

# KOCAELİ ÜNİVERSİTESİ

## BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

### PROGRAMLAMA LAB.2 – 1.PROJE

## ŞİRİNLER

MERTCAN KUŞCU 190202082@kocaeli.edu.tr

SİNAN TOPAL 190202035@kocaeli.edu.tr

#### Projenin Özeti

Programlama Laboratuvar 2- 1. Projesi olarak bizden “Şirinler” adındaki bir çalışmayı yapmamız beklenmektedir.

Biz proje için Java Programlama Dilini ve NetBeans geliştirme ortamını seçtik.

Projede biz Java Programlama Dilinde bulunan Swing kütüphanesini kullandık. Swing kütüphanesi sayesinde görsel arayüz tasarımını yapmış olduk.

Projemizde ilk olarak bize ekte verilen pdf dosyasında belirtilmiş sınıfları kendi isimleriyle ekledik. Daha sonra bu sınıfların içini bizden istenildiği gibi getter ve setter metotlarıyla doldurduk. Main sınıfımız olarak “Main” adında yeni bir class açtık. Sonra “Anamenu” isminde bir JFrame açarak oyunumuzun giriş ekranını tamamlamış olduk. Daha sonra “Oyun” isminde yeni bir JFrame açıp oyunu bu ekran üzerinden yazdığımız kodlarla ve tasarladığımız resimlerle oynatmayı başardık. Daha sonra çıkan skor 0 veya 0’ın altına indiğinde oyun bilgisayar tarafından kazanılıp oyun bitmiş olarak sayılacaktır, ancak seçilen şirin skoru 0 veya 0’ın altına düşmeden şirineye ulaşırsa oyun kullanıcı tarafından kazanılıp oyun bitmiş olarak sayılacaktır.

#### Giriş

Proje için Java programlama dili ve NetBeans geliştirme ortamını kullandık.

Java; basit, nesne odaklı, ağ anlayışlı, yorumlanmış, sağlam, güvenli, taşınabilir, tarafsız mimarisi, yüksek performanslı ve dinamik bir bilgisayar dilidir.

