



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

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

**C Programlama Dili İle Nesne Yönelimli Programlama Benzetimi**

**B201210008 - Beytullah YAYLA**

**SAKARYA**

**Mayıs, 2022**

Programlama Dillerinin Prensipleri Dersi

# Nesne Yönelimli Benzetim İle Bahis Oyunu

Beytullah Yayla

B201210008 1.Öğretim C Grubu

## Özet

Ödevimdeki temel amacım nesneye yönelik programlama paradigmasını C gibi nesne yönelimli olmayan bir dilde gerçeklemeye çalışmaktır. Nesneye yönelik programlamanın temel elemanlarından biri olan sınıfların C dilindeki karşılığı bu ödevde struct yapısıdır. Struct yapısının içinde değişkenler, fonksiyonlar vb. bulunur. Ödevde bir şans oyununun her tur sonunda kişilerin tutturduğu sayılara göre kişilerin bakiyelerinin güncellenmesi, belli bir miktarda parası olmayan oyuncuların elenmesi, masadaki bakiyenin güncellenmesi gibi kısımlar vardır. Bu kısımlarda kullanılacak 4 adet struct yapısı oluşturulmuştur. Bunlar Dosya, Kisi, Oyun structlarıdır. Nesneye dayalı programlamanın yapıcı fonksiyon, yıkıcı fonksiyon, kapsülleme ve public, private erişim belirteçleri C dilinde simüle edilmiştir. Javayla kıyaslayınca ödevin C’de gerçekleşmesi ve nesnelerin gerçek hayatla ilişkilendirilmesi daha zor olmuştur. Fakat javayla bir nesne oluşturduğumuzda arka tarafta neler olduğu konusunda bilgi sahibi olmak açısından ödev faydalıdır.

© 2022 Sakarya Üniversitesi.

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

Anahtar Kelimeler:Struct,Pointer,Bahis,Benzetim

## GELİŞTİRİLEN YAZILIM

Geliştirilen yazılım 4 adet ana structtan oluşmaktadır ve bu structların hemen hemen hepsi haberleşmektedir. Yazılımda kişiler ve bahis sayıları birer txt dosyasından okunmaktadır ve ona göre işlem yapılmaktadır. Sınıflar struct şeklinde yazılmıştır. Bu şekilde programda modülerlik sağlanmıştır. Her struct parametre olarak kendisini alır. Böylelikle nesneye dayalı programlamadaki this C dilinde benzetim yapılarak kullanılmış olur.

### 1.Structlar

#### 1.1.Header Dosyaları

**1.1.1.Kisi.h:**Bu .h dosyasında kişi yapısının bulunduracağı özellikler, fonksiyonların işaretçileri ve prototipleri bulundurulur. .c uzantılı kaynak dosyasında bu bilgilere bu header dosyasını include ederek ulaşabiliriz. Genel olarak kişinin adı, parası gibi özellikleri tutar.

**1.1.2.Dosya.h:**Dosya okuma işlemleri için gerekli olan fonksiyon işaretçilerini, fonksiyon gövdelerini ve dosyalardan döndürülen sayıları,dizileri tutar. Dosya.c kaynak dosyası için gerekli bilgiler ilgili dosyada include edilerek çağırılabilir.

**1.1.3.Oyun.h:**Oyunu uygun şekilde başlatmayı ve devam ettirmeyi sağlayan fonksiyonları ve işaretçilerini barındırır. Ayrıca kişiler yapısının dizisini ve sayılar yapısının dizisini tutar.

## 1.2.Kaynak Dosyaları

### 1.2.1.Kisi.c:

- Constructor ve fonksiyon gövdelerini tanımladığımız kısım. Constructor benzetimi yaptığımız kısımda özelliklerin değerleri atanır. İlgili yapı için heap bellek bölgesinden yer ayrılır. Fonksiyon işaretçileri kendisiyle ilgili fonksiyonların adreslerine atanır. Böylece bu işaretçileri çağırdığımız zaman aslında o fonksiyonu çağırıyor oluruz.
- KisiYazdir fonksiyonu Kisi this parametresi alır. Bu sayede this ile özelliklere erişebiliriz. KişiYazdir fonksiyonunun görevi özellikleri ekrana bastırmasıdır.

