



T.C

**KOCAELİ SAęLIK VE TEKNOLOJİ ÜNİVERSİTESİ  
MÜHENDİSLİK VE DOęA BİLİMLERİ FAKÜLTESİ  
YAZILIM MÜHENDİSLİęİ PROGRAMI  
PROGRAMLAMA LAB-1 ÖDEV-3**

**ÖDEV KONUSU  
GEOMETRİK PROBLEMLER**

**Hazırlayan  
YAVUZHAN KURŞUN  
230502047  
İBRAHİM KEÇİCİ  
230502049**

**DERS SORUMLUSU  
PROF. DR. HÜSEYİN TARIK DURU**

06.12.2023

## İÇİNDEKİLER

1. ÖZET	3
2. GİRİŞ	3
3. YÖNTEM	4
3.1 Nokta Sınıfı	4
3.1.1 Özellikler	4
3.1.2 Nokta() Yapıcı Fonksiyonu	4
3.1.3 Nokta(double xy) Yapıcı Fonksiyonu	4
3.1.4 Nokta(double x, double y) Yapıcı Fonksiyonu	5
3.1.5 Nokta(Nokta &nokta) Yapıcı Fonksiyonu	5
3.1.6 Nokta(Nokta &nokta, double ofset_x, double ofset_y) Yapıcı Fonksiyonu	5
3.1.7 double getX() Fonksiyonu	5
3.1.8 double getY() Fonksiyonu	5
3.1.9 void setX(double x) Fonksiyonu	6w
3.1.10 void setY(double y) Fonksiyonu	6
3.1.11 void setXY(double x, double y) Fonksiyonu	6
3.1.12 string toString() Fonksiyonu	6
3.1.13 void yazdir() Fonksiyonu	6
3.2 DogruParcasi Sınıfı	6
3.2.1 Özellikler	7
3.2.2 DogruParcasi(Nokta a, Nokta b) Yapıcı Fonksiyonu	7
3.2.3 DogruParcasi(DogruParcasi &dogruParcasi) Yapıcı Fonksiyonu	7
3.2.4 DogruParcasi(Nokta ortaNokta, double uzunluk, double egim) Yapıcı Fonksiyonu	7
3.2.5 Nokta getA() Fonksiyonu	8
3.2.6 Nokta getB() Fonksiyonu	8
3.2.7 void setA(Nokta nokta) Fonksiyonu	8
3.2.8 double uzunluk() Fonksiyonu	8
3.2.9 Nokta kesisimNoktasi(Nokta nokta) Fonksiyonu	8
3.2.10 Nokta ortaNokta() Fonksiyonu	8
3.2.11 string toString() Fonksiyonu	9
3.2.12 void yazdir() Fonksiyonu	9
3.3 Daire Sınıfı	9
3.3.1 Özellikler	9

Ödev No: 1	Tarih 11.12.2022	2/16
------------	------------------	------

---

3.3.2 Daire(Nokta merkez, double yarıcap) Yapıcı Fonksiyonu	9
3.3.3 Daire(Daire &daire) Yapıcı Fonksiyonu	10
3.3.4 Daire(Daire &daire, int x) Yapıcı Fonksiyonu	10
3.3.5 Nokta getMerkez() Fonksiyonu	10
3.3.6 double getYarıcap() Fonksiyonu	10
3.3.7 void setMerkez(Nokta nokta) Fonksiyonu	10
3.3.8 void setYarıcap(double yarıcap) Fonksiyonu	10
3.3.9 double alan() Fonksiyonu	11
3.3.10 double çevre() Fonksiyonu	11
3.3.11 int kesisim(Daire daire) Fonksiyonu	11
3.3.12 string toString() Fonksiyonu	11
3.3.13 void yazdır() Fonksiyonu	11
3.4 Üçgen Sınıfı	11
3.4.1 Özellikler	12
3.4.2 Üçgen(Nokta a, Nokta b, Nokta c) Yapıcı Fonksiyonu	12
3.4.3 Nokta getA() Fonksiyonu	12
3.4.4 Nokta getB() Fonksiyonu	12
3.4.5 Nokta getC() Fonksiyonu	13
3.4.6 void setA(Nokta nokta) Fonksiyonu	13
3.4.7 void setB(Nokta nokta) Fonksiyonu	13
3.4.8 void setC(Nokta nokta) Fonksiyonu	13
3.4.9 string toString() Fonksiyonu	13
3.4.10 double alan() Fonksiyonu	13
3.4.11 double çevre() Fonksiyonu	14
3.4.12 double* acılar() Fonksiyonu	14
4. SONUÇ VE ÖĞRENİLEN DERSLER	14
5. KAYNAKÇA	14

Ödev No: 1	Tarih 11.12.2022	3/16
------------	------------------	------

---

## 1. ÖZET

Bu proje, 4 adet geometrik problemi çözen kod dosyalarından oluşmaktadır.