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

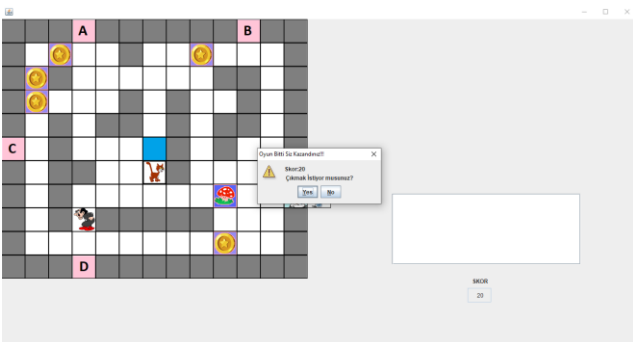
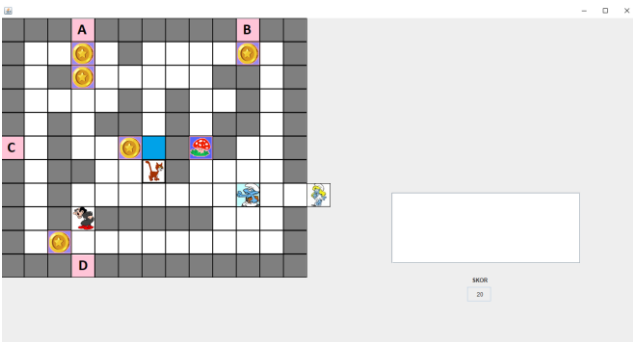
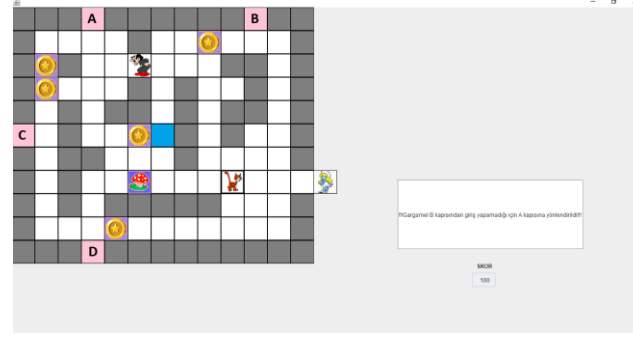
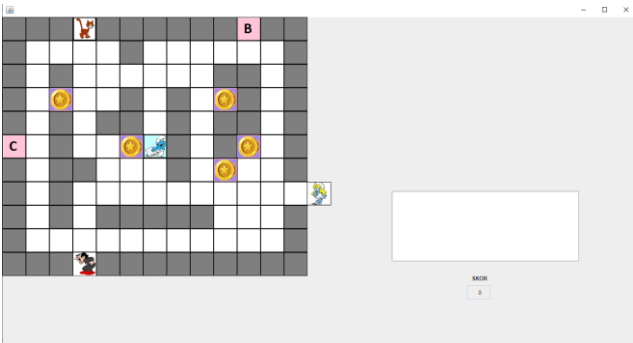
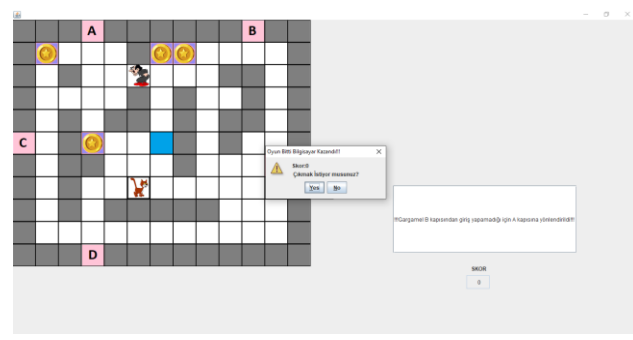
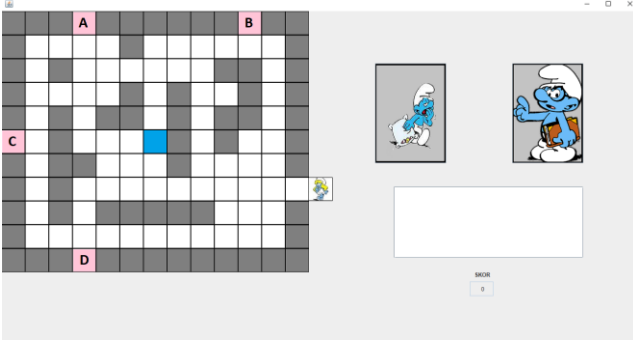
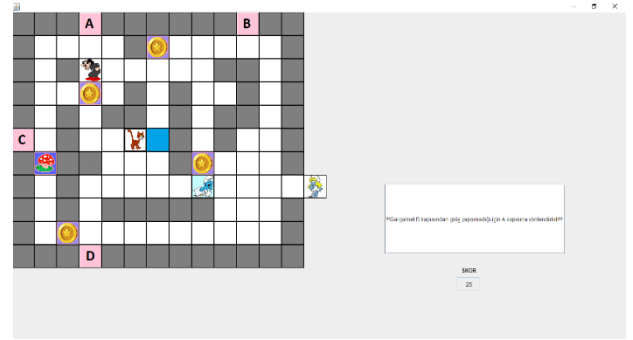
Projemizde söylenen problem Şirinler Oyunu problemi olarak adlandırılmaktadır.

Projede bizden oyunu ve oyunun takip edilebileceği bir arayüz tasarlamamız beklenmektedir. Oyunda ilk olarak şirin seçmemiz istenmektedir. Şirini seçtikten sonra klavye yardımıyla hareket ettirerek şirineye ulaşmak oyunun asıl amacıdır. Oyun esnasında düşman olarak Gargamel ve Azman şirininin yolunda engel olarak görülmektedir.

