**KOCaELİ Sağlık ve teknoloji üniversitesi**

**Yazılım Laboratuvar Ders Ödevi**

**Konu: Çizim Uygulaması**

**Hazırlayanlar: Sude DEMİRTAŞ /İclal Çengel**

**Bölüm:Bilgisayar Mühendisliği**

**Okul No:210501012 /21050013**

**4 KASIM 2023**

İçindekiler

[1 AMAÇ 3](#_Toc150351066)

[2 Proje Detayı 4](#_Toc150351067)

[3 Puan Listesi 5](#_Toc150351068)

[4 Program Dili 6](#_Toc150351069)

[5 Projeye Başlamadan Önce Neler Yaptık 7](#_Toc150351070)

[6 Proje Sayesinde Yeni Öğrendiğimiz Şeyler 8](#_Toc150351071)

[7 Projede Zorlandığımız Kısımlar 9](#_Toc150351072)

[8 Sonuç 10](#_Toc150351073)

[9 Kaynakça 12](#_Toc150351074)

# AMAÇ

Bu projenin amacı basit bir çizim uygulaması yapmamız istenmektedir.

# Proje Detayı

* Uygulama da çizim yapabilmek için öncelikle çizimi yapılacak şeklin seçilmesi gerekmektedir.Programda altı farklı şekil çizilebilmelidir.Seçilen şeklin görüntüsü diğer şekillerden farklı olmalıdır. Çizim işlemine farenin sol tuşuna basık tutarak başlanır ve bu tuş bırakılana kadar çizim devam eder.Fare tuşuna basılı tutulduktan sonra fare hareketine göre çizilecek şeklin boyutları belirlenir.
* Şekiller çizim alanının dışına taşmamalıdır.Çizim yapılmadan önce çizilecek şeklin rengi seçilebilmelidir. Bu işlem için şekillerdeki gibi bir renk paneli tasarlanmalıdır.Şekil panelinde olduğu gibi seçilen renk diğerlerinden ayırt edilebilmelidir.
* Şekil seç aracı ile çizim alanındaki şekiller seçilebilmelidir. İstenirse seçilen şeklin rengi değiştirilebilmeli veya çizim alanından silinebilmelidir.Seçme işlemi için öncelikle seçim ikonuna tıklanmalı ardından istenilen şekil seçilmelidir.Seçilen şekil aşağıda görüldüğü gibi diğer şekillerden ayırt edilebilmelidir.Şekil seçildikten sonra renk paneli kullanılarak rengi değiştirilebilmeli veya silme ikonuna tıklanarak silinebilmelidir.
* Dosya menüsünde kaydet ve aç ikonları bulunmalıdır.Kaydet ikonuna tıklandığında çizim alanı üzerinde çizilmiş olan bütün şekiller kullanıcının belirlediği bir dosyaya kaydedilecektir. (Şekle ait olan özelliklerin kaydedilmesi gerekir. Başlangıç koordinatları, boyutlar, renk vb.)
* Dosya Oku ikonuna tıklandığında öncelikle dosya aç diyalog (OpenFileDialog) paneli kullanıcının karşısına çıkacaktır. Kullanıcı bu paneli kullanarak açılmasını istediği (içerisi daha önceden kaydedilmiş şekillere ait bilgileri barındıran) bir dosyayı seçecektir. Seçilen dosya içerisindeki bilgiler kullanılarak bütün şekiller oluşturulup çizim alanında çizilecektir.
* Dosya aç ikonuna tıklandığında aşağıdaki gibi OpenFileDialog paneli kullanıcının karşısına çıkmalıdır.
* Daha önce bahsedilen menülere benzer bir text menüsü oluşturulması beklenmektedir.Yazı butonuna basıldığında çizim alanı üzerinde istenen yazıyı yazdırabilecek bir yapı oluşturulması istenmektedir.
* Yazı butonunun istenen yazısı: ABCÇDEFGĞHIİJKLMNOÖPRSŞTUÜVYZ Kalem butonuna basıldığında çizim alanı üzerinde istenen yazıyı yazmaya olanak sağlaması istenmektedir.

