**T.C**

**KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**GEOMETRİK PROBLEMLER**

**HAZIRLAYAN:**

**Müslüm Batın KILIÇ**

**220501023**

**BERKE ÇETİN**

**220501021**

**DERS SORUMLUSU:**

**PROF. DR. HÜSEYİ TARIK DURU**

**2 OCAK 2024**

**İÇİNDEKİLER**

[1. ÖZET 3](#_Toc149597653)

[2. GİRİŞ 3](#_Toc149597654)

[3. YÖNTEM.............................................................................................................................................4](#_Toc149597655)

[3.1 NOKTA SINIFI....................................................................................................................................4](#_Toc149597655)

[3.2 DOĞRUPARÇASI SINIFI.................................................................................................................................4](#_Toc149597655)

[3.3 DAİRE SINIFI....................................................................................................................................................5](#_Toc149597655)

[3.4 ÜÇGEN SINIFI...................................................................................................................................................5](#_Toc149597655)

[4. SONUÇ VE ÖĞRENİLEN DERSLER](#_Toc149597666) 6

[5. KAYNAKÇA](#_Toc149597667) 6

6. GİTHUP LİNK.....................................................................................................................................................6

# ÖZET

Bu rapor, C++ programlama dili kullanılarak geometrik şekilleri temsil eden sınıfların oluşturulduğu bir proje üzerine odaklanmaktadır. Proje içerisinde Nokta, DogruParcasi, Daire ve Ucgen sınıfları bulunmaktadır. Her bir sınıf, geometrik şekillerin özelliklerini hesaplamak ve temsil etmek için gerekli olan çeşitli metotları içermektedir.

Nokta sınıfı, iki boyutlu bir noktayı temsil eder ve konumunu x ve y koordinatları ile belirler. DogruParcasi sınıfı, iki Nokta nesnesini kullanarak bir doğru parçasını tanımlar ve bu parça üzerinde çeşitli hesaplamalar yapar. Daire sınıfı, bir merkez noktası ve yarıçapı ile bir daireyi temsil eder; bu dairenin alanını, çevresini ve başka bir daire ile olan kesişim durumunu kontrol eder. Ucgen sınıfı ise üç noktayı kullanarak bir üçgeni tanımlar ve üçgenin alanını, çevresini ve açılarını hesaplar.

Proje, nesne tabanlı programlama prensiplerini kullanarak her geometrik şeklin kendi özelliklerini ve hesaplamalarını içeren bir yapı oluşturmayı amaçlamaktadır. Bu sayede, farklı geometrik şekillerin özelliklerini hesaplamak ve ilişkilendirmek daha modüler ve anlaşılır bir şekilde gerçekleştirilebilir.

# GİRİŞ

Bu çalışma, geometrik şekillerin temsil edildiği bir C++ programlama projesine odaklanmaktadır. Genel amacı, nesne tabanlı programlama prensiplerini kullanarak farklı geometrik şekillerin tanımlanmasını, özelliklerinin hesaplanmasını ve bu şekiller arasındaki ilişkilerin modüler bir yapı içinde yönetilmesini sağlamaktır. Çalışmanın ana hedefi, her bir geometrik şeklin kendi özel sınıfında temsil edilmesi ve bu sınıflar arasında mantıklı ilişkiler kurularak genişletilebilir bir yapı oluşturmaktır.

Başlıca kullanılan yöntem, C++ programlama dilinin nesne tabanlı özelliklerinden yararlanmaktır. Her geometrik şekil için ayrı bir sınıf tasarlanmış ve bu sınıfların içerisinde şekillere özgü özellikleri hesaplayan metotlar bulunmaktadır. Ayrıca, geometrik şekiller arasındaki ilişkileri yönetmek amacıyla uygun metotlar ve sınıflar tasarlanmıştır. Bu sayede, her geometrik şekil bağımsız bir yapıya sahiptir ve genel program üzerinde birbirinden bağımsız bir şekilde kullanılabilir.

Çalışmanın temel tezleri arasında, nesne tabanlı programlamanın geometrik şekillerin modellemesi ve yönetimi için etkili bir yaklaşım olduğu vurgulanmaktadır. Her bir sınıfın kendi içinde bağımsız çalışabilir olması, genişletilebilirliği artırmakta ve kodun anlaşılabilirliğini sağlamaktadır. Ayrıca, geometrik şekiller arasındaki ilişkilerin düzenli bir şekilde yönetilmesi, programın daha modüler ve sürdürülebilir olmasını sağlamaktadır.

Bu çalışma, geometrik şekillerin temsilini ve işlenmesini kolaylaştırmak için kullanılan bir programlama yaklaşımını detaylı bir şekilde açıklamaktadır. Her bir sınıfın görev ve sorumluluklarını belirlemek, genel programın anlaşılabilirliğini artırmak ve geometrik şekillerle ilgili işlemleri daha etkili bir şekilde gerçekleştirmek amacıyla yapılmıştır.

# YÖNTEM

Projede benimsenen ana metodun nesne tabanlı programlama ilkeleri ile nasıl uygulandığına odaklanılacaktır. Her bir geometrik şekil sınıfı, sınıflar arası ilişkileri ve işlevselliği yöneten nesne tabanlı programlama prensiplerini içermektedir. Örneğin, her geometrik şekilin kendi sınıfında tanımlanmış özellikleri ve metotları bulunmakta, bu da her bir sınıfın birbirinden bağımsız çalışabilmesini sağlamaktadır.