İlk sorumuzda, Bir noktanın x ve y olmak üzere koordinatlarını nesne olarak tutar.

İkinci sorumuzda, Bir doğru parçasının iki noktasının koordinat değerlerini tutar.

Üçüncü sorumuzda, Dairenin merkezini ve yarıçapını tutan bir Daire sınıfı oluşturduk.

4. ve son sorumuzda ise, Üç adet koordinat içeren nokta değişkenlerinden Üçgen sınıfını oluşturduk.

Bu proje, geometrik problemleri çözmek için temel bir altyapı sağlamaktadır.

Bu altyapıyı kullanarak, daha karmaşık geometrik problemleri çözmek için yeni sınıflar ve yöntemler geliştirilebilir.

## 2. GİRİŞ

Bu rapor, C++ programlama dili kullanılarak oluşturulmuş dört adet geometrik sınıfın tasarımını ve implementasyonunu incelemektedir. Bu sınıflar, bir noktanın x ve y koordinatlarını tutan Nokta sınıfı, bir doğru parçasının iki noktasını tutan DogruParcasi sınıfı, bir dairenin merkezini ve yarıçapını tutan Daire sınıfı ve üç noktayı tutan Ucgen sınıfıdır.

Bu sınıflar, geometrik şekillerin temel özelliklerini temsil etmek ve hesaplamak için tasarlanmıştır. Örneğin, Nokta sınıfı, bir noktanın x ve y koordinatlarını temsil eder. DogruParcasi sınıfı, bir doğru parçasının iki noktasının koordinatlarını temsil eder. Daire sınıfı, bir dairenin merkezini ve yarıçapını temsil eder. Ucgen sınıfı, üçgenin üç noktasının koordinatlarını temsil eder.

Bu sınıflar, nesneye yönelik programlama (OOP) prensiplerini kullanarak tasarlanmıştır. Her

Ödev No: 1	Tarih 11.12.2022	4/16
------------	------------------	------

---

sınıf, belirli bir geometrik şekli temsil etmek için gerekli olan verileri ve yöntemleri içerir. Sınıflar, yapıcılar, getter ve setter yöntemleri, özel üye değişkenleri ve diğer OOP özelliklerini kullanarak tasarlanmıştır.

### 3. YÖNTEM

Bu bölümde, projede benimsenen ana metot, takip edilen yöntemler ve araçlara yer verilmelidir.

#### 3.1 Nokta Sınıfı

Bu sınıf, bir noktanın x ve y koordinatlarını tutar ve bunları değiştirmek veya yazdırmak için çeşitli yöntemler sağlar. Ayrıca, farklı parametrelerle Nokta nesneleri oluşturmak için birden fazla yapıcı fonksiyon içerir.

##### 3.1.1 Özellikler

###### 3.1.1.1 x üye değişkeni

Bu gizli üye değişken, noktanın x üzerindeki konumunu değiştirir. Veri kapsülleme yaklaşımına uygun bir şekilde tanımlanmıştır.

###### 3.1.1.2 y üye değişkeni

Bu gizli üye değişken, noktanın y üzerindeki konumunu değiştirir. Veri kapsülleme yaklaşımına uygun bir şekilde tanımlanmıştır.

#### 3.1.2 Nokta() Yapıcı Fonksiyonu

Bu yapıcı fonksiyon, Nokta sınıfından bir nesne oluşturulduğunda çağrılır. Yapıcı fonksiyon, nesnenin x ve y koordinatlarını varsayılan değerlerle (0, 0) başlatan setXY() fonksiyonunu çağırır. Yani, bir nesne oluşturulduğunda, bu nesnenin koordinatları otomatik olarak orjin (0, 0) olarak ayarlanır.

#### 3.1.3 Nokta(double xy) Yapıcı Fonksiyonu

Bu yapıcı fonksiyon, Nokta sınıfından bir nesne oluşturulduğunda çağrılır. Yapıcı fonksiyon, tek bir parametre olan xy değerini alır. Ardından, bu parametreyi kullanarak nesnenin setXY fonksiyonunu çağırarak hem x hem de y koordinatlarını belirler. Bu sayede, tek bir değerle hem x hem de y koordinatlarına aynı değeri atamış olursunuz. Yani, bir nesne oluşturulduğunda, kullanıcı istediği xy değerini belirleyebilir ve bu değer hem x hem de y

Ödev No: 1	Tarih 11.12.2022	5/16
------------	------------------	------

---

koordinatlarına atanır.

