**3 BOYUTLU GÖRÜNTÜ DOSYALARI PROJESİ**

Arda Talu

Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü

170202037 [arda.starkiler@gmail.com](mailto:arda.starkiler@gmail.com)

**Özet**

Projede farklı türdeki 3 boyutlu görüntü dosyalarından okuma yapılarak verilecek problemlerin çözümü istenmektedir. Amaç: Proje gerçekleştirimi ile beraber öğrencilerin dosyalama ve lineer cebir konularında program yazabilmesi amaçlamaktadır.

**1.Giriş**

Programlama Dili: Proje C dili kullanılarak gerçekleştirilmiştir.

Kullanılan Araçlar: Proje Code::Blocks 17.12 sürümü ve Notepad++ araçlarından faydalanılarak yapılmıştır.

**2.Algoritma**

Program başlatıldığında kullanıcıdan seçim yapması istenir.(1-2-3-4-5)

Yaptığı seçime göre program işlem yapar.

Seçim 1 olursa program, Dosya Kontrolü yapar ve dosyaların hatalarını gösterir. Dosya hatalıysa yaptığı hatayı yazar. Hata yoksa “Tüm Dosyalar Uyumludur.” çıktısı verir.

Seçim 2 olursa Dosyalardaki nokta verileri okunur ve her dosyadaki birbirine en yakın ve en uzak noktalar ekrana basılır.

Seçim 3 olursa Dosyaların içindeki noktalar incelenip hepsini içine alabilecek bir küp için kenar nokta bilgileri ekrana basılır.

Seçim 4 olursa kullanıcıdan bir noktanın x,y ve z koordinat bilgileri ve bir yarıçap bilgisi istenerek bir küre oluşturulur. Dosyadaki noktaların kürenin merkezi(alınan koordinatlar) ile arasındaki mesafe ile yarıçap kıyaslanarak kürenin içinde olup olmadığı kontrol edilir. İçinde olan noktalar ekrana basılır.

Seçim 5 olursa Dosyadaki noktaların birbirine olan uzaklıkları hesaplanıp aralarındaki uzaklıkların ortalaması ekrana basılır.

