

İTÜ



**Department of
Computer
Engineering**

BLG 252E
Object Oriented Programming
3rd Homework Templates and
Exception Handling
Report

Muhammed Raşit EROL

150150023

1 CLASSES

1.1 MONEY CLASS

1.1.1 Attributes

Private:

Money classı için attribute olarak **lira** ve **kurus** private olarak tanımlanmıştır. Private olan bu datalara erişmek için getter ve setter methodları eklenmiştir. Getter methodları datalar üzerinde değişiklik yapmadığı için const olarak tanımlanmıştır.

1.1.2 Methods

Method olarak öncelikle **const getter** ve const olmayan **setter** methodları yazılmıştır. Ayrıca genericArray' de miktar 100 ile karşılaştırılacağı ve dizinin toplamı bulunacağı için **toplama(+)**, **çıkarma(-)**, **assignment(=)**, **küçüktür(<)**, **büyüktür(>)**, ve **eşit eşit(==)** operatörleri için overloading yapılmıştır. Toplama ve çıkarma kısmında paraların 100 kurus geçen veya 0'ın altına inen değerleri için çözüm üretilmiş olup kaynak koddan incelenebilir. Ayrıca Money objesini cout ile yazabilmek için **<< operatörü** overload edilmiştir.

1.1.2.1 Constructors ve Destructors

Money classı için 4 tane constructor ve destructor yazılmıştır. Bu constructorlardan ilki **default constructor** olup obje için lira ve kurus değerlerine 0'a eşitlemektedir. Ayrıca Money objelerinin lira ve kurus değerlerini birbirine kopyalayabilmesi için **copy constructor** yazılmıştır. (lira,kurus) şeklinde obje tanımlanabilmesi için de bir tane **constructor** yazılmıştır ve son olarak Money objeleri arasında karşılaştırma yapabilmek için (double) şeklinde double türde para alan **constructor** yazılmıştır. Bu constructor virgülden sonrasını kurusa eşitlemektedir ve bu constructor sadece virgülden sonra 2 basamak alçak şekilde yazılmış olup .2 gibi tek basamaklı double yollamak istenmeyen sonuçlara neden olabilir. **Destructor**, dynamic olarak memory alınmadığı için boş bırakılmıştır.

1.2 GENERICARRAY TEMPLATE CLASS

GenericArray classı template olduğu için ödevde istenen 3 type için de çalışmaktadır.

1.2.1 Attributes

Private:

Dizinin boyutunu tutmak için **const size** değişkeni kullanılmıştır. Ayrıca sum fonksiyonu çalıştıktan sonra bonus var ise bunu daha sonra yazdırırken tespit etmek için **char bonus** flagi kullanılmıştır. Bu datalar private olarak tanımlanmıştır. Çünkü 2 değişken de fonksiyonlar dışından erişilmesi güvenli olmayan değişkenlerdir.

Public:

Public olarak ise dizinin elemanlarını tutmak için **Type *elements** tanımlanmıştır. Bu değişken template olduğu için her type için kullanılabilir. Aynı şekilde dizideki elemanların toplamını sakladığımız **total** değişkeni de Type türünde olmak zorundadır. Bu değişkenler public olarak tanımlandığı için mainden çağırılabilir.

1.2.2 Methods

Constructor ve destructor dışında dizini elemanlarını bulmak için **sum** fonksiyonu kullanılmıştır. Ayrıca type'a göre çıktı alabilmek için **const print** fonksiyonu yazılmıştır. Sum fonksiyonu template olup başka typelar için de çalışmaktadır.

1.2.2.1 Constructors ve Destructors

Bu class için 3 constructor ve destructor yazılmıştır.

Default constructor: Bu constructorda size 1'e totale 0'a ve bonus flagine 'x' atanmaktadır. Buradaki 'x' henüz sum fonksiyonun çalışmadığını göstermektedir. Çünkü sum fonksiyonu çalıştırılınca bonus varsa flag '+' yoksa '-' yapılmaktadır. Eğer sum fonksiyonuna girip de hala 'x' değerine sahip olan bir obje varsa bu objenin total değeri 100'e eşittir.

Copy constructor: Bu constructorda generic arrayin içindeki datalar başka bir objeye birer birer kopyalanmaktadır. Burada dikkat edilmesi gereken nokta size değerleri const olduğu için eşit boyutta dizilerin bu constructorı çağırması gerektiridir.

Parametre alan constructor: Bu constructor dizinin boyutunu alarak dynamic olarak alan almaktadır.

Destructor: Destructor için ise memoryden dynamic olarak alan alındığı için bu alan geri verilmektedir.

1.2.2.2 Void Print

Bu fonksiyon aldığı string objesine göre ekrana yazdırılacak tipi belirlemekte ve ona göre çıktı vermektedir. Ayrıca bu fonksiyonun içinde bonus flagi kontrol edilerek uygun çıktı verilmektedir.

1.2.2.3 Type sum

Bu fonksiyonda dizinin içindeki bütün elemanlarda gezilerek bulunan sonuç total değişkenine atılmaktadır. Bu değişken template olup her tür için çalışmaktadır. Money objesinin bu aşamada düzgün çalışması için assignment operatörü ve toplama operatörü overload edilmiştir.

Diğer kısımda 100 ile karşılaştırma yapılmaktadır. Burada da Money objesinin 100 ile karşılaştırılabilmesi için Type(100) gibi Money tipinde objeler kullanılmıştır. Bu objenin constructorundan yukarıda bahsedilmiştir. Overload edilen büyüktür, küçüktür operatörleriyle yapılan karşılaştırma sonucunda bonus flagine uygun değerler atanmaktadır. Son olarak bu fonksiyon dizinin tipinde toplam objesini return etmektedir.

[illegible]