Proje, C++ programlama dilinde geliştirilmiştir ve nesne tabanlı programlama prensiplerine uygun olarak tasarlanmıştır. C++'ın sınıf ve nesne yapısından yararlanılarak her geometrik şekil için ayrı bir sınıf oluşturulmuştur. Ayrıca, geometrik hesaplamalarda standart kütüphane fonksiyonları (örneğin,sqrt ve cos) kullanılarak doğruluk ve verimlilik sağlanmıştır.

Program, aşağıdaki temel sınıfları içermektedir:

* 1. **NOKTA SINIFI:**

Nokta sınıfı, iki boyutlu bir düzlemdeki bir noktayı temsil eder. X ve y koordinatlarına sahip bir nesnedir.

Nokta sınıfının özellikleri şunlardır:

* x: Noktanın x koordinatı.
* y: Noktanın y koordinatı.

Nokta sınıfının yöntemleri şunlardır:

* mesafe(Nokta diğerNokta): Bu yöntem, iki nokta arasındaki mesafeyi hesaplar.
* yazdir(): Bu yöntem, noktanın x ve y koordinatlarını yazdırır.
  1. **DOĞRU PARÇASI SINIFI:**

Doğru Parçası sınıfı, iki nokta arasındaki doğru parçasını temsil eder. Başlangıç ve bitiş noktalarına sahiptir.

Doğru Parçası sınıfının özellikleri şunlardır:

* baslangicNoktasi: Doğru parçasının başlangıç noktası.
* bitisNoktasi: Doğru parçasının bitiş noktası.

Doğru Parçası sınıfının yöntemleri şunlardır:

* uzunluk(): Bu yöntem, doğru parçasının uzunluğunu hesaplar.
* ortaNoktasi(): Bu yöntem, doğru parçasının orta noktasını döndürür.
* kesisimNoktasi(Nokta nokta): Bu yöntem, doğru parçası ile verilen noktayı kestiği noktayı döndürür.
* yazdir(): Bu yöntem, doğru parçasının başlangıç ve bitiş noktalarını yazdırır.
  1. **DAİRE SINIFI:**

Daire sınıfı, bir merkez ve bir yarıçapa sahip bir daireyi temsil eder.

Daire sınıfının özellikleri şunlardır:

* merkez: Dairenin merkezi.
* yaricap: Dairenin yarıçapı.

Daire sınıfının yöntemleri şunlardır:

* alan(): Bu yöntem, dairenin alanını hesaplar.
* cevre(): Bu yöntem, dairenin çevresini hesaplar.
* kesisim(Daire diğerDaire): Bu yöntem, iki dairenin kesişim durumunu döndürür.
* yazdir(): Bu yöntem, dairenin merkezini ve yarıçapını yazdırır.
  1. **ÜÇGEN SINIFI:**

Üçgen sınıfı, üç nokta arasındaki üçgeni temsil eder. Noktalarına sahiptir.

Üçgen sınıfının özellikleri şunlardır:

* nokta1: Üçgenin ilk noktası.
* nokta2: Üçgenin ikinci noktası.
* nokta3: Üçgenin üçüncü noktası.

Üçgen sınıfının yöntemleri şunlardır:

* alan(): Bu yöntem, üçgenin alanını hesaplar.
* cevre(): Bu yöntem, üçgenin çevresini hesaplar.
* acilar(): Bu yöntem, üçgenin açılarını döndürür.
* yazdir(): Bu yöntem, üçgenin noktalarını yazdırır.

Programın çalışma mantığı şu adımları içerir:

* Her bir sınıf için parametresiz yapıcı, tek parametreli yapıcı, iki parametreli yapıcı, kopya yapıcı gibi farklı yapıcılar tanımlanmıştır.
* Her sınıf, belirli geometrik özelliklere sahip olduğundan, bu özelliklere uygun metotlar içerir. Örneğin, Nokta sınıfı uzaklık hesaplamaları yapabilirken, DogruParcasi sınıfı kesişim noktasını bulabilir.
* Main fonksiyonu içinde, her bir sınıfın özelliklerini ve fonksiyonlarını test etmek amacıyla çeşitli örnekler oluşturulmuştur.

# SONUÇ VE ÖĞRENİLEN DERSLER

Bu program, C++ programlama dilinde nesne yönelimli programlama (OOP) prensiplerini kullanarak geometrik şekillerin temsil edilmesi ve üzerlerinde çeşitli matematiksel hesaplamaların yapılmasını amaçlamaktadır. Programın sonuçları şu şekildedir:

* Sınıflar arasında doğru bir ilişki kurulmuş ve her bir sınıfın kendi özelliklerini ve işlevselliğini sağlayacak şekilde tasarlanmıştır.
* Her bir sınıf, kendi içinde çeşitli metotlar ve fonksiyonlar içerir ve bu fonksiyonlar geometrik hesaplamaları gerçekleştirir.
* Main fonksiyonu içinde yapılan örnek kullanımlar, her bir sınıfın doğru çalıştığını göstermektedir.
* Programın yazımı sırasında, nesne yönelimli programlama prensipleri, sınıf tasarımı ve inheritance gibi konularda daha fazla anlayış geliştirilmiştir.
* C++ dilindeki standart kütüphane kullanılarak matematiksel hesaplamalar ve string manipülasyonları gerçekleştirilmiştir.

# KAYNAK

<https://www.w3schools.com/cpp/cpp_oop.asp>

<https://www.w3schools.com/cpp/cpp_classes.asp>

<https://www.w3schools.com/cpp/cpp_functions.asp>

https://tr.wikipedia.org/wiki/C%2B%2B

# GİTHUP LİNK

MÜSLÜM BATIN KILIÇ: https://github.com/batinkilic?tab=repositories

BERKE ÇETİN: https://github.com/berke10?tab=repositories