### 1.2.2.Dosya.c

- Constructor ve fonksiyonların gövdelerini tanımladığımız kısım. Constructor Dosyaa() parametre almaz. Dosya yapısı için heap bellek bölgesinden yer ayrılır ve fonksiyon göstericileri ve özelliklere ilgili değerler atanır.
- ReturnNumberOfKisi fonksiyonu Dosya this i parametre olarak alır. Bu fonksiyon okuduğumuz dosyadaki kişi sayısını bize döndürür.
- ReturnNumberOfSayi fonksiyonu Dosya this i parametre olarak alır.Okuduğumuz dosyadaki sayı sayısını bize döndürür.
- ReturnNumbers fonksiyonu Dosya yapısını parametre olarak alır. Görevi dosyadan okunan sayıları bir diziye atıp fonksiyon sonunda bu diziyi geri döndürmektir. Bu sayede bu dizi ileride Oyun yapısında kullanılacaktır. Bellek bölgesinden bir integer array için sayıların sayısı kadar yer ayrılır. Satırdaki sayılar integere döndürülüp arrayin ilgili indisine atanır.
- ReturnKisi fonksiyonu da Dosya yapısını parametre olarak alır. Görevi dosyadan okunan kişileri bir struct dizisine atıp oyun yapısında kullanılmak üzere geri döndürmektir.Dosya satır satır okunmuştur ve her satır için gerekli ayırma işlemleri yapıp yeni bir Kisi oluşturulmuştur. İlgili struct dizisine Kisi atanır. Son olarak geri döndürülür.
- DeleteDosya fonksiyonu heapte oluşturduğumuz Dosya yapısını serbest bırakmamızı sağlar.

### 1.2.3.Oyun.c

- Constructor ve fonksiyonların gövdelerini tanımladığımız kısım. Constructor Dosyaa() parametre almaz. Dosya yapısı için heap bellek bölgesinden yer ayrılır ve fonksiyon göstericileri ve özelliklere ilgili değerler atanır.
- Basla fonksiyonu parametre olarak gelen kişiler ve sayılar dizilerini alır. Tur ve kişiler için 2 for döngüsü kullanıldı. İlk for döngüsünde tur kontrolleri yapıldı. Her tur için aktif olan yarışmacılar ve

kontrolleri, bakiye değişimleri gibi işlemler içteki for döngüsünde yapıldı. Her tur sonunda en zengin kişinin ismi, bakiyesi gibi özellikler konsola yazdırıldı.

- DeleteOyun fonksiyonu Oyun yapısı için bellekten tahsis edilen alanı serbest bırakır.

## ÇIKTILAR

Aşağıdaki ekran çıktısı 3 kişilik bir oyun için alınmıştır. Uygun sağlamalar yapıldığında doğru olduğu teyit edilmektedir. Ayrıca hocamızın verdiği deneme dosyalarıyla da uygun sonuçlar elde edilmektedir.

```
#####
##          tur: 158          ##
##      masa parasi: 57206.62  ##
##                                     ##
## ----- ##
##          OYUN BİTTİ          ##
##                                     ##
##                                     ##
##                                     ##
#####
```

## SONUÇ

Bu çalışma nesneye dayalı programlamanın nesneye dayalı bir dil olmayan C dili ile nasıl gerçekleştirilebileceği konusunda bana güzel bir tecrübe kazandırmıştır. Ayrıca günümüz projelerinde çokça kullanılan nesneye dayalı programlama paradigmasını kullanırken arka tarafta nasıl işlemler gerçekleştirilebileceği konusunda epey bir bilgi sağladığımı düşünüyorum. Bu proje nesneye dayalı programlama yeteneklerini bir adım daha ileri taşımak isteyenler için güzel bir proje olacaktır.