# Puan Listesi

* Şekil menüsü 20 puan
* Renk menüsü 15 puan
* İşlem menüsü 15 puan
* Dosya menüsü 15 puan
* Yazılan kodlara olan hakimiyet 35 puan
* Bonus soru (Text menüsü) 15-20 puan
* Ekstra özellikler (15 puan)

# Program Dili

C++ programlama dili, yüksek seviyeli bir genel amaçlı programlama dilidir. C++'ın temeli, C programlama diline dayanırken, ona nesne yönelimli programlama (OOP) özelliklerini ekler. Bu nedenle C++ hem prosedürel programlama hem de nesne yönelimli programlama paradigmasını destekler.

Nesne Yönelimli Programlama (OOP): C++'ın en önemli özelliklerinden biri, nesne yönelimli programlamayı desteklemesidir. Bu, programın yapılandırılmasını, veri ve işlevlerin bir araya getirilmesini ve tekrar kullanılabilirlik ve modülerlik sağlamak için sınıflar ve nesnelerin kullanılmasını içerir.

Genişletilebilirlik: C++ dilinin bir başka önemli özelliği, diğer programlama dilleriyle uyumlu olması ve kullanıcıların kendi veri türlerini ve fonksiyonlarını tanımlayabilmesidir. Bu, C++'ın genişletilebilir bir dil olmasını sağlar ve farklı programlama ihtiyaçlarına uygun çözümler sunar.

Verimlilik: C++ dili, düşük seviye programlama yetenekleri sunar ve bellek yönetimi üzerinde doğrudan kontrol sağlar. Bu, C++'ı verimli ve hızlı çalışan uygulamalar geliştirmek için tercih edilen bir dil haline getirir.

Şablonlar: C++ dilinde şablonlar, genel programlama teknikleri için kullanılır. Şablonlar, türden bağımsız bir şekilde çalışan fonksiyonlar ve sınıflar oluşturmanızı sağlar. Bu, kod tekrarını azaltır ve farklı veri tipleriyle çalışabilen esnek yapılar oluşturmanızı sağlar.

Taşınabilirlik: C++ dili, platformdan bağımsız bir şekilde çalışabilir. Bu, C++'ın farklı işletim sistemleri ve donanım mimarileri üzerinde çalışabilme yeteneği sağlar. Bu da C++'ı taşınabilir ve çok platformlu uygulamaların geliştirilmesinde tercih edilen bir dil yapar.

# Projeye Başlamadan Önce Neler Yaptık

İlk olarak projenin ana hedeflerini ve kapsamını netleştirmek için bir proje tanımı ve proje planı oluşturduk. Bu adımda, projenin neyi başarmayı hedeflediğini, hangi sonuçları elde etmek istediğimizi ve proje sürecinin nasıl ilerleyeceğini belirledik.Proje için gerekli kaynakları belirledik ve planladık. Projede yer alan tüm paydaşlar arasında etkili iletişimi sağlamak için bir iletişim planı oluşturduk. Bu plan, iletişim kanallarını, toplantılarını ve raporlama süreçlerini belirlemeyi içerir. Ayrıca paydaşların bilgilendirilmesi ve güncel tutulması için iletişim stratejileri geliştirdik.Proje sürecini yönetmek ve ilerlemeyi takip etmek için bir zaman çizelgesi oluşturduk. Bu zaman çizelgesi, projenin başlangıç ve bitiş tarihlerini, aşamalarını ve kilometre taşlarını içerir. Böylece proje ilerlemesini takip edebilir ve gerekirse planı güncelleyebiliriz. Projeye başlamadan önce gereken kaynakları temin ettik veya hazırladık. Bu kaynaklar, proje için gerekli olan kodumuzda kulanıcağımız kütüphaneleri içerebilir.Bu konuyu anlamak için paint uygulamsını inceledik.Kullanacağımız kütüphaneleri nerden indiriceğimizi ve nasıl ekleyebilceğimize dair videolar bulup izledik.

