



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU**

**Koloni Savaşı (Java)**

**B181210076 - Muhammet Kemal Güvenç**

**SAKARYA**

**Mayıs, 2023**

Programlama Dillerinin Prensipleri Dersi

# Koloni Savaşı (Java)

Muhammet Kemal Güvenç

*B181210076 – I/C*

---

## Özet

İlk önce kullanıcıdan kolonilerin popülasyon bilgileri parametre olarak alınır. Alınan bu bilgiler doğrultusunda bütün koloniler oluşturulur. Koloniler oluşturulduktan sonra her tur hepsi birbiriyle savaştırılır. Bu savaşlar sonucu ölen koloniler yok sayılarak devam edilir. Ayrıca her turda koloniler savaştıktan sonra yenilenirler. Yenilenmekten kasıt şudur: Her koloninin kendi üretim taktiği doğrultusunda yemek üretir. Popülasyonları yüzde 20 büyür. Ayrıca popülasyonun 2 katı kadar yemek tüketirler. Bu şekilde koloniler mücadele eder. Mücadele belirli bir tur sonra biter ve sadece birisi kazanır. Bu programın en önemli özelliklerinden biri her ne kadar aynı parametreler verilse de farklı sonuçlar üretmesidir. Çünkü koloniler oluşturulurken her biri için rastgele üretim ve savaş taktiği verilir. Ayrıca iki koloninin beraber kalması ihtimal dahilinde değildir. Eğer her şeyleri aynı ise rastgele biri mücadeleyi kazanır. Bu rastgelelikler programın her defasında farklı sonuçlar üretmesine sebebiyet verir.

© 2023 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Faydalanmış olduğum kaynakları içerisinde belirttim. Her hangi bir kopya işleminde sorumluluk bana aittir.

Anahtar Kelimeler: Rastgelelik, Koloni, Mücadele, Popülasyon

---

## 1. GELİŞTİRİLEN YAZILIM

1. Uretim Arayüzü: İçinde 1 adet fonksiyon bulundurur. Bunlar şunlardır:

a. int Uret(): Üretim tekniği için kullanılır.

2. AUretim ve BUretim Sınıfları: Her iki sınıfta da 1'er fonksiyon bulunmaktadır. Her iki sınıf da Uretim yapısından kalıtım almaktadır. İçlerindeki fonksiyonlar şunlardır:

a. public int Uret(int populasyon, int yemek): Bu üretim taktiğinde koloninin popülasyon ve yemek değerleri birbirinden çıkarılır. Bu değer 10'u geçmemesi için 10'la modu alınır.

b. public int Uret(int populasyon, int yemek): Bu üretim taktiğinde koloninin popülasyon ve yemek değerleri birbiriyle toplanır. Bu değer 10'u geçmemesi için 10'la modu alınır. int değerinin taşmasından korunmak için negatif değer -1 ile çarpılır.

3. Taktik Sınıfı: İçinde 1 adet fonksiyon bulundurur. Bunlar şunlardır:

a. int Savas(int populasyon, int yemek): Savaş tekniği için kullanılır.

4. ATaktik ve BTaktik Sınıfları: Her iki sınıfta da 1'er fonksiyon bulunmaktadır. Her iki sınıf da Taktik sınıfından kalıtım almaktadır. İçlerindeki fonksiyonlar şunlardır:-

a. public int Savas(int populasyon, int yemek): Bu savaş taktiğinde koloninin popülasyon ve yemek değerleri çarpılır. Bu değer 1000'i geçmemesi için 1000'le modu alınır. int değerinin taşmasından korunmak için negatif değer -1 ile çarpılır.

b. public int Savas(int populasyon, int yemek): Bu savaş taktiğinde koloninin popülasyon ve yemek değerlerinin kareleri alınıp toplanır. Bu değer 1000'i geçmemesi için 1000'le modu alınır. int değerinin taşmasından korunmak için negatif değer -1 ile çarpılır.

5. Koloni Yapısı: Koloni ile ilgili bilgiler tutmakla beraber içinde 1 tane kurucu metot dahil toplam 13 farklı metot barındırmaktadır. Bahsedilen metotlar şunlardır:

a) Koloni (int populasyon, char simge): Kurucu metot.

b) public void Kazan(int kazanilanYemek): Koloni bir savaşı kazandığı için parametre olarak aldığı değeri yemek miktarına ekler.

c) public int Kaybet(int savasPuaniFarki): Koloni bir savaşı kazandığı için parametre olarak aldığı değerin yüzdelik değeri kadar yemek ve popülasyon kaybeder. Popülasyonun kaybettiği yemek miktarını diğer kolonin de bilebilmesi için bu değeri döndürür.

d) public void Yenilenme(int uretilenYemekMiktari): Koloninin savaştan sonra yapması gereken işleri yapar. Koloni üretim yapar; popülasyonu artır; yemek tüketir.

e) public bool YasiyorMu(): Koloninin yaşayıp yaşamadığını döndürür.

f) public String Goster(): Koloni hakkındaki bilgileri içeren string döndürür.

g) public int SavasTaktigi(): Koloninin savaş puanını hesaplar.

h) public int UretimTeknigi(): Koloninin ürettiği yemek miktarını hesaplar.

i) private void YuzdeselPopulasyonArttir(int yuzde): Kendisine verilen yüzde (örn %10) değeri kadar koloninin popülasyonunu arttırır.

j) private void YuzdeselPopulasyonAzalt (int yuzde): Kendisine verilen yüzde (örn %10) değeri kadar koloninin popülasyonunu azaltır.

k) private int YuzdeselYemekAzalt(int yuzde): Kendisine verilen yüzde (örn %10) değeri kadar koloninin yemek miktarını arttırır. Özellikle koloni kaybedince kullanıldığı için azaltılan yemek miktarını döndürür.

l) private void SayisalYemekArttir(int sayi): Kendisine verilen değer kadar koloninin yemek miktarını arttırır.

m) private void SayisalYemekAzalt(int sayi): Kendisine verilen değer kadar koloninin yemek miktarını azaltır.

6. Oyun Yapısı: Oyun ile ilgili bilgiler tutmakla beraber içinde 1 tane kurucu metot dahil toplam 7 farklı metot barındırmaktadır. Bahsedilen metotlar şunlardır:

a) Oyun (String veri): Oyun yapısı için nesne oluşturur.

b) public void Baslat(): Oyunu başlatır. Her turda koloniler ile ilgili bilgileri ekrana çıkartır. Onları savaşa sokar ve koloniler tur sonunda onlara yapmaları gerekeni yaptırır.

c) private void KolonilerOlustur(String veri): Popülasyon bilgilerini alıp koloniler oluşturur.

d) private void Muharebe(): Kolonileri savaştırır.

e) private void TurBitir(): Tur bittikten sonra yapılması gerekenleri yapar. Koloni üretim yapar; popülasyonu artır; yemek tüketir.

f) private void Ciz(): Kolonilere ait bilgileri ekrana yazar.

g) private boolean OyunBittiMi(): Oyunun bitip bitmediğini kontrol eder.

## 2. ÇIKTILAR

Bu programa ister aynı ister farklı girdiler verilsin hepsinde çıktı farklı olacaktır. Bunun sebebi projedeki birçok şeyin rastgele değişkenler aracılığı ile yapılmasıdır.

## 3. SONUÇ

Bu yazılım ile farklı popülasyonlara ait kolonilerin nasıl mücadele ettiğini simüle ederek görebilirsiniz.