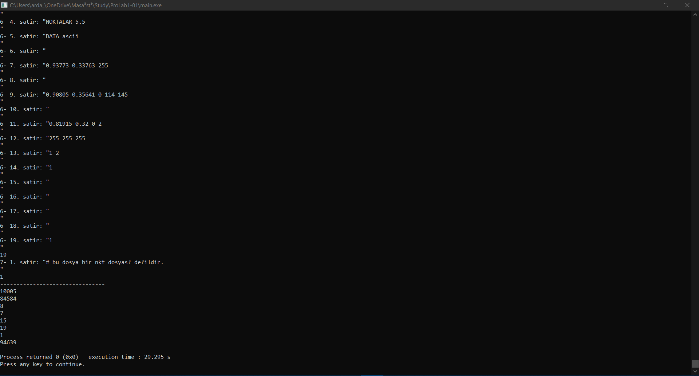
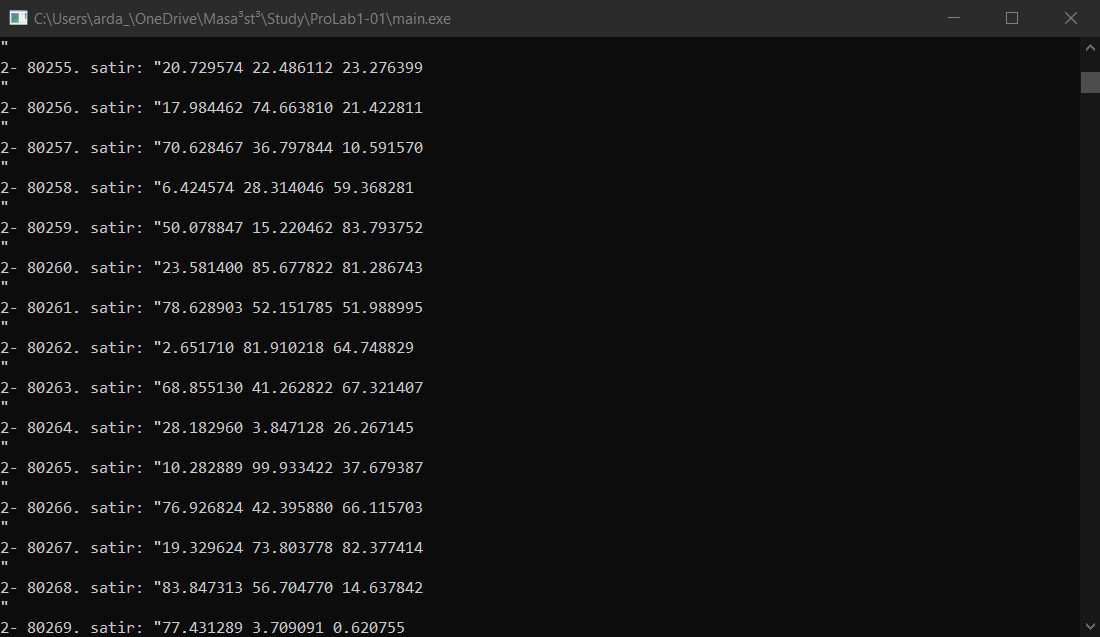
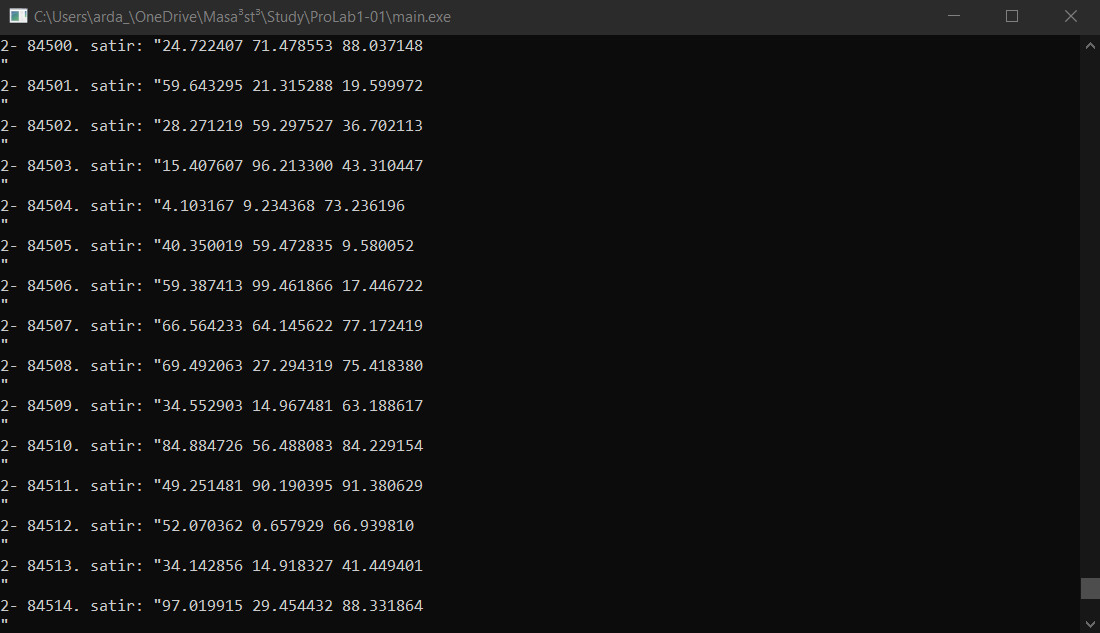
