

# KOCAELİ ÜNİVERSİTESİ

## BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

### Programlama Laboratuvarı II

#### ŞİRİNLER

*Ali Kutay Bilgilioglu*

[200202108@kocaeli.edu.tr](mailto:200202108@kocaeli.edu.tr)

#### *Projenin Özeti*

Programlama Laboratuvarı II dersi 1. projesi Şirinler' de bizden beklenen Dijkstra algoritması kullanarak bir labirent oyunu tasarlamamızdır.

Bu oyunu tasarlamak için Java programlama dili ve Netbeans geliştirme ortamını seçtim.

Bu oyunda 2 düşman karakter, 1 Şirine ve 1 seçilen Şirin vardır. Şirine labirentin çıkışıdadır ve puanı bitmeden Şirineye ulaşan Şirin oyunu kazanmış olur. Şirin Şirineye ulaşmaya çalışırken gargamel ve azman Dijkstra algoritması aracılığıyla Şirini takip etmeli, yakalamaya çalışmalıdır ve en kısa yol harita üzerinde gösterilmelidir.

Şirin haritada dolaşırken rastgele yerlerde altınlar ve mantarlar oluşur ve bunlara değen şirin puan kazanması beklenmektedir.

Bu oyunda nesneye yönelik programlama yeteneklerimizi ve veri yapıları hakkında bilgilerimizi geliştirecek ve göstereceğiz.

#### 1. GİRİŞ

Proje için Java programlama dilini ve NetBeans geliştirme ortamını kullandım.

Java programlama dili, açık kaynak kodlu, nesneye yönelik, zeminden bağımsız, yüksek verimli, çok işlevli, yüksek seviye, adım adım işletilen (yorumlanan) bir dildir. Java uygulamaları bilgisayar mimarisine bağlı olmadan herhangi bir Java Sanal Makinesi (JVM) üzerinde çalışabilen tipik bytecode' dur .

NetBeans, Oracle tarafından geliştirilen bir Java geliştirme ortamıdır ve ücretsiz olarak dağıtılmaktadır. Özellikle kullanıcı arayüzü tasarımında sağladığı kolaylıklardan dolayı tercih edilmektedir.

Swing, Java için bir GUI widget araç setidir. Arayüz tasarımı için birçok imkan sunar.

#### 2. YÖNTEM

##### **Kaba Kod**

- 1.Başla.
- 2.Dosyayı oku
- 3.harita.txt'den alınan verileri char-

Dizisine ve JLabel dizisine aktar.  
4.Değer 0 ise kırmızıya 1 ise maviye boya.  
5.Labirenti oluştur.  
6.Kullanıcıya butonlar araçlığı ile Seçmek istediği karakteri sor.  
7.Oyunu başlat.  
8.Oyuncudan komut al, bastığı tuş yönünde  
Oyuncuyu temsil eden JLabel'ın konumunu  
değiştir.  
9.Oyuncu duvarla karşılaşırsa If kontrolü ile  
labirentten çıkmasını ve duvarlardan geçmesini  
engelle.  
10.Oyuncu ve kötü karakterin konumlarını  
en kısa yolu bulmak üzere fonksiyona yolla.  
11.En kısa yolu komşuluk matrisini oluşturduktan  
sonra, komşuluk matrisine göre bul.  
12.Çıkan yol adlarını yeni oluşturulan ve içinde  
labirentin bilgisi bulunan diziye 'G' şeklinde aktar.  
13.İçinde en kısa yol bilgisini tutan char dizisini fonksiyona gönder.  
14.Fonksiyonda en kısa yola bakarak labellardan  
oluşan haritayı boya.  
15.Gargameli char dizisi kontrolü ile yeşile  
Boyanmış yol üstünde hareket ettir.  
16.Her hareket edişte oyuncu sırasını değiştir.  
17.Oyuncu ile labirentte oluşturulan altınlar  
aynı konuma gelirse skoru 5 puan arttır;  
mantar ile aynı konuma gelirse 50 puan arttır.  
18.Skor 0'a eşitse oyunu bitir.

### 3. SONUÇ

Bu projede nesneye yönelik programlamayı daha net anlamış oldum.

Hayattaki somut şeyleri kod üzerinde uygulayabileceğimi hissettim.

Veri yapıları konusunda Dijkstra algoritması üzerinde çalışırken yeni bilgiler öğrendim ve bu bilgileri kullanarak somut şeyler yaratmak güzel bir deneyim oldu.

### 4. KAYNAKÇA

- 1) [https://tr.wikipedia.org/wiki/Java\\_\(programlama\\_dili\)](https://tr.wikipedia.org/wiki/Java_(programlama_dili))
- 2) <https://tr.wikipedia.org/wiki/NetBeans>
- 3) <https://www.algoritmauzmani.com/algoritmalar/dijkstra-algoritmasi-nedir-dijkstra-ornekli-anlatim-c-kodu/>
- 4) [http://www.zafercomert.com/Medya/2015\\_05\\_11\\_2\\_121\\_69f9a888.pdf](http://www.zafercomert.com/Medya/2015_05_11_2_121_69f9a888.pdf)
- 5) [https://tr.qaz.wiki/wiki/Dijkstra's\\_algorithm](https://tr.qaz.wiki/wiki/Dijkstra's_algorithm)
- 6) <https://www.softwaretestinghelp.com/dijkstras-algorithm-in-java/>
- 7) <https://www.geeksforgeeks.org/java-program-for-dijkstras-shortest-path-algorithm-greedy-algo-7/>
- 8) <https://www.youtube.com/watch?v=q3yKyE19OR0&t=1056s>
- 9) <https://www.youtube.com/watch?v=p9Y-NBg8eto>
- 10) <https://www.youtube.com/watch?v=jT3c45XkPTg>