#### **3.1.4 Nokta(double x, double y) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, Nokta sınıfından bir nesne oluşturulduğunda çağrılır. Yapıcı fonksiyon, iki parametre alır: x ve y. Ardından, bu parametreleri kullanarak nesnenin setXY fonksiyonunu çağırarak x ve y koordinatlarını belirler. Yani, bir nesne oluşturulduğunda, kullanıcı istediği x ve y değerlerini belirleyebilir.

#### **3.1.5 Nokta(Nokta &nokta) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, Nokta sınıfından bir nesne oluşturulurken başka bir Nokta nesnesinden kopya almak için kullanılır. Parametre olarak nokta adında bir referans alır. Ardından, bu referans üzerinden getX ve getY fonksiyonlarını kullanarak nokta nesnesinin x ve y koordinatlarını alır ve bu değerleri nesnenin kendi setXY fonksiyonu ile ayarlar. Sonuç olarak, yeni oluşturulan nesne, nokta nesnesinin aynısı olur. Bu şekilde, bir nesnenin değerlerini başka bir nesneye kopyalama işlemi gerçekleştirilmiş olur.

#### **3.1.6 Nokta(Nokta &nokta, double offset\_x, double offset\_y) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, Nokta sınıfından bir nesne oluştururken, başka bir Nokta nesnesinin koordinatlarına belirli ofset değerlerini ekleyerek yeni koordinatları belirler. Parametre olarak nokta adında bir referans, offset\_x ve offset\_y adında iki double değer alır. Ardından, nokta nesnesinin getX ve getY fonksiyonlarını kullanarak mevcut koordinatları alır, ofset değerlerini ekler ve bu yeni değerleri nesnenin setXY fonksiyonu ile ayarlar. Sonuç olarak, yeni oluşturulan nesnenin koordinatları, verilen ofset değerleri kadar kaydırılmış olur.

#### **3.1.7 double getX() Fonksiyonu**

Bu fonksiyon, bir Nokta nesnesinin x koordinatını döndürür. return this->x; ifadesi, sınıfın üye değişkeni olan x'in değerini geri döndürür. Bu sayede, sınıfın dışından bu x koordinatına erişim sağlanabilir.

#### **3.1.8 double getY() Fonksiyonu**

Bu fonksiyon, bir Nokta nesnesinin y koordinatını döndürür. return this->y; ifadesi, sınıfın üye değişkeni olan y'in değerini geri döndürür. Bu sayede, sınıfın dışından bu y koordinatına erişim sağlanabilir.

Ödev No: 1	Tarih 11.12.2022	6/16
------------	------------------	------

---

### 3.1.9 void setX(double x) Fonksiyonu

Bu fonksiyon, bir Nokta nesnesinin x koordinatını belirlemek için kullanılır. Parametre olarak gelen x değeri, nesnenin x üye değişkenine atanır. Bu sayede, sınıfın dışından setX fonksiyonu çağrılarak Nokta nesnesinin x koordinatı istenilen değere ayarlanabilir.

### 3.1.10 void setY(double y) Fonksiyonu

Bu fonksiyon, bir Nokta nesnesinin y koordinatını belirlemek için kullanılır. Parametre olarak gelen y değeri, nesnenin y üye değişkenine atanır. Bu sayede, sınıfın dışından setY fonksiyonu çağrılarak Nokta nesnesinin y koordinatı istenilen değere ayarlanabilir.

### 3.1.11 void setXY(double x, double y) Fonksiyonu

Bu fonksiyon, bir Nokta nesnesinin hem x hem de y koordinatlarını belirlemek için kullanılır. setX ve setY fonksiyonlarını çağırarak, gelen x ve y değerlerini sırasıyla nesnenin x ve y koordinatlarına atar. Bu sayede, tek bir adımda hem x hem de y koordinatlarını değiştirmek mümkün olur.

### 3.1.12 string toString() Fonksiyonu

Bu fonksiyon, bir Nokta nesnesinin x ve y koordinatlarını içeren bir string ifadesini oluşturur. getX ve getY fonksiyonlarını çağırarak nesnenin koordinatlarını alır, std::to\_string fonksiyonu ile bunları stringe dönüştürür ve ardından bu değerleri birleştirerek bir string ifadesi oluşturur. Bu sayede, bir Nokta nesnesinin koordinatlarını içeren bir string ifadesine kolayca erişim sağlanabilir.