#### Yöntem

Projemize ilk olarak gerekli tüm sınıfları ekledik. Daha sonra bu sınıfların içerisine istenilen constuctorları getter ve setter metotlarıyla ekledik. Bizden istenilen şekilde bazı sınıfları diğer sınıflardan kalıtım olacak şekilde tasarladık, gereken yerlerde de override ettik. Karakter sınıfını en temel sınıf olacak şekilde hazırlayarak oyuncular ve düşman karakterlerin bu sınıftan türemesini sağladık. Lokasyon sınıfını açarak x ve y koordinatlarını tutan iki farklı değişken tutturduk. Main sınıfını açarak oyunun arayüzünü başlamasını buradan başlatmayı sağladık. Obje sınıfını açarak altın ve mantar objelerinin bu sınıftan türemesini sağladık. Oyuncu sınıfını açarak karakter sınıfından kalıtım almasını sağladık. Anamenu ve Oyun isimli iki adet JFrame classı açarak oyunumuzun görsel kısmını bu sınıflarda yürütmeyi başardık. Tüm sınıflarımızda kendilerine ait gerekli getter ve setter metotlarını da yazdık.

## Deneyisel Sonuçlar



## Sonuç

Projemizin en kısa yol sorunu dışında diğer özellikler bakımından sorunsuz çalıştığını görmüş olduk.

Bu proje sayesinde Java dilinde arayüz kütüphanesi olan Swing kullanmayı ve işlevlerini öğrenmiş olduk.

Swingde yer alan JFrame formunu kullanmayı öğrendik, arka plana nasıl fotoğraf atayacağımızı, eklediğimiz butonlara nasıl komut verebileceğimizi öğrenmiş olduk.

Kodumuzun içerisinde görüntüyü klavye yardımıyla nasıl hareket ettirebileceğimizi ve bunun sonucunda nasıl çeşitli emirler verebileceğimizi öğrenmiş olduk.

## Yalancı Kod

1 Class Karakter oluştur;

2 Class Oyuncu oluştur;

3 Tembelsirin alt sınıfını oluştur;

4 GozlukluSirin alt sınıfını oluştur;  
5 Puan alt sınıfını oluştur;  
6 Class Dusman oluştur;  
7 Gargamel alt sınıfını oluştur;  
8 Azman alt sınıfını oluştur;  
9 Class Lokasyon oluştur;  
10 Class Obje oluştur;  
11 Altın alt sınıfını oluştur;  
12 Mantar alt sınıfını oluştur;  
13 JFrame Anamenu oluştur;  
14 JFrame Oyun oluştur;  
15 Class Main oluştur;  
16 Main sınıfından arayüzü aç;  
17 Oyuncu sınıfında PuaniGoster fonksiyonu oluştur;  
18 Bu fonksiyonda puanları tut;  
19 Anamenu JFrame' ine oyunun girişinde gözükecek label'a fotoğraf ekle;  
20 Start butonu oluştur;  
21 Start butonuna basıldığında Oyun JFrame ekranının açılmasını sağla;  
22 Oyun JFrame ekranına dosyadan girilen haritayı ilgili konumlara yerleştir;  
23 Oyun JFrame ekranından oyunda gösterilecek olan şirini butonlar sayesinde seçtir;  
24 Seçilen şirini haritanın orta yerine yerleştir;  
25 Dosyadan girilen düşmanı veya düşmanları ilgili kapılara yerleştir;  
26 Oyun skorunu girilen puan olarak başlat;  
27 Klavye yardımıyla şirini harita üzerinde hareket ettir;  
28if(gargamel.getBounds().equals(ortadaki.getBounds())){  
skor=skor-15; }  
29if(azman.getBounds().equals(ortadaki.getBounds())){  
skor=skor-5; }  
30 Haritada girilen süreler içerisinde rastgele yerlere mantar ve altın objelerini yerleştir;  
31 if(mantar.getBounds().equals(ortadaki.getBounds())){  
skor=skor+50; }

32 if(altın.getBounds().equals(ortadaki.getBounds())){  
skor=skor+5; }  
33 Puan 0 veya 0'ın altında düştüğünde oyunu bilgisayar kazandı olarak yansıtarak bitirme ekranını getir;  
34 Şirin şirineye sorunsuz şekilde ulaştığında oyunu kullanıcı kazandı olarak yansıtarak bitirme ekranını getir;  
35 Bitir;

## Yaklaşımlar

Programlama Laboratuvar 2- 1. Projesi olarak bizden “Şirinler” adındaki bir çalışmayı yapmamız beklenmektedir.

