



BFS algoritması ile Snake Game

UMUR ABDULLAH CAN B200109042-ENES DURAN B200109040

BFS algoritması ile Snake Game

Projenin Özeti:

Breadth-First Search (BFS) Genişlik Öncelikli Arama algoritmasıdır. Bir graf veya ağaç yapısı üzerinde keşif yapmak için kullanılan bir algoritmadır. Genellikle en kısa yol bulma problemlerinde veya graf tabanlı oyunlarda kullanılır.

Snake Game, yılanın yemleri yiyerek büyüdüğü ve kendini çarpmadan hareket ederek oynanan basit bir oyundur. Oyun alanı genellikle bir kare matriste temsil edilir ve yılanın hareketleri de matristeki koordinatlar üzerinden gerçekleştirilir.

BFS algoritması, Snake Game projesinde yılanın hareketlerini yönetmek için kullandık. Başlangıç noktası olarak yılanın baş pozisyonu ele aldık ve bu noktadan başlayarak çevresindeki tüm komşu hücreleri ziyaret etmesini istedik. Ardından bu komşu hücrelerden başlayarak diğer komşuları ziyaret eder ve bu şekilde yılanın ilerleyeceği yolu belirledik. BFS, bir ağaç yapısı üzerinde seviye seviye ilerleyip bir düğümün tüm komşularını ziyaret ettikten sonra bir sonraki seviyeye geçer.

Snake Game projesinde BFS algoritması, yılanın yemlere ulaşmak için en kısa yolu bulmak için kullandık. Yılanın başlangıç noktasından yemlere doğru BFS algoritması uyarladık ve en kısa yolu belirledik. Bu yol, yılanın hareket etmesi gereken yolu gösterir. Her adımda yılanın hedefine doğru ilerlemesi sağlanır ve yemi yediğinde yılan büyür. Yılanın kendine çarpmadan veya oyun alanından çıkmadan en fazla yem yemesi hedeflenir.

Projede verilen görevin açıklanması ve gerçekleştirme adımları :

BFS algoritması Snake Game projesinde aşağıdaki adımlarla uyguladık:

- 1. Oyun alanı bir graf olarak temsil edilir, her hücre bir düğüm olarak kabul edilir. Başlangıç düğümü yılanın başının konumu olur.**
- 2. Yemler hedef düğümler olarak kabul edilir.**
- 3. BFS algoritması kullanılarak, yılanın başlangıç düğümünden hedef düğüme en kısa yol bulunur.**
- 4. Yılanın bir sonraki adımını belirlemek için en kısa yol üzerindeki bir düğüm seçilir. Bu genellikle başlangıç düğümüne en yakın olan düğüm olacaktır.**
- 5. Yılan, seçilen düğüme doğru hareket eder ve bir adım ilerler.**
- 6. Yılanın kendine veya sınırlara çarpması kontrol edilir. Eğer çarpma gerçekleşirse, oyun biter. Aksi takdirde, yılanın büyümesi devam etmektedir.**

Kullanılan Algoritma

BFS (Breath First Search) Geniş Öncelikli Arama Algoritması

BFS (Breadth-First Search), graf tabanlı veri yapıları üzerinde kullanılan bir arama algoritmasıdır. Bu algoritma, genellikle bir grafa ait düğümleri genişlik önceliğiyle keşfeder ve en kısa yol problemlerinin çözümünde sıkça kullanılır

BFS algoritması ayrıca graf tabanlı veri yapılarının genel yapısını incelemek, bağlantıları tespit etmek ve graf içindeki düğümleri gezinmek için de kullanılabilir. Algoritma, her bir düğümün ziyaret edildiği ve düğümlerin ziyaret sırasına göre işlendiği bir dizi adımdan oluşur.