### 3.1.13 void yazdir() Fonksiyonu

Bu fonksiyon, bir Nokta nesnesinin koordinatlarını ekrana yazdırmak için kullanılır. toString fonksiyonunu çağırarak noktanın koordinatlarını içeren bir string ifadesini alır ve ardından bu ifadeyi std::cout ile ekrana yazdırır. std::endl ifadesi, bir satır sonu karakteri ekleyerek bir alt satıra geçmeyi sağlar. Bu sayede, Nokta nesnesinin koordinatları ekrana basılabilir.

## 3.2 DogruParcasi Sınıfı

Bu sınıf, matematiksel doğru parçalarını temsil eden bir C++ sınıfıdır. İki noktanın yanı sıra uzunluk ve eğim bilgilerini kullanarak doğru parçaları oluşturabilir. Sınıf, uzunluk hesaplama,

Ödev No: 1	Tarih 11.12.2022	7/16
------------	------------------	------

---

kesişim noktası bulma ve string temsil oluşturma gibi temel işlevleri içerir. Kullanıcılar, doğru parçalarını başlangıç ve bitiş noktalarıyla kolayca oluşturabilir ve bu doğru parçalarıyla ilgili temel hesaplamaları yapabilir.

### **3.2.1 Özellikler**

#### **3.2.1.1 a üye değişkeni**

Bu üye değişkeni başlangıç noktasını temsil eder. Yapıcı fonksiyonlar aracılığıyla bu değişken, doğru parçanın oluşturulması sırasında belirlenen noktalara atanır. Bu üye değişken, doğru parçanın başlangıç noktasını depolamak ve erişmek için kullanılır.

#### **3.2.1.2 b üye değişkeni**

Bu üye değişkeni bitiş noktasını temsil eder. Yapıcı fonksiyonlar aracılığıyla bu değişken, doğru parçanın oluşturulması sırasında belirlenen noktalara atanır. Bu üye değişken, doğru parçanın bitiş noktasını depolamak ve erişmek için kullanılır.

### **3.2.2 DogruParcasi(Nokta a, Nokta b) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, "DogruParcasi" sınıfının nesnelerini oluştururken kullanılan bir inşa edici fonksiyondur. Bu fonksiyon, iki nokta parametresi alır: "a" ve "b". Bu noktalar, doğru parçasının başlangıç ve bitiş noktalarını temsil eder. Yapıcı fonksiyon, "a" ve "b" noktalarını sınıfın özel üye değişkenlerine atanarak doğru parçasını oluşturur. Bu sayede, bir "DogruParcasi" nesnesi oluşturulduğunda başlangıç ve bitiş noktaları belirlenmiş olur.

### **3.2.3 DogruParcasi(DogruParcasi &dogruParcasi) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, "DogruParcasi" sınıfının kopya oluşturucu (copy constructor) olarak adlandırılır. Fonksiyon, mevcut bir "DogruParcasi" nesnesinin kopyasını alarak yeni bir nesne oluşturur. Bu işlem, nesne kopyalandığında başka bir nesnenin bellek alanını aynı verilerle doldurmayı amaçlar. Yani, mevcut bir doğru parçasının özellikleri, aynı değerlere sahip yeni bir doğru parçasına aktarılır.

### **3.2.4 DogruParcasi(Nokta ortaNokta, double uzunluk, double egim) Yapıcı Fonksiyonu**

Bu yapıcı fonksiyon, "DogruParcasi" sınıfının bir başka türdeki yapıcı fonksiyonudur. Fonksiyon, bir nokta, uzunluk ve eğim parametrelerini alarak doğru parçası oluşturur. İlgili hesaplamaları yaparak, verilen orta nokta etrafında belirtilen uzunluk ve eğime sahip bir

Ödev No: 1	Tarih 11.12.2022	8/16
------------	------------------	------



---

doğru parçasını tanımlar. Bu sayede, kullanıcılar doğru parçalarını daha soyut bir şekilde, orta nokta, uzunluk ve eğim değerleriyle oluşturabilirler.

### **3.2.5 Nokta getA() Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç noktasını (Nokta A) döndürerek, bu noktanın değerlerine erişimi sağlar. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının başlangıç noktasını öğrenebilir ve bu bilgiyi program içinde kullanabilirler.

### **3.2.6 Nokta getB() Fonksiyonu**

Bu fonksiyon, doğru parçasının bitiş noktasını (Nokta B) döndürerek, bu noktanın değerlerine erişimi sağlar. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının bitiş noktasını öğrenebilir ve bu bilgiyi program içinde kullanabilirler.

### **3.2.7 void setA(Nokta nokta) Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç noktasını belirlemek için kullanılır. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının başlangıç noktasını istedikleri bir nokta değeriyle güncelleyebilirler. Bu şekilde, program içinde doğru parçasının başlangıç noktasını değiştirmek mümkün olur.

