

KAREKAPAN OYUNU

algorıtma ve programlama – 1 proje 2

Ömer Tarık İlhan Zembat | 05210000285 Celaldoğan Güneş | 05200000067

İçindekiler

# Programcı Kataloğu:

* Analiz
* Gerçekleştirim
* Test
* Raporlama
* Kullanılan Veri yapıları ve fonksiyonlar

# Kullanıcı Kataloğu:

* Kullanım Kılavuzu

# Programcı Kataloğu

**Analiz Süreci:**

Programın gereksinimleri belirlendi ve izlenilecek yol haritası planlandı. Ayrıca raporlama aşamasında kolaylık olması için kronometre ayarlandı. Bu süreç 1 saat 31 dakika zaman aldı.

**Gerçekleştirim Süreci:**

Programın kodları oluşturuldu. Bu süreçte PyCharm Professional Edition uygulamasındaki CodeWith Me altyapısıyla eşlenik kodlama yapıldı. Süreç içerisinde syntax hatalarını engelleme amaçlı testler yapıldı. Bu süreç 21 saat 50 dakika zaman aldı.

**Test Süreci:**

Programın yazılması bittikten sonra program sürekli test edildi. Bunun için rastgele girdi oluşturma yöntemleri kullanıldı. Çeşitli hatalar bulundu ve düzeltildi. Bu süreç 2 saat zaman aldı. Bu sürenin çoğunu “iki nokta arası boşluk mu” kontrolündeki bir mantık hatası aldı.

**Raporlama Süreci:**

Kronometre ve projenin yedeklendiği GitHub reposundaki bilgilere bakıldı ve süreler hesaplandı. Rapora eklemek için ekran görüntüleri çekildi ve görseller bulundu. Bu süreç 1 saat zaman aldı.

**Kullanılan Veri Yapıları:**

Oyundaki taşların konumlarını saklamak için tek ve iki boyutlu, oluşmuş olan karelerin verilerini saklamak için üç boyutlu listeler kullanıldı.

