

2020-2021 Güz Dönemi
Ege Üniversitesi Bilgisayar Mühendisliği Bölümü
112 Algoritma ve Programlama II Dersi
ARASINAV PROJESİ TANITIM KAĞIDI

Veriliş Tarihi: 03.05.2021

Teslim Tarihi: 22.05.2021 Saat 23:55

Genel Problem Tanımı:

Dosyadan okunan veriler kullanılarak geometrik nesnelere ilişkin bir sınıf hiyerarşisi oluşturulması, soyut sınıf yazımı, çok-biçimlilik (polymorphism) gerçekleştirimi ve "polymorphic" işleyişin örneklenmesi, bir arayüzün (Comparable interface) gerçekleştirilmesi, dizi tanımı ve kullanımı gibi konuların ele alınması hedeflenmektedir.

Gereksinimler:

GeometrikNesne Sınıfı

- * Geometrik nesneleri genel olarak temsil eden ve soyut (abstract) bir sınıf olan "GeometrikNesne" sınıfı yazılacaktır. Bu sınıf aynı zamanda Java'da bulunan "Comparable" arayüzünü de gerçekleştirecektir.
- * Bu sınıf nesneye verilen etiketi ("Daire 1" gibi bir isim) tutmak için String tipinde "etiket" değişkeni ile şeklin oluşturulma tarihini tutmak için Date sınıfı tipinde (Ders kitabındaki "Date" sınıfını aynen projenize katarak kullanabilirsiniz) tarih değişkenlerine sahiptir.
- * Parametresiz yapıcı (constructor) metot, Date sınıfındaki parametresiz yapıcı metodu kullanarak tarih oluşturacaktır.
- * Etiket ve tarih için parametreler alan iki parametrelili yapıcı metot oluşturulacaktır.
- * "Copy constructor" olarak adlandırılan yapıcı metot oluşturulacaktır. *NOT: Ders kitabında "Privacy Leak" olarak adlandırılan durumların olmamasını sağlayarak yazınız.*
- * Get ve Set metotları (yine eğer gerekiyorsa ilgili metotta "privacy leak" engellenecek şekilde yazılmalıdır).
- * toString metodu yazılmalıdır.
- * Bu sınıfta soyut (abstract) metotlar olan "double alanHesapla()", "double cevreHesapla()" ve "int compareTo(Object o)" metotlarının imzaları vardır.

Daire Sınıfı

- * GeometrikNesne sınıfından kalıtım ile türetilen **Daire** sınıfı yazılacaktır.
- * Bu sınıf double tipinde **yaricap** değişkenine sahiptir.
- * Parametresiz yapıcı (constructor) metot, yaricap değerine 1.0 atamaktadır. .
- * Gerekli verileri parametre olarak alan yapıcı metot oluşturulacaktır.

- * "Copy constructor" olarak adlandırılan yapıcı metot oluşturulacaktır.
- * Yarıcap değişkeni için Get ve Set metotları yazılmalıdır.
- * toString metodu yazılmalıdır.
- * Üst sınıftaki soyut metotların gerçekleştirimleri yapılmalıdır. Daire alanı için $\pi * r * r$, çevresi için $2 * \pi * r$ formülleri kullanılabilir. CompareTo metodu şu şekilde gerçekleştirilecektir: eğer metodun üzerinde çağrıldığı daire nesnesinin yarıcapı parametrede verilen daire nesnesinin yarıcapından büyük ise 1, eğer metodun üzerinde çağrıldığı daire nesnesinin yarıcapı parametrede verilen daire nesnesinin yarıcapından küçük ise -1 ve eşitlik durumunda 0 döndürülecektir.