Projede biz Java Programlama Dilinde bulunan Swing kütüphanesini kullandık. Swing kütüphanesi sayesinde görsel arayüz tasarımı yapmış olduk.

Projemizde ilk önce görsel tasarımlarımızı tamamlamayı, ondan sonra da kod kısmına dönerek projemize devam etmeyi planlamıştık. Planlarımızı adım adım uygulayarak projemizin sonucuna yaklaştık.

## Kazanımlar

Bu proje sayesinde Java dilinde arayüz kütüphanesi olan Swing kullanmayı ve işlevlerini öğrenmiş olduk.

Swingde yer alan JFrame formunu kullanmayı öğrendik, arka plana nasıl fotoğraf atayacağımızı, eklediğimiz butonlara nasıl komut verebileceğimizi öğrenmiş olduk.

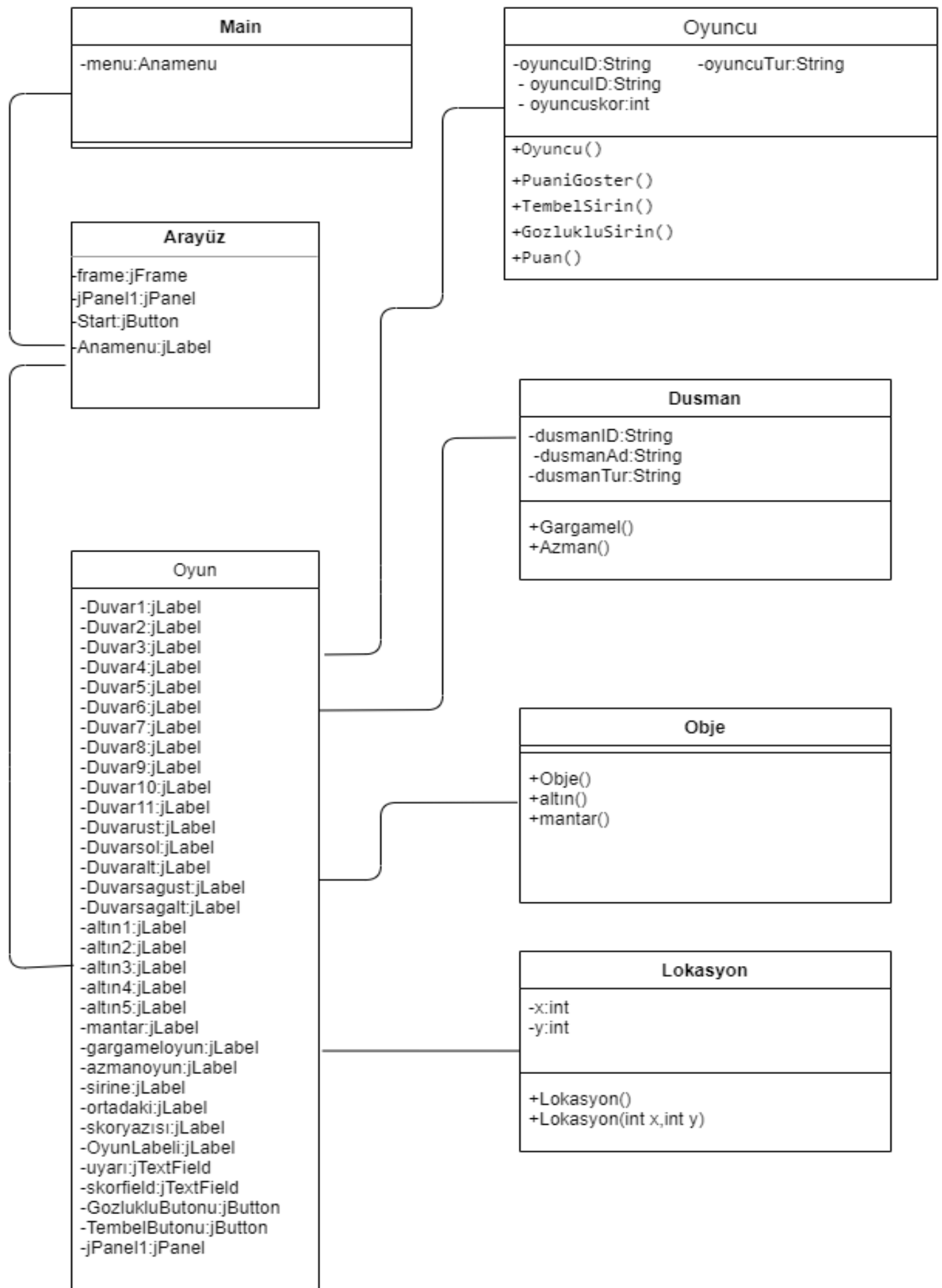
Kodumuzun içerisinde görüntüyü klavye yardımıyla nasıl hareket ettirebileceğimizi ve bunun sonucunda nasıl çeşitli emirler verebileceğimizi öğrenmiş olduk.

