

ŞİRİNLER

Oğuzhan Koç ve Yusuf Arslan

Bilgisayar Mühendisliği

Kocaeli Üniversitesi

oguzhankoc9855@hotmail.com | ysf.arslan2017@gmail.com

190201184-190201183

ÖZET

“Şirinler” adlı projemizde 2 adet iyi karakter,2 adet kötü karakter bulunmaktadır. Kötü karakterler, Azman ve Gargamel ; iyi karakterlerimizde Tembel Şirin ve Gözlükü Şirindir. Kötü karakterlerimiz grafik üzerindeki kapılardan giriş yapıp iyi karakteri Şirineye ulaşmadan yakalamaya çalışmaktadır. İyi karakterlerimiz yakalanmadan önce altın ve mantarları toplayarak skorunu yükseltmeye çalışır. Program, iyi karakterin konumuna göre kötü karakterlerin en kısa uzaklığını bulmaktadır

GİRİŞ

Şirinler projesinde kullanılmak için açılan “harita.txt” adında bir txt dosyası bulunmaktadır. Txt içinde programda grafik üzerine bastırılacak olan kötü karakterlerin isimleri başlangıç konumları ve haritanın bilgileri bulunmaktadır.

Program, bu bilgilerin ışığında grafiği oluşturup kullanıcıya hangi iyi karakterle başlamak istediğini seçtirtir. Seçim işlemi gerçekleştikten sonra iyi karakter, grafik üzerinde belirtilen sarı hücreden oyuna başlar. Kullanıcı, iyi karakterin hareketini klavyedeki yön tuşlarını kullanarak sağlamaktadır. Kapılarda yer alan kötü karakterlerin hareketleri, iyi karaktere olan en kısa mesafeleri hesaplanarak yürütülür. Bu işlemler program tarafından yapılmaktadır. Kullanıcının kontrol ettiği iyi karakter skoru 0’ın altına düşmeden şirine ulaşmalıdır aksi halde iyi karakter için oyun kaybedilmiş olunur.

2.Temel Bilgiler

Program Java programlama dilinde geliştirilmiş olup, geliştirme ortamı olarak “Eclipse IDE for Java Developers-2020-09” kullanılmıştır. Projemize başlarken ilk olarak yol haritamızı çıkardık. Projenin isterlerini analiz edip bu isterler üzerine araştırmalar yaptık.

3.Tasarım

Şirinler programının programlama aşamaları altta belirtilen başlıklar altında açıklanmıştır.

3.1.Hiyerarşik Yapı:

Şirinler Projesinde azman, gargamel sınıfları dusman; gozlukluSirin, tembelSirin ve puan sınıfları oyuncu sınıfından kalıtımlıdır. Projemizde bulunan düşman ve oyuncu sınıfları özelliklerini ata sınıfları olan karakter sınıfından almaktadır. Bunun yanı sıra hareketliliği sağlamak amacıyla locasyon sınıfımızda mevcuttur. Ayrıca grafik üzerinde gösterim yapmak için oyun ve Node sınıflarımızda bulunmaktadır.

locasyon(class):Koordinat düzlemine göre x ve y koordinatlarını tutan,Matris sistemine göre satır sütunları tutar. En kısa yolun hesaplamasında yardımcı değişkenler tanımlanmıştır.

karakter(class):Bu sınıfta dusman ve oyuncu adlı iki temel sınıf bulunmaktadır.Bu sınıflardan dusman azman ve gargamel sınıflarını, oyuncu tembelSirin ve gozlukluSirin sınıflarını bulundurmaktadır.Ad,tür,koordinat değerleri gibi değişkenlere sahiptir.Constructor,Get,Set ve enkısayol metotları yer almaktadır. Zaman Karmaşıklığı: $O(n^2)$

düşman(class):Gargamel ve azman sınıflarını bulunduruyor olup karakter sınıfından kalıtımolarak ata sınıfıdır.Constructor,Get,Set ve enkısayol metotları bulunmaktadır.