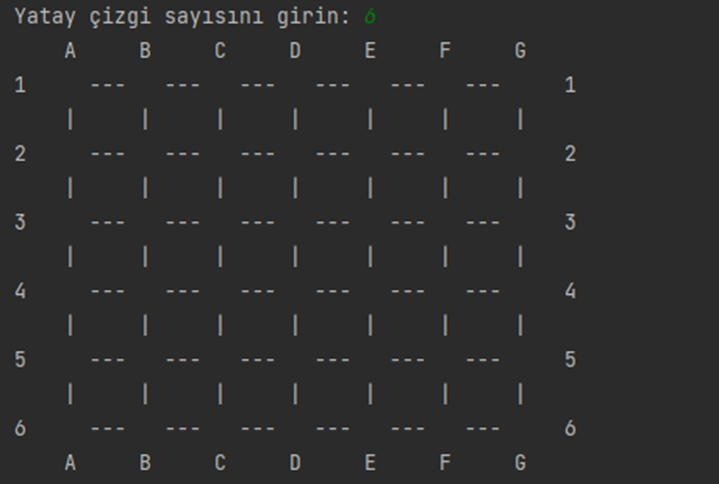
**Fonksiyonlar:**

1. **f\_haritayi\_olustur(): Yatay çizgi sayısını alıp taşların konumlarını saklayan 2 boyutlu liste oluşturur.**
2. **f\_aralik\_kontrolu(): Sayı alır ve girilen sayının girilen belirtilen sınırlar arasında olup olmadığına bakar.**
3. **f\_tas\_yerlestirme(): Oyunun ilk aşamasında Kullanıcıların sırayla haritada boş olan bir noktaya taş yerleştirmesini sağlar.**
4. **f\_konum\_bosmu(): Taş yerleştirme ve diğer oyun etaplarında kullanıcı tarafından girilen noktanın boş olup olmadığını kontrol eder.**
5. **f\_konum\_haritada\_varmi(): Girilen konumun veya konumların harita listesinin boyutlarına uygun olup olmadığını kontrol eder.**
6. **f\_metni\_sayi\_konumuna\_cevir(): String olarak alınan konumu liste indekslerine çevirir.**
7. **f\_harfi\_sayiya\_cevir(): Girilen konumdaki harfi sayıya çevirir.**
8. **f\_tas\_eleme(): Oyunun ikinci aşamasını kontrol eder. Taraflar kaç kare oluşturduysa rakiplerinden o kadar taş silmesini sağlar. Sıfır kare oluşması** **durumunda beyaz oyuncunun bir siyah taşı silmesini sağlar.**
9. **f\_tas\_sil(): Oyunun herhangi bir etabında bir oyuncu rakibinin taşını silecek olursa bu işlemi kontrol eder ve siler.**
10. **f\_oyunun\_govdesi(): Taş eleme etabından sonra kalan taş sayısını hesaplar. Oyuncuların sırayla hamle yapmalarını sağlar. Oyun bitiş koşulu sağlandıysa oyunu bitirir.**
11. **f\_hareket\_etme(): Oyuncudan konumu alır ve oyunun kurallarına göre taşların hareket etmesini sağlar.**
12. **f\_iki\_nokta\_arasi\_bosmu(): Oyuncudan alınan 2 nokta arasında taş olup olmadığını kontrol eder.**
13. **f\_kareleri\_hesaplama(): Oyun alanında oluşmuş olan kareleri hesaplar ve üç boyutlu liste olarak depolar.**
14. **f\_yeni\_kare\_olustumu(): Eski ve yeni kare verilerinin depolandığı listeleri karşılaştırır ve yenisiyle günceller.**
15. **f\_konum\_kare\_icindemi(): Girilen konum oyun alanındaki herhangi bir karenin içinde mi kontrol eder.**

# Kullanıcı Kataloğu

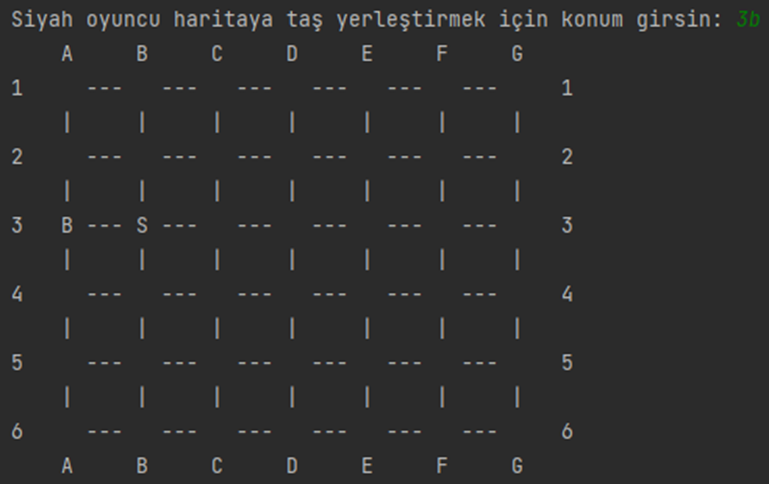
**Oyunun Oynanışı**

• Oyun alanını oluşturmak için yatay çizgi sayısı girilir. Bu sayı 3-7 arasındadır. Dikey çizgi sayısı, yatay çizgi sayısından her zaman 1 fazladır.