# Proje Sayesinde Yeni Öğrendiğimiz Şeyler

Projemizin konusuyla ilgili olarak derinlemesine bir bilgi birikimi edindik. Araştırma yaparken, konuyla ilgili güncel trendleri, en iyi uygulamaları ve sektördeki gelişmeleri yakından takip ettik. Bu sayede, konu hakkında daha geniş bir perspektif kazandık ve uzmanlığımızı artırdık.

Bu projeyi yönetme sürecinde, proje yönetimi becerilerimizi geliştirdik. Zaman yönetimi, kaynak planlaması, risk yönetimi ve iletişim becerileri gibi önemli becerileri pratiğe döktük. Proje takvimini oluşturma, etkili iletişim kurma konularında daha yetkin hale geldik.

Projede kullanılan belirli teknikler veya araçlar üzerinde çalışarak yeni teknik yetkinlikler kazandık. Örneğin, proje yönetimi yazılımları, veri analizi araçları veya tasarım programları gibi özel araçlarla çalışarak bu alanlardaki becerilerimizi artırdık. Bu projenin gerektirdiği teknikleri öğrenerek, kendimizi teknolojik açıdan da geliştirdik.

Projede bir ekip olarak çalışarak takım çalışması becerilerimizi geliştirdik. Ekip üyeleriyle etkili iletişim kurma, görevleri koordine etme, işbirliği yapma ve çözüm odaklı çalışma gibi yetenekleri pekiştirdik. Birlikte çalışmanın önemini ve başarılı bir şekilde işbirliği yapmanın yollarını öğrendik.

Projenin farklı aşamalarında karşılaştığımız sorunlarla uğraşarak problem çözme yeteneklerimizi geliştirdik. Yaratıcı çözümler bulma, analitik düşünme, karar verme ve esneklik gibi becerilerimizi kullanarak zorlukları aştık. Bu deneyimler, gelecekteki projelerde karşılaşabileceğimiz sorunlara daha etkili bir şekilde yaklaşmamızı sağladı.

Projeyi tamamlarken, elde ettiğimiz sonuçları etkili bir şekilde sunma ve raporlama becerilerimizi geliştirdik.

# Projede Zorlandığımız Kısımlar

Bu projede, C++ dilinde basit bir çizim uygulaması geliştirme sürecinde yaşadığımız sorunlar ve karşılaştığımız olumsuz deneyimleri paylaşmak istiyoruz.Proje için kullanmayı planladığımız grafik kütüphanesiyle ilgili yaşadığımız sorunlar oldukça can sıkıcıydı. Kütüphanenin doğru bir şekilde entegre edilmesi ve kullanılması için gerekli dökümantasyon eksikti. Bu durum, projenin başlangıcında zaman kaybına yol açtı ve projenin ilerlemesini olumsuz etkiledi.Projeye uygun olmayan bir dilin seçilmesi, projenin başarısını olumsuz etkileyebilir. Bu nedenle, projenin gereksinimlerine en uygun dilin seçilmesi ve bu dilin projede karşılaşılan zorlukları minimize etmesi büyük önem taşır. Bu durumda, projemizin daha etkili ve verimli bir şekilde geliştirilmesi için uygun bir programlama dili seçimi yapılması gerektiğini düşünüyoruz.

# Sonuç

Bu proje, C++ programlama dilini daha iyi anlama ve kullanma amacıyla gerçekleştirildi. Projemizin sonucunda aşağıdaki başarıları elde ettik:

C++ Dilinde Yetkinlik: Proje boyunca C++ dilini daha iyi anlama ve kullanma becerilerimizi geliştirdik. Dilin temel yapılarını, sözdizimini ve en iyi uygulamalarını daha iyi anladık.