### **3.2.8 double uzunluk() Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç ve bitiş noktaları arasındaki uzunluğu hesaplar ve bu uzunluğu double veri türünde bir değer olarak döndürür. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının uzunluğunu elde edebilir ve bu bilgiyi program içinde kullanabilirler.

### **3.2.9 Nokta kesisimNoktasi(Nokta nokta) Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç ve bitiş noktaları arasındaki uzunluğu hesaplar ve bu uzunluğu double veri türünde bir değer olarak döndürür. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının uzunluğunu elde edebilir ve bu bilgiyi program içinde kullanabilirler.

### **3.2.10 Nokta ortaNokta() Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç ve bitiş noktalarının orta noktasını hesaplar ve bu noktayı temsil eden bir "Nokta" nesnesini döndürür. Kullanıcılar, bu fonksiyon aracılığıyla

Ödev No: 1	Tarih 11.12.2022	9/16
------------	------------------	------

---

doğru parçasının orta noktasını elde edebilir ve bu bilgiyi program içinde kullanabilirler.

### **3.2.11 string toString() Fonksiyonu**

Bu fonksiyon, doğru parçasının başlangıç ve bitiş noktalarını birleştirerek, doğru parçasını bir string ifadesine dönüştürür. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının string temsilini elde edebilir ve bu bilgiyi program içinde kullanabilirler. Bu string temsil, genellikle doğru parçasının başlangıç ve bitiş noktalarını içerir, örneğin "NoktaA-NoktaB".

### **3.2.12 void yazdir() Fonksiyonu**

Bu fonksiyon, doğru parçasını ekrana yazdırmak amacıyla kullanılır. Kullanıcılar, bu fonksiyon aracılığıyla doğru parçasının bilgilerini ekrana basabilirler. Bu, program içindeki doğru parçasının görsel temsilini sağlamak için kullanılır ve genellikle kullanıcıya bilgi vermek veya hata ayıklama süreçlerinde yardımcı olmak için kullanılır.

## **3.3 Daire Sınıfı**

Bu sınıf, merkez noktası ve yarıçap bilgisiyle bir daireyi temsil eder. İki farklı constructor ile daire oluşturabilir: biri merkez ve yarıçapı kullanarak, diğeri başka bir daireyi ve bir çarpanı alarak. Sınıf, dairenin alanını ve çevresini hesaplayabilme yeteneğine sahiptir. Kesişim durumunu kontrol eden bir metodu bulunmaktadır. Son olarak, toString metodu, dairenin merkez noktası ve yarıçap bilgisini birleştirip bir string olarak döndürür.

### **3.3.1 Özellikler**

#### **3.3.1.1 merkez üye değişkeni**

"Daire" sınıfında kullanılan bir Nokta nesnesini temsil eder. Bu nokta, dairenin geometrik konumunu belirtir ve x, y koordinatlarıyla tanımlanır. "Merkez" üye değişkeni, dairenin oluşturulması ve geometrik işlemlerde kullanılması için merkez noktasının bilgisini saklar.

#### **3.3.1.2 yarıçap üye değişkeni**

"Daire" sınıfında kullanılan bir double türündeki değeri temsil eder ve bir dairenin merkezinden kenarına olan uzaklığını belirtir. Bu üye değişkeni, dairenin boyutunu ve geometrik özelliklerini tanımlamak için kullanılır. Dairenin oluşturulması ve çeşitli hesaplamalarda kullanılması için yarıçap bilgisini içerir.

### **3.3.2 Daire(Nokta merkez, double yarıçap) Yapıcı Fonksiyonu**

Bu fonksiyon, bir Nokta nesnesi ve bir double değeri parametre olarak alır. Parametreler,

Ödev No: 1	Tarih 11.12.2022	10/16
------------	------------------	-------

---

dairenin merkez noktasını (x, y koordinatları) ve yarıçapını belirler. Yapıcı fonksiyon, girilen değerlere dayalı olarak bir daire nesnesi oluşturur ve bu daireyi tanımlayan temel geometrik bilgileri içerir.

### **3.3.3 Daire(Daire &daire) Yapıcı Fonksiyonu**

Bu fonksiyon, bir diğer "Daire" nesnesini referans olarak alır. Alınan dairenin merkez noktası ve yarıçap bilgilerini kullanarak yeni bir daire nesnesi oluşturur. Bu şekilde, mevcut bir dairenin özellikleri temel alınarak bir kopya oluşturulabilir.

### **3.3.4 Daire(Daire &daire, int x) Yapıcı Fonksiyonu**

Bu fonksiyon, bir diğer "Daire" nesnesini ve bir tamsayı değerini parametre olarak alır. Alınan dairenin merkez noktası ve yarıçap bilgilerini kullanarak, yarıçapı çarpan değeri ile çarparak yeni bir daire nesnesi oluşturur. Bu sayede, mevcut bir dairenin boyutlarını belirli bir çarpanla artırarak yeni bir daire elde edilebilir.

