanslar

[1] (), [Online]. Available: https : / / towardsdatascience . com / introduction-to-genetic-algorithms-including-example-code-.

[2] (), [Online]. Available: https://realpython.com/python-gui-tkinter/.

[3] (), [Online]. Available: https://selenium-python.readthedocs.io/.

[4] (), [Online]. Available: https://play2048.co/.