Dikdortgen Sınıfı

- * GeometrikNesne sınıfından kalıtım ile türetilen **Dikdortgen** sınıfı yazılacaktır.
- * Bu sınıf double tipinde **en ve boy** değişkenlerine sahiptir.
- * Parametresiz yapıcı (constructor) metot, en ve boy değerlerine 1.0 atamaktadır. .
- * Gerekli verileri parametre olarak alan yapıcı metot oluşturulacaktır.
- * "Copy constructor" olarak adlandırılan yapıcı metot oluşturulacaktır.
- * En ve boy değişkenleri için Get ve Set metotları yazılmalıdır.
- * toString metodu yazılmalıdır.
- * Üst sınıftaki soyut metotların gerçekleştirimleri yapılmalıdır. Dikdortgen alanı için $en * boy$, çevresi için $2 * (en + boy)$ formülleri kullanılabilir. CompareTo metodu şu şekilde gerçekleştirilecektir: eğer metodun üzerinde çağrıldığı dikdörtgen nesnesinin alanı parametrede verilen dikdörtgen nesnesinin alanından büyük ise 1, eğer metodun üzerinde çağrıldığı dikdörtgen nesnesinin alanı parametrede verilen dikdörtgen nesnesinin alanından küçük ise -1 ve eşitlik durumunda 0 döndürülecektir.

Silindir Sınıfı

- * GeometrikNesne sınıfından kalıtım ile türetilen **Silindir** sınıfı yazılacaktır.
- * Bu sınıf double tipinde **yaricap ve uzunluk** değişkenlerine sahiptir.
- * Parametresiz yapıcı (constructor) metot, yarıcap ve uzunluk değerlerine 1.0 atamaktadır. .
- * Gerekli verileri parametre olarak alan yapıcı metot oluşturulacaktır.
- * "Copy constructor" olarak adlandırılan yapıcı metot oluşturulacaktır.
- * Yarıcap ve uzunluk değişkenleri için Get ve Set metotları yazılmalıdır.
- * toString metodu yazılmalıdır.
- * Bu sınıfa özel olarak "**hacimHesapla**" metodu yazılacaktır. Hacim hesaplama için $\pi * yarıcap * yarıcap * uzunluk$ formülü kullanılabilir.
- * Üst sınıftaki soyut metotların gerçekleştirimleri yapılmalıdır. Silindir alanı için $2 * yarıcap * \pi$ (yaricap + uzunluk), silindir çevresi için $(2 * 2 * yarıcap * \pi) + (2 * uzunluk)$ formülleri kullanılabilir.

CompareTo metodu şu şekilde gerçekleştirilecektir: eğer metodun üzerinde çağrıldığı silindir nesnesinin hacmi parametrede verilen silindir nesnesinin hacminden büyük ise 1, eğer metodun üzerinde çağrıldığı silindir nesnesinin hacmi parametrede verilen silindir nesnesinin hacminden küçük ise -1 ve eşitlik durumunda 0 döndürülecektir.

Deneme Sınıfı

Deneme sınıfı içinde main metot yazarak aşağıda belirtilenleri yapınız.

Aşağıda verilen input.txt dosyasını okuyarak oluşturacağınız nesneleri GeometrikNesne tipinde tanımlı olan bir dizi içine sırası ile yerleştiriniz. (*Proje kontrolü sırasında farklı veriler içeren benzer başka bir dosya kullanacağımızı gözden kaçırmayınız.*)

input.txt

```
daire dai1 8.0 date 12 30 2002
dikdortgen dik1 10.0 5.0 11 18 2003
daire dai2 5.0 date 7 18 2001
daire dai3 10.0 date 5 18 1990
silindir sil1 20.0 15.0 date 9 10 2018
silindir sil2 5.0 8.0 date 4 23 1980
dikdortgen dik2 4.0 8.0 date 1 14 2000
```

Dai3 etiketli nesneyi kullanarak "copy constructor" ile dördüncü bir daire nesnesi oluşturunuz. Dik2 etiketli nesneyi kullanarak "copy constructor" ile üçüncü bir dikdörtgen nesnesi oluşturunuz. Sil2 etiketli nesneyi kullanarak "copy constructor" ile üçüncü bir silindir nesnesi oluşturunuz. Bu nesneler de diziye eklenecektir.

ÖDEV KAĞIDINDA GÜNCELLEME-1:

Proje kontrolü sırasında farklı bir dosya verilebileceğini düşünerek, "copy constructor" kullanımlarını şu şekilde yapınız: Son oluşturulan daire nesnesini kullanarak yeni bir daire nesnesi oluşturunuz. Son oluşturulan dikdörtgen nesnesini kullanarak yeni bir dikdörtgen nesnesi oluşturunuz. Son oluşturulan silindir nesnesini kullanarak yeni bir silindir nesnesi oluşturunuz. Yeni oluşturulan bu nesneleri de diziye ekleyiniz.