### **3.3.5 Nokta getMerkez() Fonksiyonu**

Bu fonksiyon, bir dairenin merkez noktasını döndürür. Fonksiyon, Nokta türünde bir nesne olan merkez noktasını ifade eden üye değişkeni içerir. Bu sayede, programın başka kısımlarında dairenin merkez noktasına erişim sağlanabilir ve bu bilgi kullanılabilir.

### **3.3.6 double getYaricap() Fonksiyonu**

Bu fonksiyon, bir dairenin yarıçap değerini döndürür. Fonksiyon, double türünde bir üye değişken olan yarıçapı içerir. Bu sayede, programın diğer bölümlerinde dairenin yarıçapına erişim sağlanabilir ve bu bilgi üzerinde çeşitli hesaplamalar yapılabilir.

### **3.3.7 void setMerkez(Nokta nokta) Fonksiyonu**

Bu fonksiyon, bir dairenin merkez noktasını belirlemek için kullanılır. Fonksiyon, Nokta türünde bir parametre olan "nokta"yı alır ve bu noktayı, daire nesnesinin merkez noktası olarak atar. Böylece, programın diğer kısımlarında dairenin merkezini güncellemek için kullanılabilir.

### **3.3.8 void setYaricap(double yaricap) Fonksiyonu**

Bu fonksiyon, bir dairenin yarıçapını belirlemek için kullanılır. Fonksiyon, double türünde bir parametre olan "yaricap"ı alır ve bu değeri, daire nesnesinin yarıçap üye değişkenine atar.

Ödev No: 1	Tarih 11.12.2022	11/16
------------	------------------	-------

---

Böylece, programın diğer kısımlarında dairenin yarıçapını güncellemek için kullanılabilir.

### **3.3.9 double alan() Fonksiyonu**

Bu fonksiyon, bir dairenin alanını hesaplar. Alan hesaplaması için dairenin yarıçapını kullanarak,  $\pi$  (pi) sayısı ile çarpma işlemi gerçekleştirir ve sonucu döndürür. Bu fonksiyon sayesinde, programın diğer bölümlerinde dairenin alanına kolayca erişim sağlanabilir.

### **3.3.10 double cevre() Fonksiyonu**

Bu fonksiyon, bir dairenin çevresini hesaplar. Çevre hesaplaması için dairenin yarıçapını kullanarak,  $2\pi$  (2 pi) sayısı ile çarpma işlemi gerçekleştirir ve sonucu döndürür. Bu fonksiyon sayesinde, programın diğer bölümlerinde dairenin çevresine kolayca erişim sağlanabilir.

### **3.3.11 int kesisim(Daire daire) Fonksiyonu**

Bu fonksiyon, bir dairenin başka bir daire ile kesişip kesişmediğini kontrol eder. İki dairenin yarıçap ve merkez konumları karşılaştırılarak, kesişim durumu tespit edilir. Fonksiyon, kesişim durumunu belirten bir tamsayı değeri döndürür: 0 (kesişim yok), 1 (daireler tamamen örtüşüyor), ve 2 (diğer durumlar). Bu sayede, programın diğer kısımlarında daireler arasındaki kesişim durumu hakkında bilgi elde edilebilir.

### **3.3.12 string toString() Fonksiyonu**

Bu fonksiyon, bir dairenin merkez noktası ve yarıçap bilgisini birleştirip bir string olarak döndürür. Bu sayede, dairenin özelliklerini anlamak için kullanışlı bir temsil elde edilir. Programın diğer bölümlerinde, dairenin bilgilerini yazdırmak veya başka bir şekilde kullanmak için bu string temsili kullanılabilir.

### **3.3.13 void yazdir() Fonksiyonu**

Bu fonksiyon, bir dairenin bilgilerini ekrana yazdırmak için kullanılır. Fonksiyon, "toString" fonksiyonunu çağırarak dairenin özelliklerini bir string olarak elde eder ve ardından bu stringi ekrana yazdırır. Bu sayede, programın çalışma anında dairelerin bilgilerini görsel olarak gözlemlemek veya hata ayıklamak için kullanılabilir.