oyuncu(class): Gargamel ve azman sınıflarını bulunduruyor olup karakter sınıfından kalıtımolarak ata sınıfıdır.Constructor,Get,Set ve enkısayol metotları bulunmaktadır.

oyun(class):Dosya okuması ,oyunu çalıştırılması yürütülmesi ve hesaplamaların yapılması bu sınıfta yapılmıştır. Oyunun grafik işlemleri bu sınıfta gerçekleştirilmiştir.Zaman Karmaşıklığı: $O(n^4)$

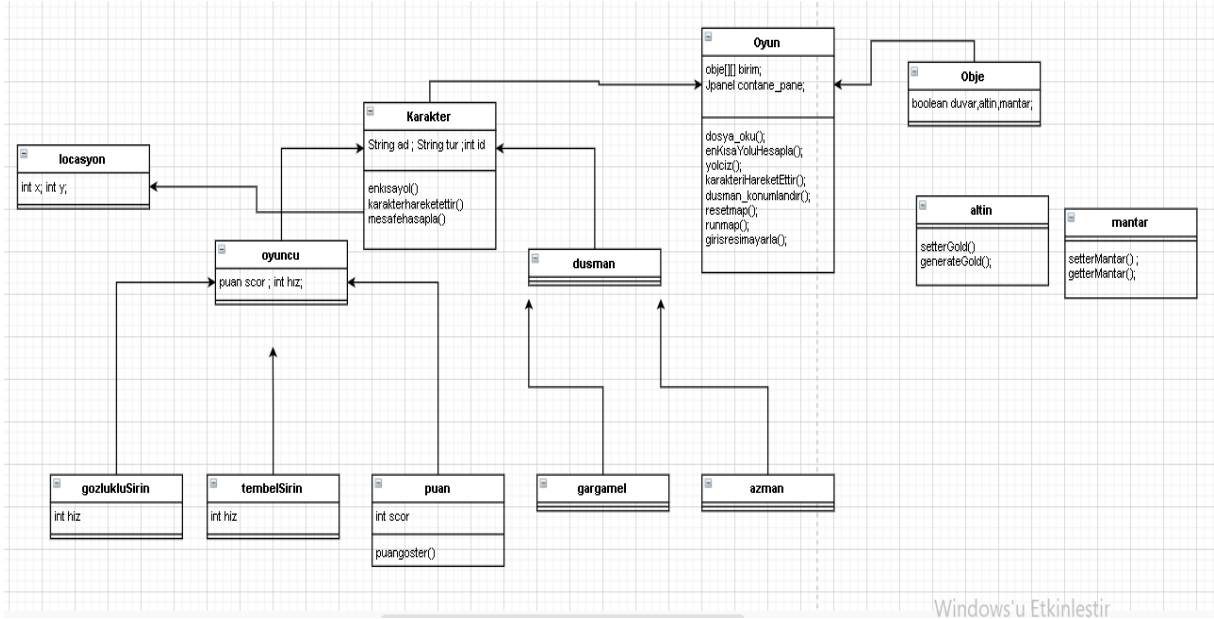
obje(class):altın ve mantar adlı iki sınıf bulunmaktadır. .Constructor,Get,Set metotları bulunmaktadır.

4.Karşılaşılan Sorunlar ve Çözümleri

4.1.Dinamik Arayüz Kullanımı:Arayüz üzerine devamlı hareket olacağı için dinamik bir arayüz yapmak istedik.Projeye başlamadan önce java ile yapmış benzer projeleri inceledik.

4.2.En Kısa Yolu Bulma Algoritması:Projenin isterleri doğrultusunda bizden istenen Djikstra Algoritmasını araştırmaya başladık.Karşılaştığımız sorun djikstrayı grafik ve karakterin hareketine uyarlamaktı.Bunun için benzer projeleri inceledik ve djikstra hakkında araştırmalar yaptık.Bu bilgilerin ışığında projemizi tamamlamış olduk.

