**3 Boyutlu Görüntü Dosyaları Projesi**

Osman Şimşek

Bilgisayar Mühendisliği

Kocaeli Üniversitesi

180202048

simsekosman15@hotmail.com

Özet

Bu projede bir klasörün içinde bulunan dosyalar, .nkt uzantılı olarak verilmiştir. Program .nkt uzantılı dosyaların içinde verilen dosya bilgilerini ve ayrı ayrı verilen 3 boyutlu nokta verilerini okur ve noktalar standartlara uygunsa veriler bellekte saklanmak üzere işleme sokulur.

Dosyaların kontrolü yapılıp veriler çekildikten sonra kullanıcıdan yapılması istenen işlemler menü aracılığıyla istenir. Yapılan işlemler

output.nkt olarak dosyaya çıktısı verilir. Yapılan işlemler bittikten sonra kullanıcı programdan

çıkış yapabilir ve işlemlerin kontrolünü

output.nkt dosyalarından

kontrolünü sağlayabilir.

1.Giriş

3 boyutlu görüntü dosyaları projesi kullanıcının vermiş olduğu dosyalardan .nkt uzantısına sahip olanları okuyarak işleme başlar. Dosya kontrolü belirli formatlarda gerçekleştirilir hatalı dosyaların ayıklanması sağlanır. Dosyaları ayıklanmasından sonra dosya içindeki veriler bellekte saklanır. Saklanan verilerin ardından verilmiş olan menü üzerinden kullanıcı tarafından istenen işlem gerçekleştirilerek **output.nkt** dosyasının içerisine sonuçlar yazdırılır.

Dosya okuma ve menü işlemleri ile ilgili gerekli detaylar tasarım kısmında bahsedilmiştir.

Yener Emin Elibol

Bilgisayar Mühendisliği

Kocaeli Üniversitesi

180202054

yenereminelibol@gmail.com

2.TEMEL BİLGİLER

Program **C** programlama dilinde gerçekleştirilmiş olup geliştirilme ortamı olarak **CodeBlocks** kullanılmıştır.

3.TASARIM

*A) Algoritma*

Program çalıştırıldığı zaman bir ara yüz karşılar, kullanıcıya yardımcı olması için işlemlerin hangi sırada gerçekleştirildiğini gösterir. Klasördeki .nkt uzantılı dosyaları tespit edip aldıktan sonra fgets() fonksiyonu ile satır satır okuma işlemi gerçekleştirilip dosyanın kontrolü sağlanıp, sayısal veriler bellekte tutulmaya başlanır. Dosyanın satır satır kontrolü şu şekilde sağlanır:

**1.adım: # Noktalar dosya format**

Eğer ki bu yapı yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez.

**2.adım: VERSİON1**

Eğer ki bu yapı yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez.

**3.adım: ALANLAR x y z / ALANLAR x y z r g b** Dosyanın bu formatında iki farklı tip vardır. Bunlardan bir tanesi sadece x y z koordinat bilgileri içeren format veya x y z koordinat bilgilerine ek olarak r g b renk kodları verilmiş bu adım için verilmiş iki dosya tipi vardır. Eğer ki bu yapı yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez.

**4.adım: NOKTALAR**

Bu adımdaki NOKTALAR değeri dosyanın içinde bulundurduğu nokta adet sayılarını yansıtmaktadır. Eğer ki bu yapı yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez. Eğer doğruysa NOKTALAR stringinin yanındaki sayı değeri ile işlem sırasında program tarafından sayılan nokta adedi yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez.

**5.adım: DATA ascii / DATA binary**

Dosyanın bu formatında iki farklı tip vardır.

Bu tiplerden biri ascii diğeri ise binary tiptir.

Bu adımda okunan data tipi dosyanın hangi tipte yazıldığını belirler. Eğer ki bu yapılardan herhangi biri yanlış ise dosya hatalıdır bu dosya için işleme devam edilmez.

Bu adımlardan sonra alınan veriler ile uygun kontroller sağlandıktan sonra kullanıcı menüde belirtilen işlemleri seçerek koordinat düzlemi üzerinde belirli işlemler yaptırılmasını sağlar.

**Bu işlemler:**

**1**. Dosyalardaki hataları bulup bastırılması hatasız dosyalarda da “Tüm dosyalar uyumludur”

yazdırılması.

