

- 1- Bir frame aç
- 2- Oyun 1 mi Oyun 2 mi seçeneğini sor
- 3- Eğer Oyun 1 seçilirse Izgara 1 mi Izgara 2 mi sor
- 4- Eğer Izgara 1 Seçilirse UrlOku metodunu çağır
- 5- Verilen Url adreslerinden ilkinde git ve buradaki verileri satır satır oku ve bir stringe at.
- 6- Okunan verileri int'e cast et ve bir matrise ata.
- 7- Eğer Izgara 2 seçilirse UrlOku metodunu çağır.
- 8- Verilen Url adreslerinden ikincisine git ve buradaki verileri verileri satır satır oku ve bir stringe at.
- 9- Okunan verileri int'e cast et ve bir matrise ata.
- 10- Elde ettiğin matrisde engellere denk gelmeyecek şekilde rastgele başlangıç ve bitiş noktaları oluştur.
- 11- Eğer bitiş noktası ile başlangıç noktası aynı yere denk gelirse veya bir engele denk gelirse tekrardan atama yap.
- 12- Elde ettiğin matris, başlangıç ve bitiş noktalarını YolAra metoduna yolla.
- 13- Robot sınıfından bir nesne üret ve bu robotun x ve y koordinatlarına başlangıcın x ve y koordinatlarını ata.
- 14- YolAra metodunda Robotun önce sağını kontrol et eğer boşsa metodu robotun y konumunu bir arttıracak şekilde yeniden çağır
- 15- Sonra solunu kontrol et eğer boşsa metodu robotun y konumunu bir azaltacak şekilde yeniden çağır
- 16- Sonra Yukarısını kontrol et eğer boşsa metodu robotun x konumunu bir arttıracak şekilde yeniden çağır
- 17- Sonra aşağısını kontrol et eğer boşsa metodu robotun x konumunu bir azaltacak şekilde yeniden çağır.
- 18- En sonunda eğer robotun konumu bitiş konumuna eşitse metodu bitir .
- 19- Bu adımları ArrayListe ekle ve Robotun gittiği yolu oluştur.
- 20- YolAra fonksiyonda oluşan yolu graphics metodunda göstermek için bir for döngüsü aç ve bu döngüyü iki iki arttır bu şekilde Robotun gittiği yolu tutan ArrayListin verilerine x ve y koordinatları olarak ulaş.
- 21- Robotun gittiği Yolu tutan ArrayListte bulunan yerlere (x ve y koordinatlara) ok koy ve bunları beyaza boy.
- 22- Bu ArrayListteki x ve y koordinatları 1 azaltıp 1 arttırarak robotun 1 adım ötesinde bulunan engelleri siyaha serbest alanları beyaza boy.
- 23- Aynı zamanda HafızaTut metodunu çağır ve bu metotta robotun gittiği ve gördüğü yolları hafızada tut.
- 24- EnKısaYolBul sınıfından bir nesne oluştur ve EnKısaYol metodunu çağır.
- 25- Bu metoda robotun hafızada tuttuğu matris gönder.
- 26- Hücre Sınıfı oluştur satır, sütun, mesafe ve önceki değişkenlerine sahip olsun.
- 27- Hücre sınıfından hafıza matrisin boyutlarında bir matris oluştur ve mesafe değişkenine 100 ata.
- 28- Hücre sınıfından bir başlangıç oluştur ve başlangıç noktasını buna ata ve mesafe değişkenini 0 yap.
- 29- Hücre tipinde bir kuyruk oluştur ve başlangıcı buna at.
- 30- Daha sonra kuyruktan ilk elemanı al ve bu noktanın komşularını ziyaret et(sağ,sol,üst,aşağı).
- 31- Eğer geçerli noktadan daha kısa bir yol varsa kuyruğa dahil et.
- 32- En sonda en Kısa mesafeler belirlenerek bir ArrayList oluştur.
- 33- Graphic metodunda en kısa yolu göster.

- 34- Eğer oyun2 Seçilirse Kullanıcıdan boyut bilgisi al
- 35- Boyut bilgisine göre matris oluştur
- 36- Oluşturduğun matriste rastgele labirent oluştur
- 37- Labirentin sol üst köşesini başlangıç sağ alt köşesini bitiş yap
- 38- Oluşturulan labirent matrisini LabirentYol metoduna gönder ve metodu çağır
- 39- LabirentYol metodunda ilk önce robotun sağ tarafını kontrol et eğer boşsa sağa git
- 40- Sırasıyla Robotun solunu, Yukarısını, Aşağısını kontrol et ve boşsa öncelik sırasına göre git.
- 41- Robotun gittiği yerleri işaretle.
- 42- Eğer birden fazla gidecek seçenek varsa orayı stack'e ekle.
- 43- Eğer robotun gidebileceği yer kalmazsa, etrafı engel veya geldiği yer ise stackten En sonki yol ayrımını çek ve robotu oraya götür.
- 44- Tüm bu adımları yaparken bir ArrayListte robotun x ve y koordinatlarını kaydet
- 45- Bu arrayList üzerinden Graphic metodunda Robotu yönlendir ve Robotun geçtiği noktaları boyay.
- 46- Robotun geçtiği noktaların 1 adım ötesinde engel varsa siyaha engel yoksa beyaza boyay.
- 47- En sonunda robotun en kısa yolunu bulmak için Robotun tekrar geçtiği noktaları içermeyen bir ArrayList oluştur.
- 48- Bu ArrayListi Graphic metodunda boyay