5.UML Diyagramı



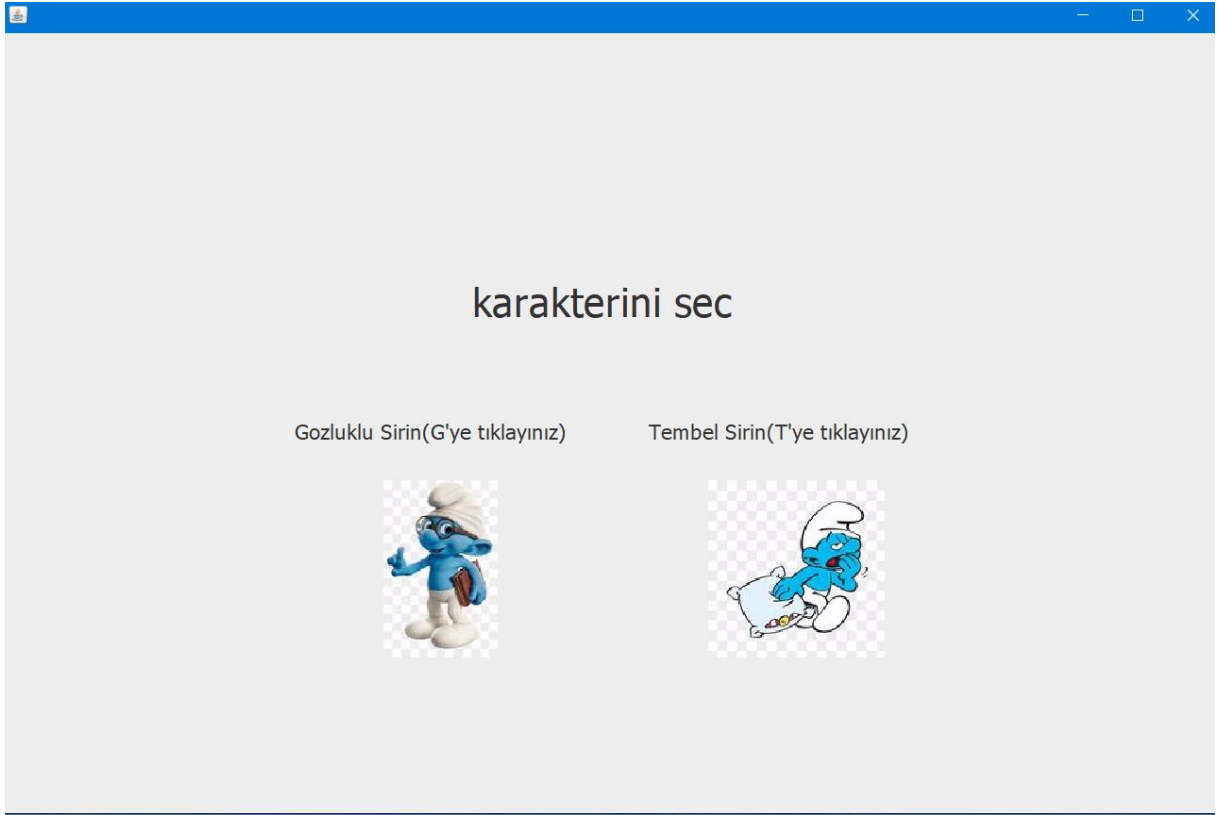
6.Pseudo(sözde) Kod

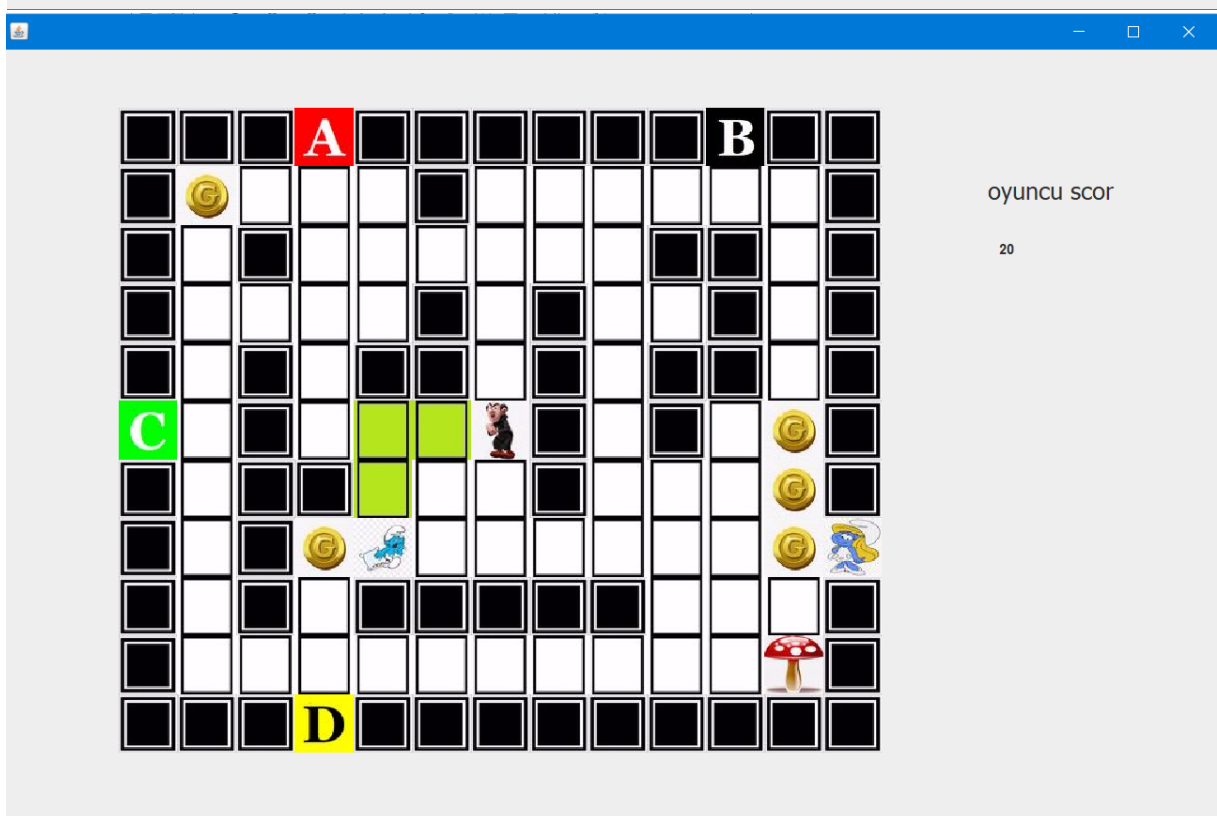
- 1-Oyun çalıştırılır.
- 2-harita.txt dosyasından bilgiler alınır.
- 3-dusman karakterlerin bilgileri tutulur.
- 4-Harita bilgileri alınır.
- 5-Haritadan alınan bilgilere göre haritadaki yol ve duvarlar konumlandırılır.
- 6- Duman karakterleri konumlandırılması yapılır.
- 7-Klavyeden oyuncu seçimi yapılır.
- 8-Haritada gold ve mantar konumlandırılması yapılır.
- 9-oyuncu konumlandırılır.
- 10-Klavyeden alınan bilgilere göre oyuncu hareket ettirilir.
- 11-Düşman karakterler ve oyuncu arasında en kısa yol hesaplanır.
- 12-Düşman hareket ettirilir.
- 13- Düşman karakterler ve oyuncu arasında en kısa yol çizilir.
- 14-Oyuncunun geçtiği konumlarda altın veya mantar var olup olmadığı kontrol edilir.
- 15-Eğer oyuncun geçtiği yerlerde altın veya mantar varsa oyuncunun skoru arttırılır.
- 16-Düşman eğer oyuncuyu yakalarsa puan düşülür.
- 17-Skor 0'ın altına düşerse oyuncu oyunu kaybeder.
- 18-Ekrana oyunu kaybettiniz şeklinde yazı gelir.

