ÜÇ BOYUTLU GÖRÜNTÜ DOSYALARI PROJESİ

Mehmet Ali KILIÇLI – Mehmet ÖZDEMİR

Kocaeli Üniversitesi

[kiliclimehmetali@gmail.com](mailto:kiliclimehmetali@gmail.com) - [mehmetoziron@gmail.com](mailto:mehmetoziron@gmail.com)

**ÖZET**

Koordinat ve renk bilgileri verilen 3 boyutlu nokta verilerinin dosyadan okuma yapılarak bazı işlemler yapılması.

**1. GİRİŞ**

Projede binary ve ascii tipindeki nokta ve bu noktaların renk bilgilerini başlık kontrolü(nokta bilgilerinin sadece koordinat olarak mı yoksa koordinat bilgisi yanında renk bilgilerinin de verildiği mi, dosyadaki verilen nokta sayısının doğruluğu, dosyanın bilgilerinin ascii tipinde mi yoksa binary tipinde mi olduğu) yaptıktan sonra bazı bilgileri kullanıcı tarafından istenen işlemlere tabi tutulur.Bunlar birbirine en yakın ve en uzak nokta bilgileri, tüm noktaları içine alacak en küçük küpün köşe nokta bilgileri, merkezinin koordinatları ve yarıçap uzunluğunun kullanıcıdan alınan küre içindeki nokta bilgileri.

**2. Temel Bilgiler**

Proje gelişiminde;

Geliştirme ortamı olarak “CodeBlocks” kullanılmıştır.

Projede istenen işlemler lineer cebir ve matematik bilgileri göz önüne alınarak hesaplanmıştır.

**3. Tasarım**

Proje aşağıdaki başlıklar altında geliştirilmiştir.

**3.1 Yazılım Tasarımı**

Projenin yazılım aşaması bu başlık altında bulunan konular tarafınca geliştirilmiştir.

Projede 0-1-2-3-4-5 (0 tuşu programı bitirir) tuşları haricinde yapılan herhangi bir tuşlamada program tarafından uyarı alınarak tekrar girdi alınır. Dosya kontrolü yapılmadan işlemler yapılmaması gerektiği için ilk girdi olarak 1 tuşuna basılması istenmektedir. Aksi taktirde uyarı verir ve tekrar girdi istenir.

**3.2 Algoritma**

**18-27** - Klosördeki dosyaların isimlerini al.

**27-39** - “.nkt” uzantılı ise kaydet.

**41-42** - Seçilen dosyaları ekrana yaz.

**45** - While döngüsü

**47** - Seçim ara yüzünü ekrana yaz

**48-60** - Seçim 1’den farklıysa tekrar girdi iste.

**61-67** – Seçim 0’sa programı kapat, seçim 1’den küçük 5’ten büyükse tekrar girdi iste.

**70-75** – Sıradaki dosyayı okuma modunda aç.

**78-82** – Dosyanın satır sayısını hesapla ve satır sayısına göre iki boyutlu pointer dizisi oluştur.

**83-90** - 0'dan satır sayısına kadar her satırın uzunluğunu artır.

**91-99** - Dosyadaki satırları teker teker satirlar[] dizisine ata.

**106** - Satır sayısını ekrana yaz.

**110** - Seçim 1'se geçerli dosyayı ekleme modunda aç.

**112-114** - Seçim 1 dosya kontrolünü output dosyasına yaz.

**115-126** - Eğer dosyanın üçüncü satırı "Alanlar x y z" ise 6. satırdan itibaren Boşluk\_Say\_xyz fonksiyonuna satırları teker teker gönder.

Fonkiyon 0 değerini göndermişse o satırın hatalı olduğunu yaz. Hata değerini artır.

**127-139** - Değilse eğer dosyanın üçüncü satırı "Alanlar x y z r g b" ise 6. satırdan itibaren Boşluk\_Say\_xyzrgb fonksiyonuna satırları teker teker gönder.

Fonkiyon 0 değerini göndermişse o satırın hatalı olduğunu yaz. Hata değerini artır.

**140-146** - Değilse alan bilgileri hatalıdır yazdır. Hata değerini artır.

**148-158** - Eğer 4. satırdaki nokta sayı bilgisi nokta sayısıyla aynı değilse hata raporu ver. Hata değerini artır. Değilse onay raporu ver.

**160-172** - Eğer 5. satır bilgisi ascii ise "Dosya acsii tipindedir" yaz. Değilse eğer 5. satır bilgisi binary ise "Dosya binary tipindedir" yaz. Değilse hata raporu ver. Hata değerini artır.

**173-184** - Eğer hata değeri 0'dan farklıysa, output dosyasına ve ekrana "Hatalı dosya devre dışı bırakıldı" yaz, bu dosyayı devre dışı bırak, sıradaki dosyaya geç.

**186-189** - Output dosyasına "Geçerli dosya uyumludur" yaz. Sıradaki dosyaya geç.

(70.Satırdaki for döngüsüne dön)

**193-200** - Eğer dosya "x y z" tipindeyse 6. satırdan itibaren sayiya\_cevir fonksiyonuna gönder.

**200-208** - Eğer dosya "x y z r g b" tipindeyse 6. satırdan itibaren sayiya\_cevir\_renkli fonksiyonuna gönder.

**210-220** - Eğer seçim 4 ise kullanıcıdan küre merkez koordinatlarını ve küre yarıçap bilgisini al. Output dosyasına seçim4'ü, açılan dosyayı, kullanıcıdan alınan küre merkez koordinatı ve küre yarıçapını yaz.