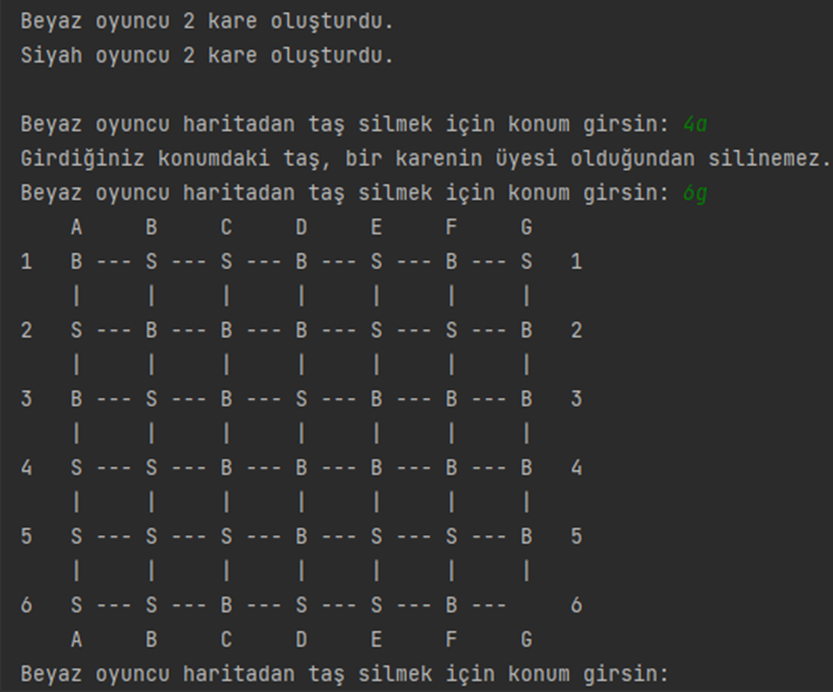
• Oyun alanı oluşturulunca taş yerleştirme etabı başlar. Oyuncular sırayla konum girerek boş olan konumlara taşlarını yerleştirir. Konum girilirken önce konumu belirten harf, sonra sayı bitişik olarak girilir.

metin, klavye, elektronik eşyalar içeren bir resim

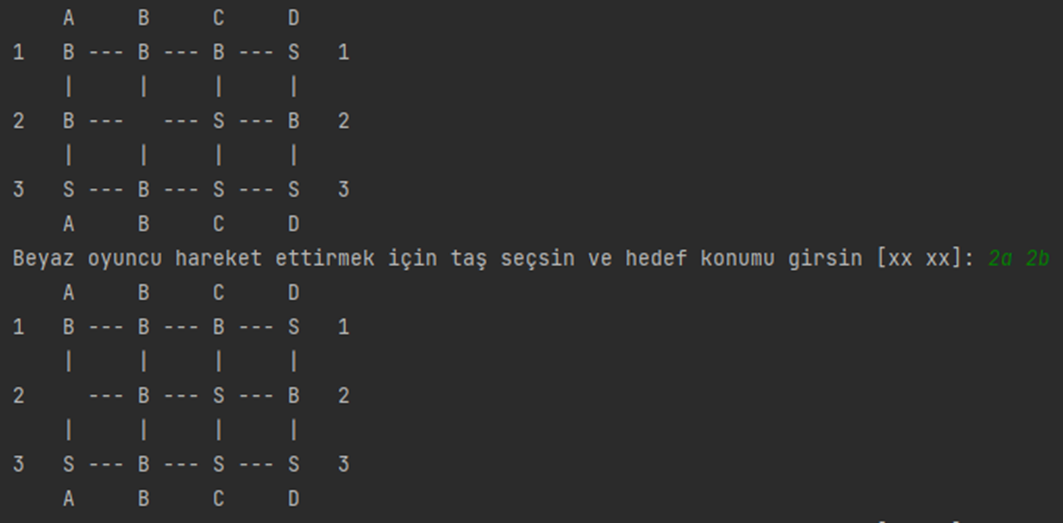
Açıklama otomatik olarak oluşturuldu

• Tüm taşlar yerleştirildikten sonra oyuncular kaç kare oluşturduklarını hesaplar ve rakiplerinin o kadar taşını siler. Önce beyaz oyuncu rakibinden silebileceği tüm taşları siler, sonra aynısını siyah oyuncu yapar. Hiç kare oluşmadıysa beyaz oyuncu, siyah oyuncunun bir taşını siler.