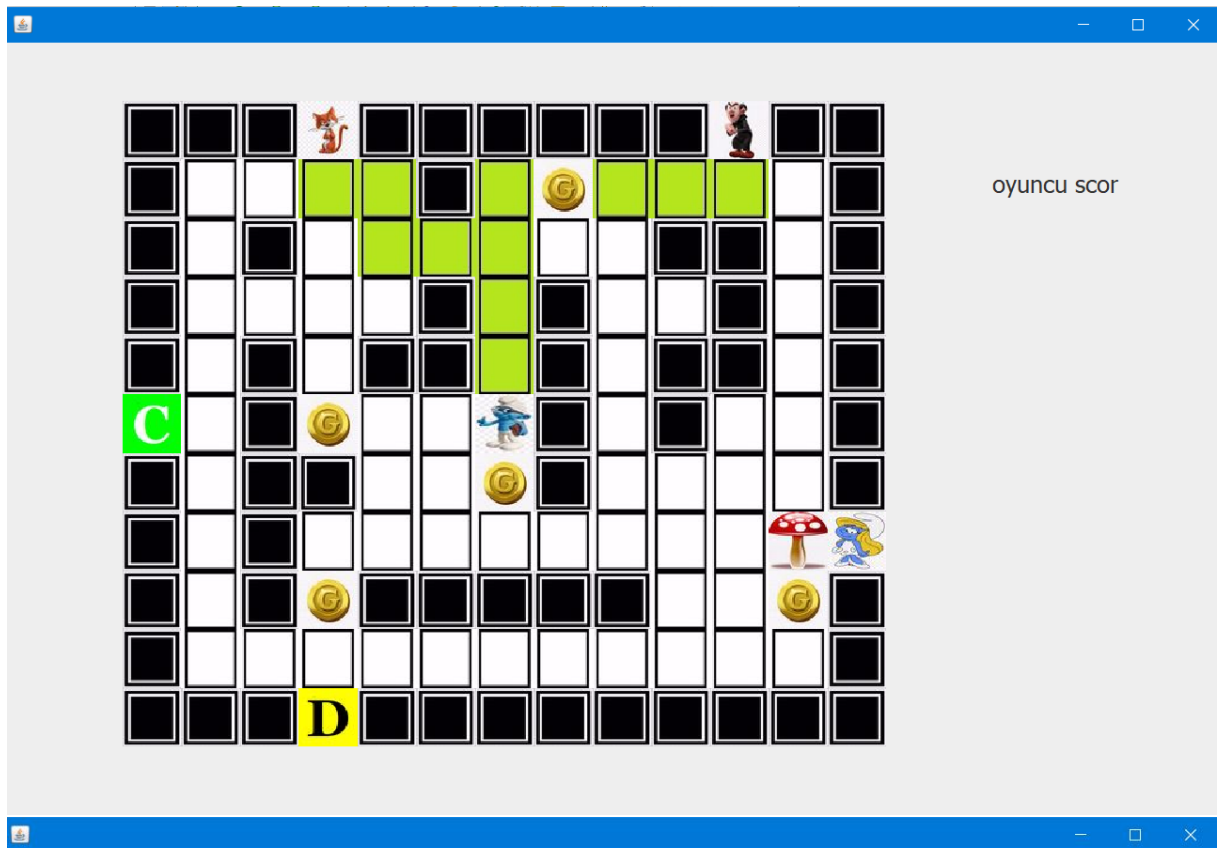
19-Eğer oyuncu şirineye ulaşırsa oyunu kazanır.

20-Ekrana oyunu kazandınız yazar.

7-Ekran Çıktıları









8-Kaynakça

<https://www.programiz.com/dsa/dijkstra-algorithm>

https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm

<https://www.youtube.com/watch?v=eVV6p0axFYc>

<http://bilgisayarkavramlari.com/2010/05/13/dijkstra-algoritmasi-2/>

https://www.w3schools.com/java/java_files_read.asp