**2**. iki nokta arası en yakın ve en uzak noktaların bulunması

**3**.Tüm noktaları kapsayan en küçük çizilebilecek küpün koordinat düzlemi üzerindeki köşe bilgilerini bulmaktır.

**4**.Kullanıcıdan alınan küre bilgileri ile çizilen kürenin içinde kalan noktaların tespiti

**5**.iki nokta arası bulunan uzaklıkların ortalaması

Yapılan işlemlerin sonuçları output.nkt dosyasına bastırılır. İşlem menüsünden çıkış yapmak için klavyeden 6 tuşuna basılır ve program sonlandırılır.

*B) Denklemler*

**1. İki Nokta Arası Uzaklık Denklemi:**

Uzaklık=((x2-x1)^2+(y2-y1)^2+(z2-z1)^2)^1/2

**2. Küre Denklemi:**

Kullanıcıdan alınan değerler:

cx = kullanıcıdan alınan kürenin x değeri

cy = kullanıcıdan alınan kürenin y değeri

cz = kullanıcıdan alınan kürenin z değeri

cr = kullanıcıdan alınan kürenin yarıçapı

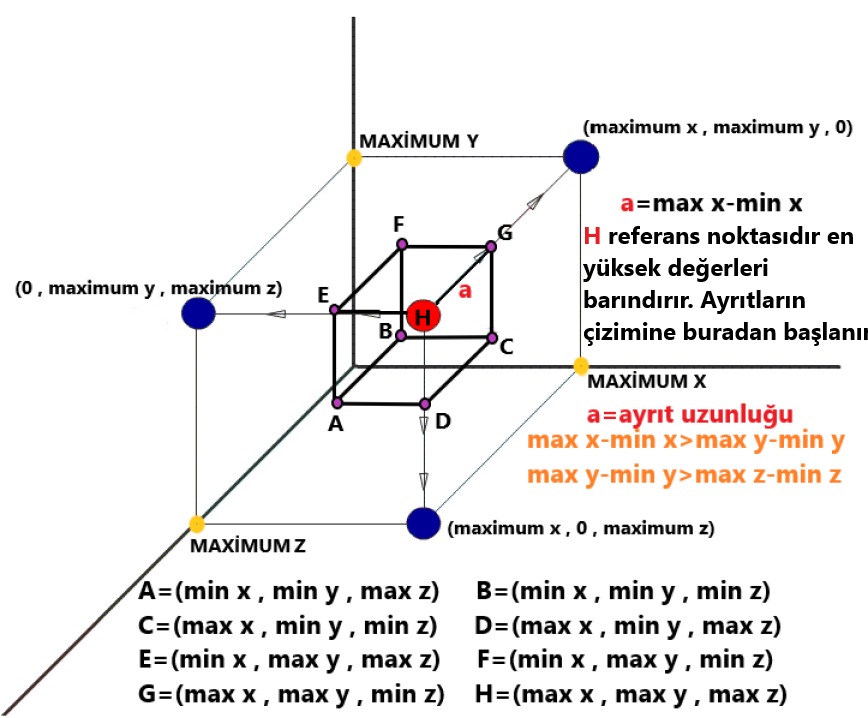
Uzaklık=((x-cx)^2+(y-cy)^2+(z-cz)^2)^1/2

(x, y, z değerleri dosyadan çekilen nokta bilgileri)

Eğer ki:

Uzaklık <= cr ( Nokta Küpün İçindedir )