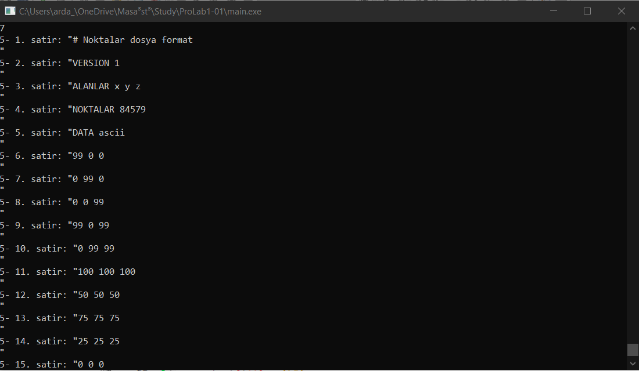
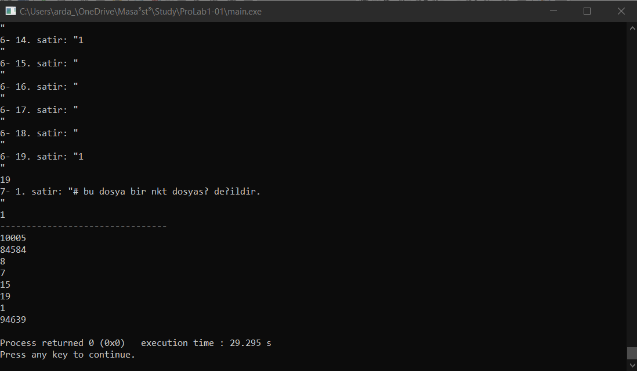
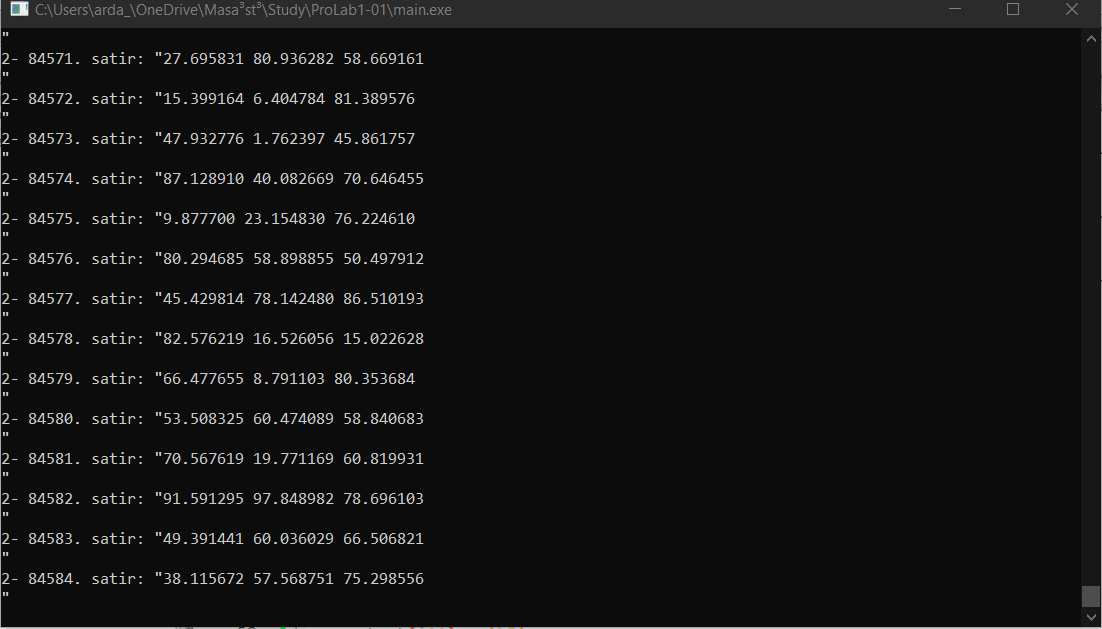
**3.Sözde Kod**