**223** - Küre merkez, yarıçap bilgilerini ve her nokta bilgisini kure\_hesabi fonksiyonuna gönder.

**224-243** - kure\_hesabi fonksiyonunun gönderdiği değer -1'den büyükse, eğer dosyanın renk bilgileri yoksa output dosyasına nokta koordinatlarını yaz. Değilse output dosyasına nokta koordinatlarını ve renk bilgilerini yaz. Sıradaki dosyaya geç.

(70.Satırdaki for döngüsüne dön)

**244-274** - Eğer seçim 2 veya 5 ise, enbenk fonksiyonuna tüm nokta koordinatlarını ayrı ayrı gönder. Fonksiyonun gönderdiği iki nokta arası uzaklıkları topla. En yakın ve en uzak noktaları bul.

**275-287** - Eğer seçim 5 ise toplanan iki nokta arası uzaklıkları işlem sayısına böl. Output dosyasına noktaların uzaklık ortalamasını yaz. Sıradaki dosyaya geç.

(70.Satırdaki for döngüsüne dön)

**288-310** - Eğer seçim 2 ise, eğer noktalar renk bilgileri olmadan verilmişse output dosyasına en yakın ve en uzak nokta koordinatlarını yaz. Değilse output dosyasına en yakın ve en uzak nokta koordinatlarını ve renk bilgilerini yaz. Sıradaki dosyaya geç.

(70.Satırdaki for döngüsüne dön)

**312-368** - Eğer seçim 3 ise en küçük x ve en büyük x'in farkını, en küçük y ve en büyük y'nin farkını, en küçük z ve en büyük z'nin farkını bul ve en büyüğünü küpün kenar uzunluğu yap. En küçük x,y ve z nin kesişim koordinatlarını küpün köşesi kabul et ve bulunun uzaklığını da kullanarak küpü oluştur. Küpün köşe koordinatlarını output dosyasına yaz. Sıradaki dosyaya geç

(70.Satırdaki for döngüsüne dön)

**3.3 Fonksiyonlar**

enbenk : Alınan noktaların birbirine olan uzaklıklarını hesaplar ve ana fonksiyona gönderir.

kure\_hesabi : Herhangi bir noktanın, merkez koordinatları ve yarıçap uzunluğu kullanıcıdan alınan kürenin içinde kalıp kalmadığını kontrol eder.

sayiya\_cevir : Gönderilen satırın koordinat bilgilerini x,y ve z düzlemlerinde ayrı ayrı işlem yapabilmek için ayırır.

sayiya\_cevir\_renkli : Dosyadan alınan koordinat bilgilerini x,y ve z düzlemlerinde ayrı ayrı işlem yapabilmek için ayırır. “sayiya\_cevir” fonksiyonundan farkı koordinat bilgilerinin yanında renk kodlarına da sahip olmasıdır.

Bosluk\_Say\_xyz : Gönderilen satırın koordinat bilgilerinin yanlış olup olmadığını satırdaki boşluk sayısı yardımıyla bulur.

Bosluk\_Say\_xyzrgb : Gönderilen satırın koordinat ve renk bilgilerinin yanlış olup olmadığını satırdaki boşluk sayısı yardımıyla bulur.

**3.4 Kullanılan Kütüphaneler**

#include <stdio.h>

#include <stdlib.h>

#include <dirent.h>

#include <math.h>

**3.5 Arayüz**

“.nkt” uzantılı dosyaları açma ve okuma.<dirent.h> kütüphanesi kullanarak çözdük.

Programın ara yüzü 1,2,3,4,5 işlem için tuşlar ve 0 programı kapatmak için tasarlanmıştır.

**4. Karşılaşılan Sorunlar Ve Çözüm Yöntemleri**

“.nkt” uzantılı dosyaları açma ve okuma.<dirent.h> kütüphanesi kullanarak çözdük.

Verilen örnek dosyalardaki nokta bilgilerinin fazla olmasından kaynaklı bellekten yer alımının yeterli gelmemesi. Çözümü “malloc” fonksiyonuyla nokta satısı kadar bellekten yer almak.

Tüm noktaları içine alacak en küçük küpün köşe nokta bilgilerini bulmak. Çözümü Lineer cebir ve matematik ders notlarını tekrar gözden geçirdik.

**5. Kazanımlar**

Herhangi bir uzantısı bulunan dosya açma.

Veri sayısı çok fazla olan dosya okuma işlemlerinde bellekten daha fazla yer edinme.

Lineer cebir ve matematik konularında bilgileri pekiştirmek.

**6. Kaynakça**

Kocaeli Üniversitesi – Bilgisayar Mühendisliği 2018-2019 Programlama2 (C Programlama) Ders Notları (Onur Gök ve Suhap Şahin)

Web-Youtube-Navad Ba-“part 28 - DIR , opendir , readdir , struct dirent”

7. **Ekran ve dosya çıktıları ekran görüntüleri**