**3. Küp Denklemi:**

****

C) *Kullanılan Fonksiyonlar*

**Void dosyabul();**

.nkt dosyalarının klasörden okunup tespit edildiği fonksiyondur

**Void dosyaoku();**

Diğer fonksiyonların belirli bir sırada çağırılıp işleme alınmasını sağlar

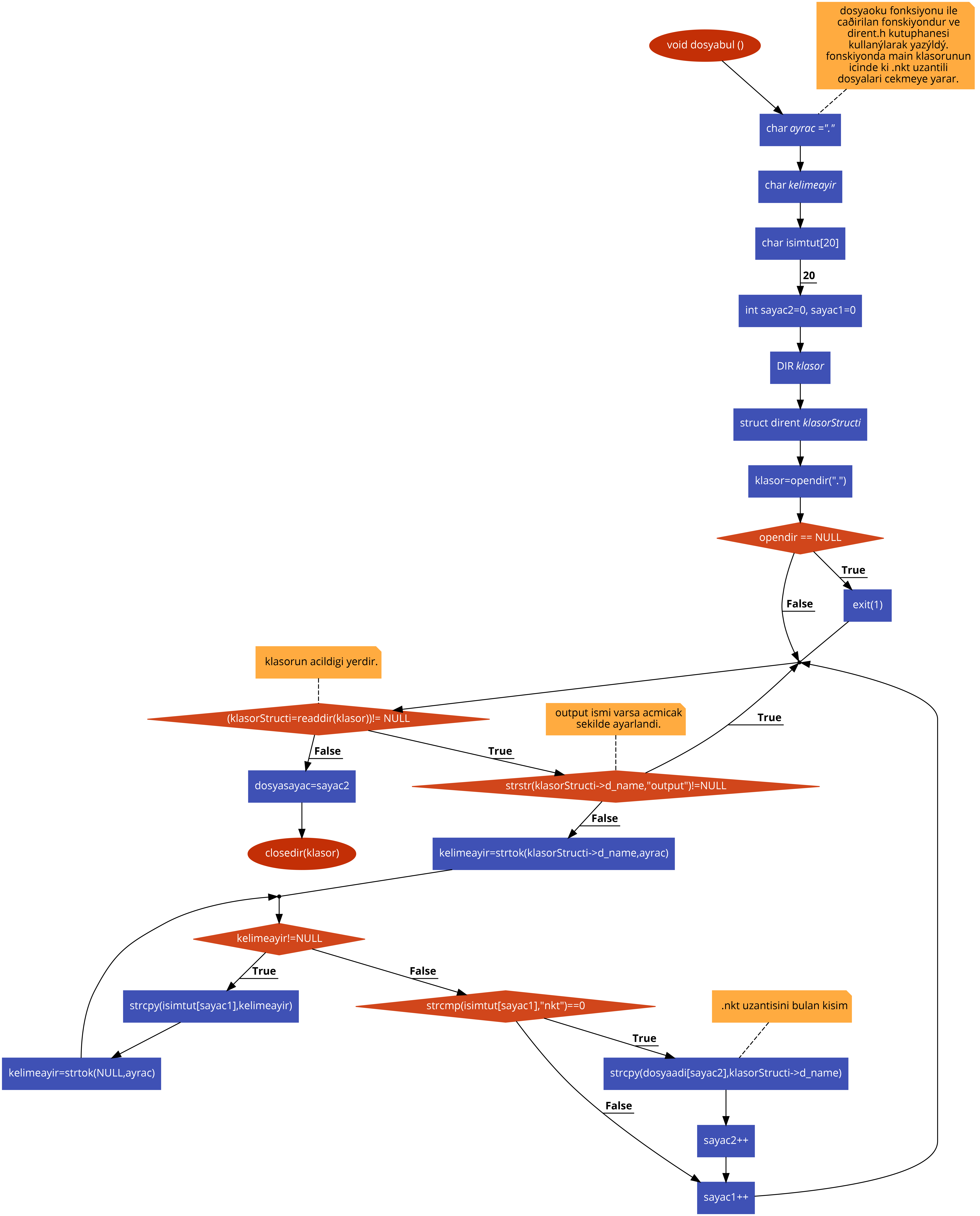
**Void ascii\_binary();**

Ascii ve binary tipindeki dosyaların okunduğu hatalarının bulunduğu ve verilerin belleğe çekildiği yerdir.

**Void menuisleri();**

Menünün tasarlanıp işleyişinin gerçekleştiği yerdir.5 tane case kullanılmıştır. Her bir işlem için ascii ve binary çıktılar uygun bir şekilde verilmiştir

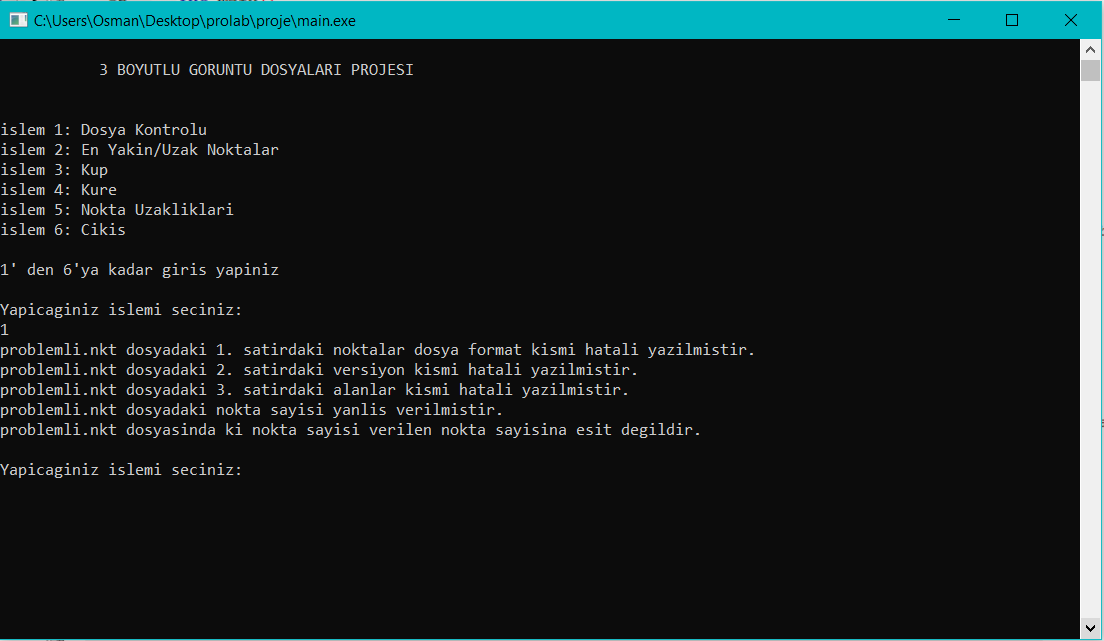
*D) Akış Diyagramı*



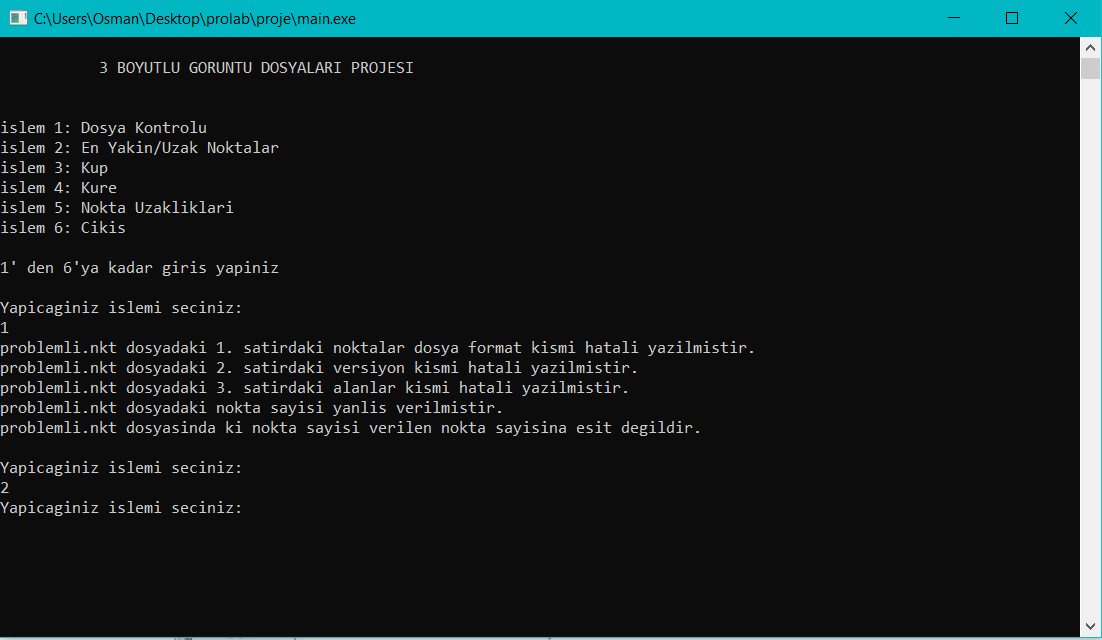
*( .nkt uzantılı dosyaları çekme algoritması )*

*E) Sonuçlar*

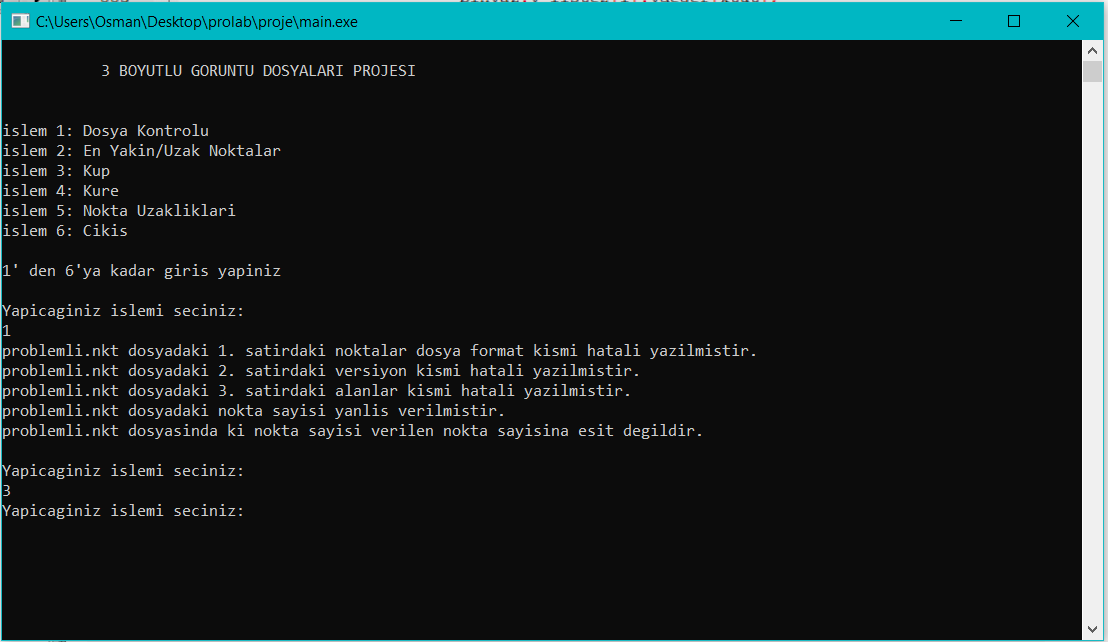
Program çalıştırıldığında dosya kontörlünü sağlayıp, sağlam dosyalar için 5 farklı işlemi gerçekleştirebilmektedir. Her bir dosya ve işlem için Output verilmektedir. Proje, işlemleri doğru şekilde yapıp ve doğru outputlar verdiği için proje hedefe yeteri kadar ulaşmıştır.

*F) Ekran çıktıları*

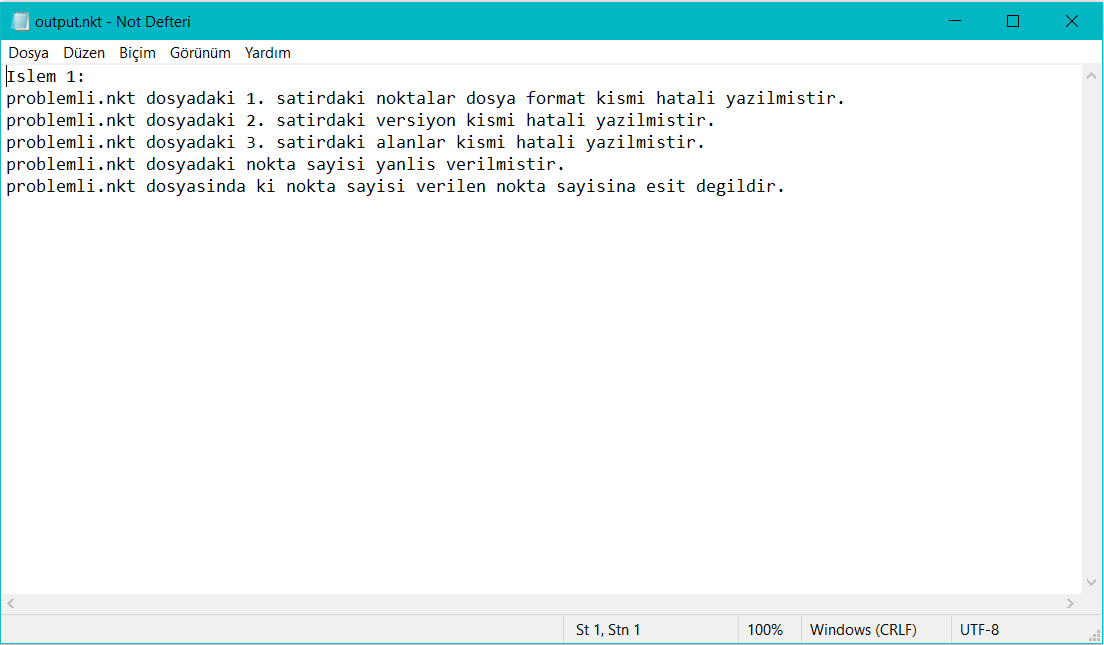
*İşlem 1’in ekran çıktısı*



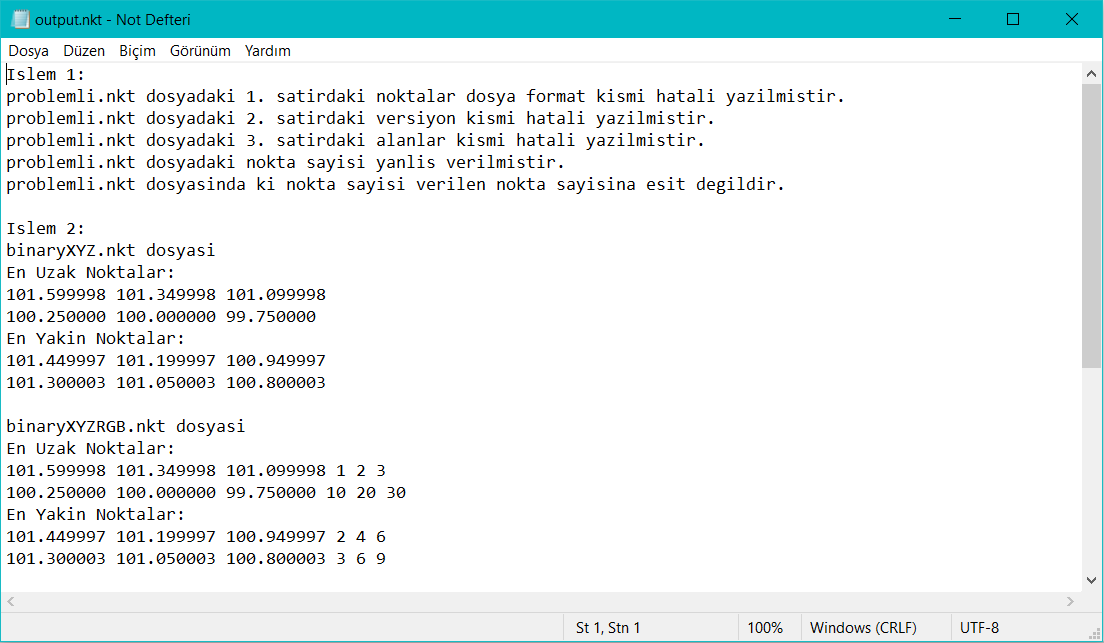
*İşlem 2’nin ekran çıktısı*



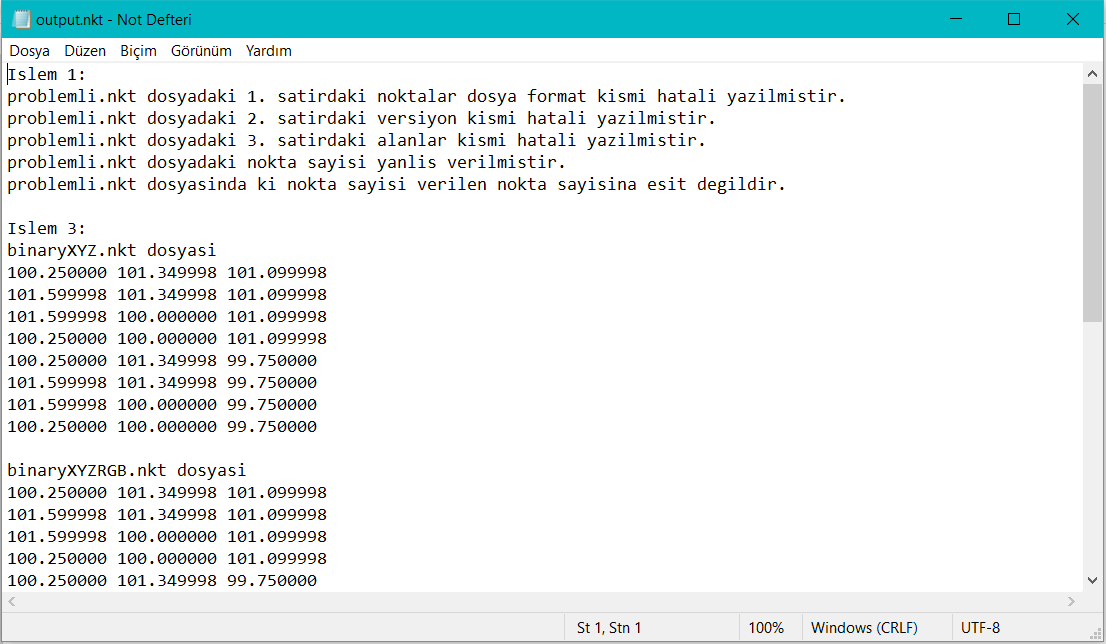
*İşlem 3’nin ekran çıktısı*



*İşlem 1’in output.nkt ekran çıktısı*



*İşlem 2’nin output.nkt ekran çıktısı*



*İşlem 3’in output.nkt ekran çıktısı*

*G) Karşılaşılan Hatalar ve Çözüm Yaklaşımları*

Binary dosyalarında x y z r g b formatında okuma yaparken zorlandık çünkü x y z değerleri float r g b değerleri int değer olduğu için int değerleri cmd ekranına bastıramıyorduk fakat 1. Sınıf ders slaytlarında dosya slaydından yararlanıp yaptık.

*H) Kazanımlar*

*1)* dirent.h kütüphanesi ile klasör içindeki dosya adlarına erişim yapabildik.

*2)* dosya okuması yaparken struct dizilerinin kullanımlarını gördük.

*3)* fread() fonksiyonu ile binary dosyalarından sayısal veri çekmeyi ve bunu structlar yardımıyla yapmayı gördük*.*

*4)* struct yapıları ile pointer kullanmayı her bir pointera bellekte yer açıp bellekte ki boyutunu değiştirmeyi gördük.

*K) Kaynakça*

*1)* dirent.h

<https://pubs.opengroup.org/onlinepubs/007908799/xsh/dirent.h.html>

<https://c-for-dummies.com/blog/?p=3246>

*2)* dosya işlemleri ( ascii, binary )