## **3.4 Üçgen Sınıfı**

Bu sınıf, üç noktayı içerir ve üçgenin özelliklerini hesaplamak için çeşitli fonksiyonlara sahiptir. "alan" fonksiyonu, üçgenin alanını hesaplar; "cevre" fonksiyonu, üçgenin çevresini

Ödev No: 1	Tarih 11.12.2022	12/16
------------	------------------	-------

---

döndürür. "acilar" fonksiyonu, üçgenin iç açılarını derece cinsinden hesaplar. Ayrıca, "toString" fonksiyonu, üçgenin bilgilerini içeren bir string döndürerek nesnenin metinsel temsilini sağlar. Bu sınıf, üçgenlerin geometrik özelliklerini doğru bir şekilde hesaplamak için matematiksel formülleri kullanır.

### **3.4.1 Özellikler**

#### **3.4.1.1 a üye değişkeni**

"Ucgen" sınıfında üçgenin birinci noktasını temsil eder. Bu değişken, üçgenin geometrik konumunu belirlemek ve işlemek için kullanılır. Üçgenin köşe noktalarını tutan bu değişken, üçgen objesinin temel yapı taşlarından biridir.

#### **3.4.1.2 b üye değişkeni**

"Ucgen" sınıfında üçgenin ikinci noktasını temsil eder. Bu değişken, üçgenin geometrik konumunu belirlemek ve işlemek için kullanılır. Üçgenin köşe noktalarını tutan bu değişken, üçgen objesinin temel yapı taşlarından biridir.

#### **3.4.1.3 c üye değişkeni**

"Ucgen" sınıfında üçgenin üçüncü noktasını temsil eder. Bu değişken, üçgenin geometrik konumunu belirlemek ve işlemek için kullanılır. Üçgenin köşe noktalarını tutan bu değişken, üçgen objesinin temel yapı taşlarından biridir.

### **3.4.2 Ucgen(Nokta a, Nokta b, Nokta c) Yapıcı Fonksiyonu**

Bu fonksiyon, üçgenin köşe noktalarını belirleyerek, üçgen objesinin başlangıç durumunu oluşturur. Yapıcı fonksiyon, sınıfın başka fonksiyonları tarafından kullanılarak üçgenin geometrik özelliklerini işlemek için temel bilgileri sağlar.

### **3.4.3 Nokta getA() Fonksiyonu**

Bu fonksiyon, üçgen objesinin oluşturulduğu noktalar arasındaki birinci noktanın bilgisini sağlar. Bu bilgi, daha sonra sınıfın diğer fonksiyonları tarafından kullanılarak üçgenin çeşitli özelliklerinin hesaplanmasında kullanılır.

### **3.4.4 Nokta getB() Fonksiyonu**

Bu fonksiyon, üçgen objesinin oluşturulduğu noktalar arasındaki ikinci noktanın bilgisini sağlar. Bu bilgi, daha sonra sınıfın diğer fonksiyonları tarafından kullanılarak üçgenin çeşitli özelliklerinin hesaplanmasında kullanılır.

Ödev No: 1	Tarih 11.12.2022	13/16
------------	------------------	-------

---

### 3.4.5 Nokta getC() Fonksiyonu

Bu fonksiyon, üçgen objesinin oluşturulduğu noktalar arasındaki üçüncü noktanın bilgisini sağlar. Bu bilgi, daha sonra sınıfın diğer fonksiyonları tarafından kullanılarak üçgenin çeşitli özelliklerinin hesaplanmasında kullanılır.

### 3.4.6 void setA(Nokta nokta) Fonksiyonu

Bu fonksiyon, Nokta sınıfından bir nesne alarak, üçgenin birinci noktasını bu nesneye eşitleyerek belirler. Bu işlem, üçgen objesinin geometrik özelliklerini güncellemek ve yeniden hesaplamak için önemlidir. Yani, "setA" fonksiyonu, üçgenin birinci noktasını değiştirmek amacıyla kullanılır ve sınıfın içindeki diğer fonksiyonlar tarafından çağrılabilir.

### 3.4.7 void setB(Nokta nokta) Fonksiyonu

Bu fonksiyon, Nokta sınıfından bir nesne alarak, üçgenin ikinci noktasını bu nesneye eşitleyerek belirler. Bu işlem, üçgen objesinin geometrik özelliklerini güncellemek ve yeniden hesaplamak için önemlidir. Yani, "setB" fonksiyonu, üçgenin ikinci noktasını değiştirmek amacıyla kullanılır ve sınıfın içindeki diğer fonksiyonlar tarafından çağrılabilir.

### 3.4.8 void setC(Nokta nokta) Fonksiyonu

Bu fonksiyon, Nokta sınıfından bir nesne alarak, üçgenin üçüncü noktasını bu nesneye eşitleyerek belirler. Bu işlem, üçgen objesinin geometrik özelliklerini güncellemek ve yeniden hesaplamak için önemlidir. Yani, "setC" fonksiyonu, üçgenin üçüncü noktasını değiştirmek amacıyla kullanılır ve sınıfın içindeki diğer fonksiyonlar tarafından çağrılabilir.