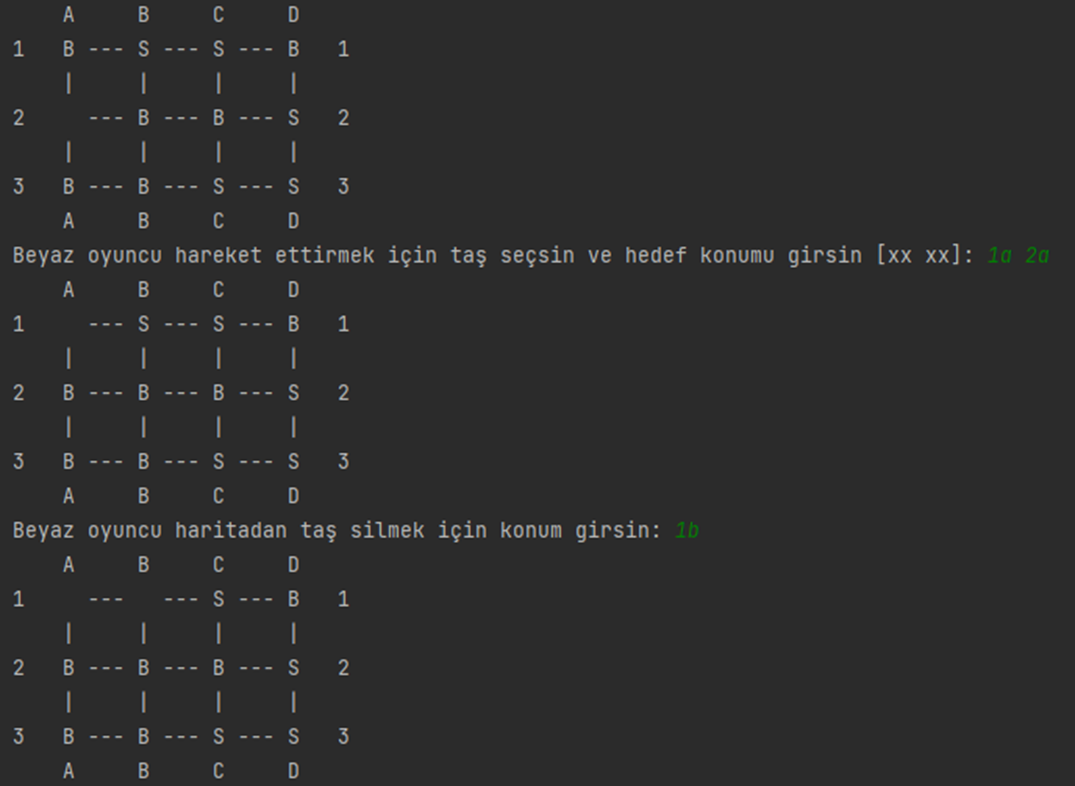
**Uyarı: Oyunun herhangi bir etabında hiçbir şart altında bir kareye ait olan taş silinemez.**

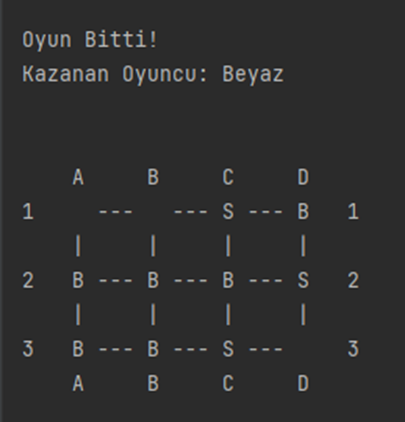


• Taş eleme aşaması bittikten sonra oyuncular sırayla taşlarını hareket ettirmeye başlar. Bu aşamada hamle yapılırken tek seferde önce hareket edilecek taşın konumu, sonra hedef konum yazılır. Taşlar dikey veya yatay yönde hareket edebilir ve hareket sırasında başka bir taşın üstünden geçemez. Ayrıca hareket edilecek hedef konum boş olmalıdır. Bu aşamada konum yazım şeklini fotoğrafta görebilirsiniz.



• Hareket sonucunda kare oluştuysa kareyi oluşturan oyuncu, rakibinin bir taşını siler. Daha önceden yazıldığı gibi, karelere ait olan taşlar hiçbir şekilde silinemez.



• Oyun, oyunculardan birinin üç taşı kalana kadar sırayla devam eder.