Nesne Yönelimli Programlama (OOP): Projede C++'ın nesne yönelimli programlama (OOP) özelliklerini uygulayarak, daha yapılandırılmış ve modüler kodlar oluşturduk. Sınıflar, nesneler, kalıtım, çoklu kalıtım, polimorfizm gibi OOP kavramlarını kullanarak programları daha esnek ve sürdürülebilir hale getirdik.

Verimli ve Performanslı Kodlama: C++ dilinin doğrudan bellek yönetimi ve düşük seviye kontrol yeteneklerini kullanarak verimli ve performans odaklı kodlama yapma becerilerimizi geliştirdik. Bellek kullanımını optimize etme, veri yapılarını etkin kullanma ve algoritmalarda hızlı çalışma prensiplerini uygulama konusunda ilerleme kaydettik.

Şablonlar ve Genel Programlama: Projede C++'ın şablonlar özelliğini kullanarak genel programlama tekniklerini uyguladık. Şablonlar sayesinde türden bağımsız kod yazma yeteneğini kazandık ve daha esnek ve yeniden kullanılabilir fonksiyonlar ve sınıflar oluşturabildik.

Hata Ayıklama ve Sorun Giderme: Projeyi gerçekleştirirken, hata ayıklama ve sorun giderme becerilerimizi geliştirdik. Hataları bulma, hata ayıklama araçlarını kullanma ve kodu optimize etme konularında deneyim kazandık. Bu da daha sağlam ve hatasız kodlar oluşturmamıza yardımcı oldu.

Proje sonucunda, Projede zorlandığımız kısımlar olsa da, azim ve kararlılıkla projemize devam ettik. Bu süreçte yaşadığımız zorluklar, bize daha fazla öğrenme fırsatı sundu ve problem çözme yeteneklerimizi geliştirdi. Özellikle projenin başlangıç aşamalarında dil seçimiyle ilgili yaşadığımız tereddütler, uygun bir programlama dili seçimi konusundaki önemli bir ders oldu.

Zorluklarla karşılaştığımızda pes etmek yerine, problem çözme becerilerimizi kullanarak çözümler üretmeye odaklandık. Bu süreçte, eksikliklerimizi fark edip geliştirmek için birbirimize destek olduk. Zorluklar karşısında direnç göstermek ve çözüm odaklı düşünmek, projemizi başarıya taşıma konusundaki kararlılığımızı artırdı.

Yaşadığımız zorluklara rağmen projemizi tamamlamak, bize sadece teknik bilgi ve deneyim kazandırmakla kalmadı, aynı zamanda ekip olarak birlikte çalışabilme ve dayanışma yeteneklerimizi de güçlendirdi. Bu deneyim, gelecekte karşılaşacağımız zorluklarla baş etme yetimizi artırmış ve bize daha güçlü bir şekilde ilerleme motivasyonu sağlamıştır.

Bu projede yaşadığımız zorluklar, öğrenme sürecimizin önemli bir parçası oldu. Deneyimlediğimiz her zorluk, daha güçlü, daha bilgili ve daha dirençli bir ekip olmamıza katkıda bulundu. Bu projede edindiğimiz tecrübelerle gelecekteki projelerimizi daha bilinçli ve etkili bir şekilde yönetme ve başarıya ulaştırma konusundaki güvenimizi artırdı. Bu süreçte yaşadığımız zorlukları aşarak, projemizi başarıyla tamamlamanın gururunu yaşıyoruz**.**

# Kaynakça

**C++ Programlama Dili:**

**"C++ How to Program" (Kitap): Paul Deitel, Harvey Deitel**

**"The C++ Programming Language" (Kitap): Bjarne Stroustrup**

**C++ referans dokümantasyonu:** [**https://en.cppreference.com/**](https://en.cppreference.com/)