### 3.4.9 string toString() Fonksiyonu

Bu fonksiyon, üçgenin köşe noktalarının koordinatlarını içeren bir metinsel temsil oluşturur. Bu sayede, üçgen objesinin bilgileri kolayca okunabilir bir formatta elde edilir. "toString" fonksiyonu, üçgenin çeşitli özelliklerini görüntülemek veya raporlamak amacıyla kullanılır.

### 3.4.10 double alan() Fonksiyonu

Bu fonksiyon, üçgenin köşe noktalarını kullanarak üçgenin alanını matematiksel bir formülle hesaplar. Ardından, hesaplanan alan değerini geri döndürerek, üçgenin geometrik özelliklerini temsil eden önemli bir bilgi sunar. "alan" fonksiyonu, üçgenin alanını hesaplamak ve bu değeri kullanarak çeşitli matematiksel operasyonlar gerçekleştirmek amacıyla kullanılır.

Ödev No: 1	Tarih 11.12.2022	14/16
------------	------------------	-------

---

### 3.4.11 double cevre() Fonksiyonu

Bu fonksiyon, üçgenin köşe noktaları arasındaki uzunlukları kullanarak üçgenin toplam çevresini hesaplar. Sonuç olarak, üçgenin çevresini temsil eden bir sayı değeri elde edilir. "cevre" fonksiyonu, üçgenin dış hatlarını belirlemek ve çeşitli geometrik hesaplamalarda kullanılmak üzere tasarlanmıştır.

### 3.4.12 double\* acilar() Fonksiyonu

Bu fonksiyon, üçgenin her bir iç açısını hesaplamak için trigonometrik formülleri kullanır. Hesaplanan açı değerleri, dinamik olarak oluşturulan bir double dizisi içinde saklanır ve bu dizi fonksiyon tarafından geri döndürülür. "acilar" fonksiyonu, üçgenin geometrik özelliklerini daha geniş bir bağlamda anlamak ve analiz etmek amacıyla kullanılır.

## 4. SONUÇ VE ÖĞRENİLEN DERSLER

Bu proje, C++ programlama dili kullanılarak oluşturulmuş dört adet geometrik sınıfı incelemektedir. Bu sınıflar, bir noktanın x ve y koordinatlarını tutan Nokta sınıfı, bir doğru parçasının iki noktasını tutan DogruParcasi sınıfı, bir dairenin merkezini ve yarıçapını tutan Daire sınıfı ve üç noktayı tutan Ucgen sınıfıdır.

Bu sınıflar, nesne yönelimli programlama (OOP) prensiplerini kullanarak tasarlanmıştır. OOP, nesneleri ve aralarındaki ilişkileri tanımlamak için kullanılan bir yazılım geliştirme yaklaşımıdır. Bu yaklaşım, programlamayı daha anlaşılır, sürdürülebilir ve tekrar kullanılabilir hale getirir.

Bu sınıflar, OOP prensiplerini aşağıdaki şekilde yansıtmaktadır:

**Kapsülleme:** Sınıflar, verileri ve yöntemlerini bir arada saklar. Bu, verilerin güvenliğini ve gizliliğini sağlar.

**Soyutlama:** Sınıflar, nesnelerin gerçek uygulamasını ayrıntılara inmeden tanımlar. Bu, sınıfları daha esnek ve yeniden kullanılabilir hale getirir.

**İnheritans:** Sınıflar, diğer sınıfların özelliklerini ve davranışlarını miras alabilir. Bu, sınıfların yeniden kullanımını ve kod tekrarını azaltır.

Ödev No: 1	Tarih 11.12.2022	15/16
------------	------------------	-------

---

Polimorfizm: Sınıflar, aynı adı taşıyan birden fazla yöntemi tanımlayabilir. Bu, programlamayı daha esnek ve güçlü hale getirir.

Bu sınıflar, OOP prensiplerini kullanarak, belirli bir geometrik şekli temsil etmek için gerekli olan verileri ve yöntemleri sağlar. Bu sınıflar, C++ programlama dilini kullanarak geometrik şekillerle çalışmak için güçlü ve esnek bir temel sağlar.

## 5. KAYNAKÇA

- <https://www.quantstart.com/articles/Mathematical-Constants-in-C/>

**Bu Github Profillerimiz :**

**Yavuzhan Kurşun** : <https://github.com/yavuzhankursun>

**İbrahim Keçici** : <https://github.com/ibrahimibo7>

Ödev No: 1	Tarih 11.12.2022	16/16
------------	------------------	-------