Özetlemek gerekirse, BFS algoritması, graf tabanlı veri yapıları üzerinde genişlik önceliğiyle düğümleri keşfeden bir arama algoritmasıdır. En kısa yol problemlerinin çözümünde sıkça kullanılır ve kuyruk veri yapısını kullanarak düğümleri takip eder.

Akış Şeması:

Başla

Oyun tahtası oluştur ve başlangıç durumunu ayarla

Yılanın başını ve yiyeceği yerleştir

Döngü:

Kuyruk, başlangıç konumunu içeren bir BFS kuyruğu oluştur

Ziyaret edilen hücrelerin bir haritasını tutmak için bir BFS haritası oluştur

BFS kuyruğunu başlangıç konumuyla başlat

BFS kuyruğu boş olana kadar devam et:

Mevcut konumu BFS kuyruğundan al

Eğer mevcut konum yiyecekse:

Yılanın boyunu artır

Yiyeceği rastgele bir yere yerleştir

Sonraki hamleler için geçerli konumu işaretlemek için BFS haritasında güncelle

Mümkün tüm hareketleri (yukarı, aşağı, sol, sağ) kontrol et:

Eğer hamle geçerli ve henüz ziyaret edilmemişse:

Hamleyi BFS kuyruğuna ekle

BFS haritasında hamlenin mevcut konumunu güncelle

Eğer BFS kuyruğu boşsa:

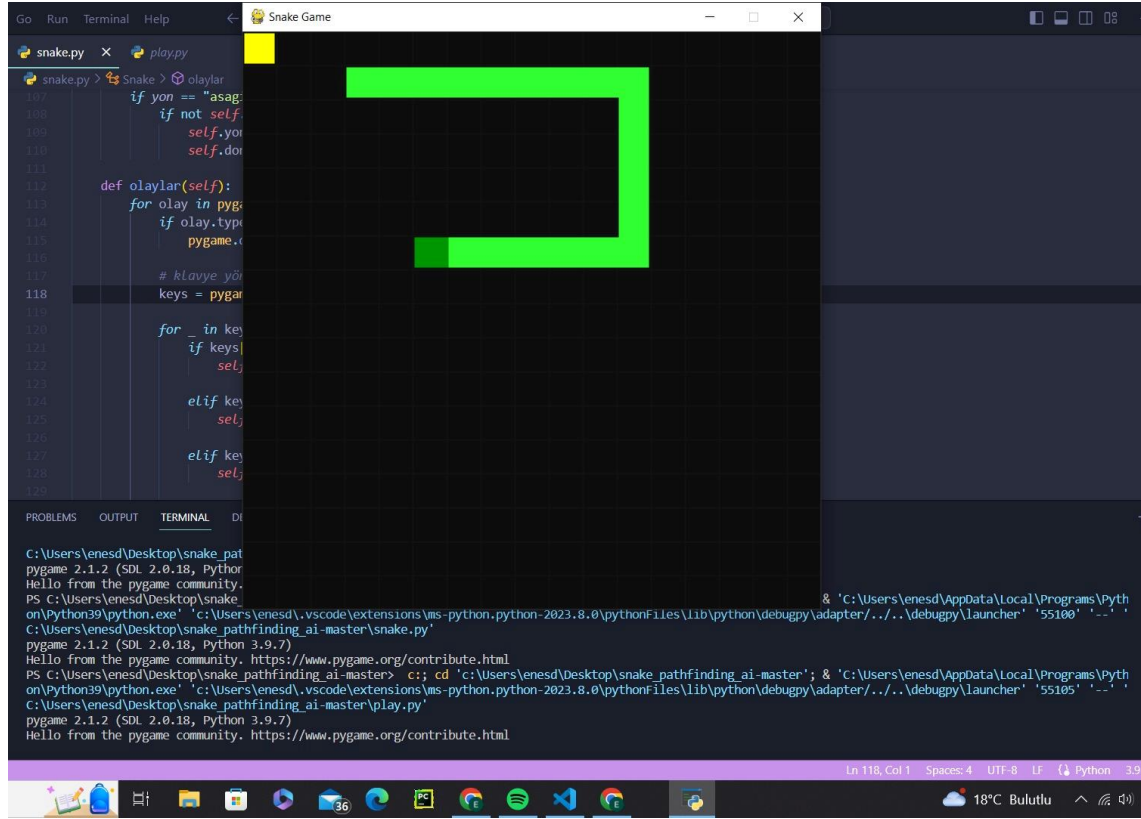
Oyunu bitir

Oyun tahtasını güncelle ve yılanın durumunu göster

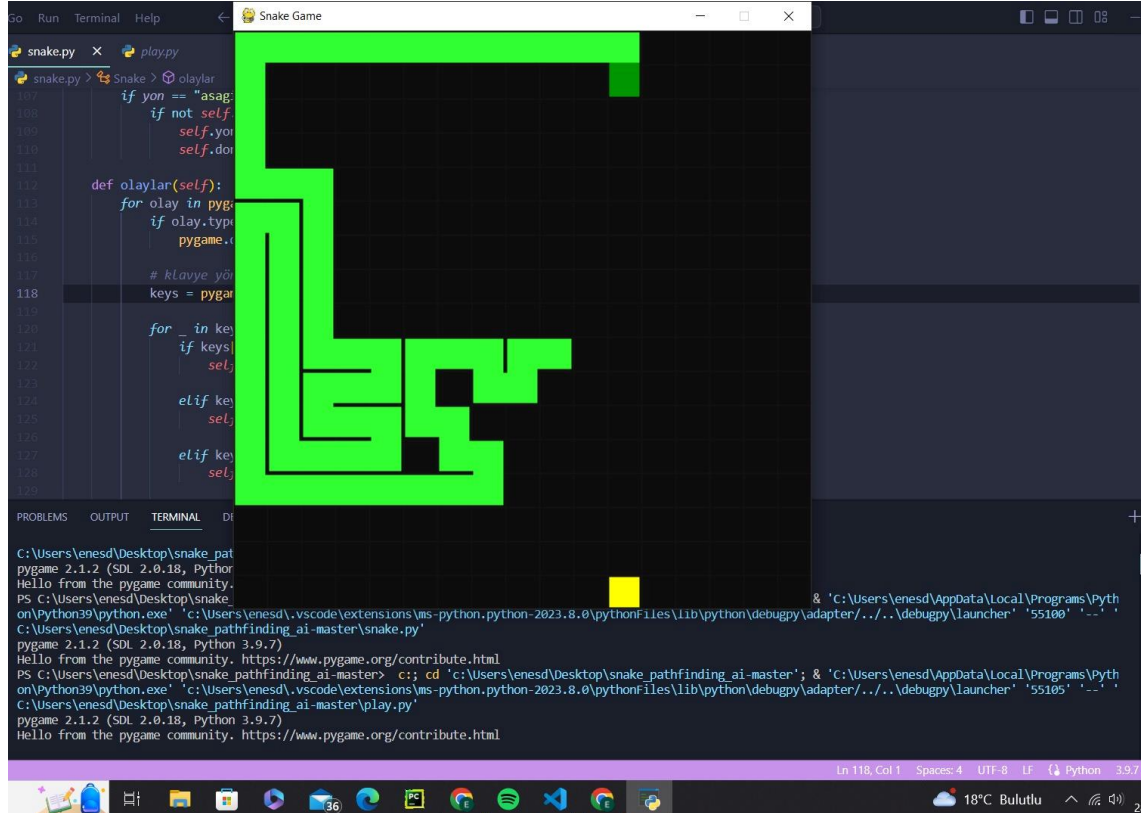
Bitir

PROJENİN EKРАН ÇIKTILARI

Oyunun başlangıç ekran görüntüsü:



Oyunun ilerlemiş ekran görüntüsü:



Oyunun bitmiş halinin ekran görüntüsü:

