

# 3 BOYUTLU NOKTA BİLGİLERİNİN OKUTULMASI VE İŞLENMESİ

Taha İsmet Başar  
Bilgisayar Mühendisliği Bölümü  
Kocaeli Üniversitesi  
basartaha@gmail.com

Şevki Karagöl  
Bilgisayar Mühendisliği Bölümü  
Kocaeli Üniversitesi  
sevkikaragol@outlook.com

**Özet-**Klasör içerisine yerleştirilmiş olan farklı türdeki 3 boyutlu görüntü dosyalarından(nokta bilgisi)okutma işlemi gerçekleştirilmesi istenmektedir. Bu noktalar xyz koordinat düzlem bilgileri ve rgb renk kodlarını temsil etmektedir.Projenin ilk ayağında dosya kontrol işlemi gerçekleştirilir eksik ya da hatalı hazırlanmış dosyalar süzülür ve uyarı bilgisi sunulur.Gerek yeter koşulu sağlayan dosyalarla işleme devam edilir ve sırasıyla, dosyaların içerisinde tutulan nokta bilgilerinden birbirlerine en yakın ve en uzak nokta bilgileri hesaplanması ve nokta numaları saptanması,tüm noktaları içine alacak bir küpün kenar nokta bilgileri merkezi 3 boyutlu koordinati ve yarıçap bilgileri kullanıcıdan istenilen bir kürenin içinde kalan noktaların bilgileri ,her bir noktanın birbirine olan uzaklıklarının ortalaması hesaplanmaktadır. Takibi yapılan işlemler klasör içerisinde bir dosya altında saklanır.

**Anahtar kelimeler-**  
**Nokta,dosya,hesaplama,görüntü,düzlem ve üç boyut**

## I.GİRİŞ

Proje klasörün içinde bulunan tüm dosyalardan ayrı ayrı nokta okunur ve bellekte tutularak proje kapsamında istenen işlemleri gerçekleştirir. İşle-

nen ve hesaplanan veriler dosya üzerine “.nkt” uzantısı ile kaydedilir. Dosya işlemleri ve lineer cebirin bir arada kullanımına yönelik bir çalışma gerçekleştirilmektedir.Aynı zamanda öğrencinin matematiksel işlemleri ide aracılığıyla bilgisayar ortamına aktarması amaçlanmıştır. Proje için tercih edilen programlama dili “C” olmuştur.

## II.TEMEL BİLGİLER

Bu proje C programlama dilinde geliştirilmiş olup, geliştirme ortamı olarak “Code Blocks ver. 17.12” kullanılmıştır.İlk etapta proje için bir yol haritası çıkarılarak ön hazırlık sürecine girilmiştir. Bu aşamada projenin isterlerine yönelik araştırmalar gerçekleştirmesi adına grup içerisinde bir iş bölümü yapılmış olup elde edilen veriler doğrultusunda projenin ana hatları ortaya çıkarılmış ve büyük ölçüde karışılabilen problemler saptanıp çözümlendirildikten sonra derleyici ortamında projenin ilk adımları atılmıştır.Proje yaklaşık 1.5 hafta gibi süre içerisinde tamamlanmıştır.

### III.YÖNTEM

Main fonksiyon aracılığıyla öncelikle klasörün içerisinde bulunan txt dosyalarına erişim sağlandı. Bunun için ise dirent.h kütüphanesi kullanıldı ve gerek yeter koşulu sağlayan dosyalar üzerinde işlemler yapıldı. Bu işlemler aracılığıyla birbirlerine en yakın en uzak noktalar, noktaları içine alan en küçük küp ve kullanıcıdan alınan veriler aracılığıyla kürenin içinde kalan noktalar tespit edildi. Aynı zamanda programa veri girişi istenerek kullanıcılara bu projede aktif bir rol verildi. Ve bu işlemler yapılırken binary ve ascii verilerden yararlanıldı.

#### ***Projede kullanılan fonksiyonlar:***

boslukkontrol() fonksiyonu aracılığıyla klasör içerisine eklenen dosyaların nokta bilgilerinin kontrolü sağlandı.

void kure() fonksiyonu ile merkezinin koordinatı ve yarıçapı bilgileri kullanıcıdan alınmış olan noktalar küre içerisine alındı.

void kup() fonksiyonu ile dosyadan okunan noktalar ile kupun kenar noktaları hesaplandı.

rgb() fonksiyonunda ,okuduğumuz hatasız dosya içerisindeki satılarda r g b nokta bilgilerinin kontrolü yapıldı ve r g b nokta bilgileri eksik olan noktanın yeri saptandı.

void rgb\_baski()fonksiyonunda r g b bilgisi eksik olan satır kullanıcıya bildirildi.

void ister5() fonksiyonunda noktaların birbirlerine olan uzaklıklarının ortalaması elde edildi ve kullanıcıya bildirildi.

void asciiokuma() fonksiyonunda .nkt uzantılı olup ascii içeriğe sahip olan noktalar proje başlangıcında tanımlanan struct verisi aracılığıyla bellekte tutuldu.

void binaryokuma() fonksiyonunda .nkt uzantılı olup binary içeriğe sahip olan noktalar proje başlangıcında tanımlanan struct verisi aracılığıyla bellekte tutuldu.

***void kup\_binary, void kure\_binary, void ister5\_binary*** Bu fonksiyonlar aracılığıyla ascii veriler için yapılan hesaplamalar binary veriler için de yapıldı ve kullanıcıya bilgi vermek aracılığıyla bir struct verisinde tutuldu.

### IV.SÖZDE(PSEUDO) KOD

***Verilen proje için hazırladığımız programın pseudo kodları şu şekildedir:***

1-Program çalıştı.

2-Program, içerisinde yer aldığı dosya yolunda bulunan verileri okudu ve .nkt uzantılı dosyaları ayıkladı.

3-Kullanıcıya,menü aracılığıyla gerçekleştirmek istediği işlem soruldu.

***\*ek 1***

4-Kullanıcı 1 tuşuna bastı.

5-Program,okuma yaparak kullanıcıya klasör içerisinde bulunan ve .nkt uzantısına sahip olan dosyalar arasından ilk olarak başlık,x y z ve r g b bilgileri hatasız olanları bildirdi.Ardından başlık bilgileri hatasız fakat r g b verilerinde eksiklik bulunan noktalar kullanıcıya bildirildi.

***\*ek 2***

7-Program, dosyaları okuyarak projede kullanılması amaçlanan hatasız verileri dosyalarının isimlerini bir dizi aracılığıyla bellekte tuttu.

8-Menü tekrar görüntülendi ve kullanıcıya yapmak istediği işlem soruldu.

9-Kullanıcı 2 tuşuna bastı.

10-Program, ayıkladığı sağlıklı dosyalardan okuduğu tüm noktaları bir döngü aracılığıyla birbirleriyle kıyasladı ve aralarındaki uzaklıkları buldu ve bunların arasından en ırak ve en yakın noktaların yerlerini bir değışkende tuttu. Kullanıcıya, hesaplanmış uzaklık değeri hakkında bilgi verdi.

*\*ek 3*

11-Menü tekrar görüntülendi ve kullanıcıya yapmak istediği işlem soruldu.

12-Kullanıcı 3 tuşuna bastı.

13-Program, belleğine kaydettiği noktaların tümünü içine alan küpün, alt ve üst kenarlarında bulunan köşelerin nokta bilgilerini kullanıcıya aktardı.(Bu köşe bilgileri x y z noktalarının en büyük, en küçük ve fark değeri bulunması ile oluşturuldu.)

*\*ek 4*

14-Menü tekrar görüntülendi ve kullanıcıya yapmak istediği işlem soruldu.

15-Kullanıcı 4 tuşuna bastı.

16-Program ,küre oluşturmak için kullanıcıdan kürenin x,y,z ve r bilgilerini istedi.

17-Kullanıcı x,y,z ve r bilgilerini girdi.

18-Program,girilen verileri kullanarak bir küre

oluşturdu ve bu kürenin içinde kalan noktaları kullanıcıya bildirdi.(Bu veriler girilen merkez ile okunan noktalar arasındaki uzaklıkların hesaplanması ve yarıçapla kıyaslanması ardından bulundu.)

*\*ek 5*

*\*ek 5.1*

19-Menü tekrar görüntülendi ve kullanıcıya yapmak istediği işlem soruldu.

20-Kullanıcı 5 tuşuna bastı.

21-Program kullanıcıya daha önce de hesapladığı,noktaların birbirlerine olan uzaklıklarının ortalama değeri bildirdi.(Ortalama bilgileri karşılaştırma sayısına bölünerek ortalama bulundu.)

*\*ek 6*

22-Program gerçekleştirilen bu işlemler sırasında kullanıcıya bildirdiği verileri output dosyalarına yazdırdı.

23-Menü tekrar görüntülendi ve kullanıcıya yapmak istediği işlem soruldu.

24-Kullanıcı -1 tuşuna bastı ve menüden çıkış işlemi gerçekleşti.

25-Program sonlandırıldı.

## V.KULLANILAN KUTUPHANELER VE EKLER

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>→Dizi işlemleri için
kullanıldı.
```

#include <dirent.h>→Hatasız dosyaların ayıklanması için kullanıldı.  
#include <math.h>→Matematiksel işlemler için kullanıldı.  
#include <ctype.h>→isdigit işlemi için kullanıldı.

## Ekler

\*ek1

```
ISLEMLER

1)Dosya Kontrolu
2)En Yakın/Uzak Noktalar
3)Kup
4)Kure
5)Nokta Uzakliklari Ortalama
Programdan cikis yapmak icin -1'e basiniz.
Yapmak istediginiz islem nedir?:
```

\*ek2

```
Yapmak istediginiz islem nedir?:1
Hatasiz dosyalar:
binaryXYZRGB.nkt
bos.nkt dosyasinin 25.satirinda Rgb bilgileri eksiktir.
veriler1.nkt
```

\*ek3

```
Yapmak istediginiz islem nedir?:2

1.nokta 101.449997 101.199997 100.949997 2 4 6
2.nokta 101.300003 101.050003 100.800003 3 6 9
0.nokta 101.599998 101.349998 101.099998 1 2 3
8.nokta 100.400002 100.150002 99.900002 9 18 27

4558.nokta 97.079498 33.694393 22.512339 76 44 81
4599.nokta 97.035797 33.674644 22.449869 182 107 105
754.nokta 4.981094 0.087787 99.749023 81 178 204
7087.nokta 94.957733 98.022102 0.558709 10 192 126
```

\*ek4

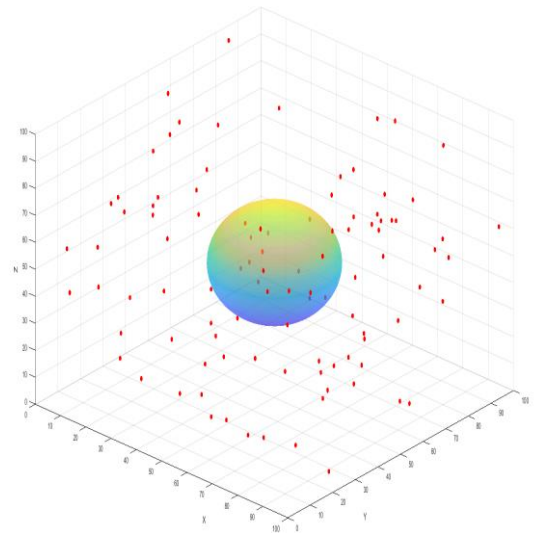
```
Alt taban
99.999300 0.003270 0.012146
0.015070 0.003270 0.012146
0.015070 99.987500 0.012146
99.999298 99.987503 0.012146
Ust taban
99.999298 0.003270 99.996376
0.015070 0.003270 99.996376
0.015070 99.987503 99.996376
99.999298 99.987503 99.996376
```

\*ek5

```
Kullanıcıdan alınan x değeri:0.000000
Kullanıcıdan alınan x değeri:0.000000
Kullanıcıdan alınan x değeri:0.000000
Kurenin yarıcap değerini giriniz:8

Kullanıcıdan alınan r değeri:8.000000
Kurenin icinde kalan noktalar:
4.858072 3.756398 2.055958 219 156 241
3.010062 4.686761 1.248630 204 182 66
0.601166 5.537703 4.908463 218 171 231
```

\*ek5.1



*\*ek 6*

```
Yapmak istediginiz islem nedir?:5  
Binary uzaklik ortalamasi=0.866024  
Ascii ortalama=66.097465
```

## VI.KAYNAKÇA

- [1][stackoverflow.com/c-programming-simple-steps.com](https://stackoverflow.com/c-programming-simple-steps.com)
- [2] [hexed.it](https://hexed.it)
- [3][pubtekno.com](https://pubtekno.com)
- [4][bilgigunlugum.net](https://bilgigunlugum.net)
- [5]Youtube-Bilgisayar Kavramları
- [6]Youtube-Yazılım Bilimi
- [7][embedded.kocaeli.edu.tr](https://embedded.kocaeli.edu.tr)
- [8][edestek.kocaeli.edu.tr](https://edestek.kocaeli.edu.tr)