GeometrikNesne tipinde parametre alan static bir "polymorphicYazdir" metodu yazınız. Bu metotta, parametre olarak gelen nesnenin bilgilerini, çevre ve alan değerlerini yazdırınız. Eğer parametre olarak gelen nesne Silindir nesnesi ise hacim değeri de yazdırılmalıdır. Bu metot, main içinden çağrılacaktır.

GeometrikNesne tipinde iki adet parametre alan static bir "karsilastir" metodu yazınız. Bu metot, daha önce her sınıf için yazdığınız compareTo metotlarını kullanacaktır. "karsilastir" metodu da main içinden çağrılacaktır. Bu metodun kullanımını iki ve üçüncü daire, üç ve dördüncü daire, bir ve

ikinci dikdörtgen, iki ve üçüncü dikdörtgen, bir ve ikinci silindir ve iki ve üçüncü silindir nesnelerinin karşılaştırılması için main içinde örnekleyiniz.

ÖDEV KAĞIDINDA GÜNCELLEME-2:

Güncelleme-1 de belirtilenlere göre karşılaştırma metodu denemelerini şu şekilde yapınız:

Karşılaştır metodunun kullanımını ilk ve son daire nesneleri, son daire ve ondan bir önceki daire nesneleri; ilk ve son dikdörtgen nesneleri, son dikdörtgen ve ondan bir önceki dikdörtgen nesneleri; ilk ve son silindir nesneleri, son silindir ve ondan bir önceki silindir nesnelerinin karşılaştırılması için "main" içinde örnekleyiniz. (Söz edilen nesnelere, "copy constructor" ile oluşturulanlar dahildir, diğer deyiş ile son daire nesnesi "copy constructor" ile oluşturulan daire nesnesidir.)

Son olarak bir adet daha tek boyutlu bir dizi tanımlayarak, bu dizinin içine sırası ile çevre ortalaması (tüm geometrik nesneler göz önüne alınarak), alan ortalaması (tüm geometrik nesneler göz önüne alınarak), hacim ortalaması (silindirler göz önüne alınarak), en küçük çevre değeri, en büyük çevre değeri, en küçük alan değeri, en büyük alan değeri, en küçük hacim değeri ve en büyük hacim değeri bulunarak aktarılmalıdır. (En küçük ve en büyük değerler bulunurken, hacim için silindirler, diğerleri için tüm geometrik nesneler dikkate alınmalıdır.)

ÖDEV KAĞIDINDA GÜNCELLEME-3: Bu tek boyutlu dizi içeriği ekrana yazdırılmalıdır. Yazdırma formatını (ondalık basamak sayısını) istediğiniz gibi ayarlayabilirsiniz.

Tarihler, Rapor İçeriği ve Diğer Bilgiler:

1. Projeler tek kişi olarak (bireysel) yapılacak olup, **22 Mayıs 2021** günü **saat 23:55'e** kadar Ege Ders ortamında açılacak ilgili kaynak kod yükleme klasörüne yüklenecektir. NetBeans ortamında kodlanarak çalıştırılacak Java projesine ilişkin klasör sıkıştırılmış (rar gibi) halde, dosya adı öğrenci numarası (örneğin 05090004235.rar gibi) olacak şekilde sisteme yüklenmelidir.
2. Kaynak kodlarda, Deneme sınıfı da dahil olmak üzere gerekli yerlerine yeteri kadar açıklama satırları eklenmiş olmalıdır. Yetersiz açıklama satırı olması durumunda puan kırılacaktır.
3. Sistemde sorun olabileceğini düşünerek yüklemeyi son dakikaya bırakmayınız. Arasınava projesi olduğu için geç kalan projeler kabul edilemeyecektir.
4. Başka kişiden alıntı olduğu anlaşılan projeler, gerek alıntı yapılan gerekse alıntı yapan kişi için değerlendirme dışı bırakılacaktır. Üniversite senatosunun, proje şeklindeki sınavlarda benzerlik tespiti yapılması ve kopya durumlarında resmi disiplin prosedürlerinin uygulanması konusunda bir yazısı vardır.