## Eksiklikler

Bu projemizde kullanılması gereken “Dijkstra Algoritması” nı biz amacını ve yöntemini hem görsel olarak hem de kod üzerinde olarak çok iyi şekilde anladık ancak bize verilen haritanın üzerinde bu algoritmayı nasıl kullanabileceğimizi tam olarak başaramadık.

Algoritmanın mantığını iyi anladığımızdan dolayı raporumuza bu algoritmanın kullanıldığı bir örneğin zaman ve bellek karmaşıklığı hesabını ekledik. Ancak kodumuzda haritaya uyarlayamadığımız için gösteremedik.

## UML SINIF DIAGRAMI



## Algoritmanın Bellek Karmaşıklığı Hesabı

Biz “Dijkstra Algoritması” nın mantığını anladık. Koda yansıtma şeklini de anladık ancak projede bize verilen haritaya bu algoritmayı nasıl yansıtacağımızı bulamadık. Bundan dolayı bizim farklı bir çalışmada aynı mantıktaki kodları kullandığımız aşağıdaki linkteki koda göre zaman karmaşıklığı hesabını yaparak rapora eklemeyi doğru bulduk.

<https://algorithms.tutorialhorizon.com/djkstras-shortest-path-algorithm-adjacency-matrix-java-code/>

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Bu fonksiyon toplam zaman karmasikligi =  $n+2$ 'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Bu fonksiyon toplam zaman karmasikligi =  $n$ 'dir.

//Zaman Karmasikligi= 1'dir.

//Bu fonksiyon toplam zaman karmasikligi  $n.n+4= n^2+4$ 'tür.

//Zaman Karmasikligi= 1'dir.

//Bu fonksiyon toplam zaman karmasikligi =  $n$ 'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

//Zaman Karmasikligi= 1'dir.

/\*Zaman Karmaşıklığı=  $n^2+3n+31$

BigO Notasyonu  $O(n^2)$  olarak bulunur.\*/

## Kaynakça

[https://www.youtube.com/watch?v=O76IT\\_H4uxY&t=589s](https://www.youtube.com/watch?v=O76IT_H4uxY&t=589s)

<http://www.yasinhoca.com/2018/01/basit-oyun-haritas-yapm.html>

<https://www.youtube.com/watch?v=6GsDX6g9ELQ>

<https://www.youtube.com/watch?v=ILBns7jPl2U&list=PLzIWkToFwqHSZuhkX5NEojR7ZMWE1XQu7&index=47>

<https://www.youtube.com/watch?v=Qnjd7pTNJxY>

<https://www.youtube.com/watch?v=4fuOe2D8cRY>

<https://www.youtube.com/watch?v=7wJC2UG3dlO>

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

<https://www.youtube.com/watch?v=bImvsFNym9I>

<https://www.youtube.com/watch?v=SN-FYlvtwOE>

<https://mcaliskanyurekblog.wordpress.com/2018/04/02/en-kisa-yol-problemi-dijkstra-algoritmasi/>

[https://www.youtube.com/watch?v=Za7\\_S8vhd-8](https://www.youtube.com/watch?v=Za7_S8vhd-8)

<https://algorithms.tutorialhorizon.com/djkstras-shortest-path-algorithm-adjacency-matrix-java-code/>

MERTCAN KUŞCU [190202082@kocaeli.edu.tr](mailto:190202082@kocaeli.edu.tr)

SİNAN TOPAL [190202035@kocaeli.edu.tr](mailto:190202035@kocaeli.edu.tr)