[*https://www.bilgigunlugum.net/prog/cprog/c\_dosya*](https://www.bilgigunlugum.net/prog/cprog/c_dosya)

[*https://drive.google.com/open?id=1o0eQ8ufppL1jhByWKZrp-YXzV\_mVPWGZ*](https://drive.google.com/open?id=1o0eQ8ufppL1jhByWKZrp-YXzV_mVPWGZ)

[*http://bilgisayarkavramlari.sadievrenseker.com/2008/10/22/c-ile-dosya-islemleri/*](http://bilgisayarkavramlari.sadievrenseker.com/2008/10/22/c-ile-dosya-islemleri/)

*<https://stackoverflow.com/questions/17598572/read-write-to-binary-files-in-c>*

*3) string işlemleri*

[*https://drive.google.com/open?id=1rS0BlbQWn7XOmCMbTkUBWtB5cxlRHQtk*](https://drive.google.com/open?id=1rS0BlbQWn7XOmCMbTkUBWtB5cxlRHQtk)

[*http://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/c/string.htm*](http://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/c/string.htm)

[*https://stackoverflow.com/questions/7228438/convert-double-float-to-string*](https://stackoverflow.com/questions/7228438/convert-double-float-to-string)

*4) bellek işlemleri*

[*https://drive.google.com/open?id=1GKTD6-zhsAbAC1Lg2-V3Nbl-GDbWKsdw*](https://drive.google.com/open?id=1GKTD6-zhsAbAC1Lg2-V3Nbl-GDbWKsdw)

[*https://www.bilgigunlugum.net/prog/cprog/c\_bellek*](https://www.bilgigunlugum.net/prog/cprog/c_bellek)