1. Başla
2. stdio.h kütüphanesini ekle
3. stdlib.h kütüphanesini ekle
4. string.h kütüphanesini ekle
5. math.h kütüphanesini ekle
6. "veriler1.nkt" dosyasını fp1 olarak tanımla
7. "veriler2.nkt" dosyasını fp2 olarak tanımla
8. "binaryXYZRGB.nkt" dosyasını fp3 olarak tanımla
9. "binaryXYZ.nkt" dosyasını fp4 olarak tanımla
10. "testVerileri.nkt" dosyasını fp5 olarak tanımla
11. "problemli.nkt" dosyasını fp6 olarak tanımla
12. "nkt\_olmayan\_dosya.benioku" dosyasını fp7 olarak tanımla
13. Struct Nokta1 n1
14. float x[10]
15. float y[10]
16. float z[10]
17. int r[10]
18. int g[10]
19. int b[10]
20. Struct Nokta1X n1X
21. char x[10]
22. char y[10]
23. char z[10]
24. int r[10]
25. int g[10]
26. int b[10]
27. Struct Nokta2 n2
28. float x[10]
29. float y[10]
30. float z[10]
31. void stringtohexString(char\* input, char\* output)
32. input[loop] eşit değil iken '\0'’a
33. sprintf((char\*)(output+i),"%02X", input[loop])
34. loop+=1
35. i+=2
36. output[i++] = '\0'
37. int main()
38. fp1 olarak tanımlanmış dosya1’i okumak için aç
39. fp2 olarak tanımlanmış dosya2’yi okumak için aç
40. fp3 olarak tanımlanmış dosya3’ü okumak için aç
41. fp4 olarak tanımlanmış dosya4’ü okumak için aç
42. fp5 olarak tanımlanmış dosya5’i okumak için aç
43. fp6 olarak tanımlanmış dosya6’yı okumak için aç
44. fp7 olarak tanımlanmış dosya7’yi okumak için aç
45. YAZ ----MENU----
46. YAZ 1-Dosya Kontrolu
47. YAZ 2-En Yakin/Uzak Noktalar
48. YAZ 3-Kup
49. YAZ 4-Kure
50. YAZ 5-Nokta Uzakliklari
51. YAZ Seciminizi yapiniz:
52. OKU secim
53. Eğer dosya1 NULL ise
54. YAZ Dosya1 acilamadi
55. Eğer dosya2 NULL ise
56. YAZ Dosya2 acilamadi
57. Eğer dosya3 NULL ise
58. YAZ Dosya3 acilamadi
59. Eğer dosya4 NULL ise
60. YAZ Dosya4 acilamadi
61. Eğer dosya5 NULL ise
62. YAZ Dosya5 acilamadi
63. Eğer dosya6 NULL ise
64. YAZ Dosya6 acilamadi
65. Eğer dosya7 NULL ise
66. YAZ Dosya7 acilamadi
67. Switch(secim)
68. CASE 1
69. fgets(buf, 100, dosya1) değil iken NULL
70. s1++
71. s++
72. YAZ ("1- %d. satir: \"%s\"\n", s1, buf)
73. i=6’dan i=s1 olana kadar i’yi arttır
74. fscanf (dosya1,"%f %f %f %d %d %d",&n1.x[i], &n1.y[i], &n1.z[i], &n1.r[i], &n1.g[i], &n1.b[i])
75. YAZ ("%f %f %f\n",&n1.x[i], &n1.y[i], &n1.z[i])
76. YAZ ("%d\n",s1)
77. fgets(buf, 100, dosya2) değil iken NULL
78. s2++
79. s++
80. YAZ ("2- %d. satir: \"%s\"\n", s2, buf)
81. i=6’dan i=s2 olana kadar i’yi arttır
82. fscanf(dosya2,"%f %f %f",&n2.x[i], &n2.y[i], &n2.z[i])
83. YAZ ("%d\n",s2)
84. fgets(buf, 100, dosya3) değil iken NULL
85. s3++
86. s++
87. i=6’dan i=s3 olana kadar i’yi arttır
88. fscanf (dosya1,"%f %f %f %d %d %d",&n1.x[i], &n1.y[i], &n1.z[i], &n1.r[i], &n1.g[i], &n1.b[i])
89. Eğer s3 büyükse 5
90. YAZ ("3- %d. satir: \"%s\"\n", s3, buf)
91. fscanf(dosya3,"%s %s %s %d %d %d", &n1X.x[i], &n1X.y[i], &n1X.z[i], &n1X.r[i], &n1X.g[i], &n1X.b[i])
92. YAZ ("%d\n",s3)
93. fgets(buf, 100, dosya4) değil iken NULL
94. s4++
95. s++
96. YAZ ("4- %d. satir: \"%s\"\n", s4, buf)
97. i=6’dan i=s4 olana kadar i’yi arttır
98. fscanf(dosya4,"%f %f %f",&n2.x[i], &n2.y[i], &n2.z[i])
99. YAZ ("%d\n",s4)
100. fgets(buf, 100, dosya5) değil iken NULL
101. s5++
102. s++
103. YAZ ("5- %d. satir: \"%s\"\n", s5, buf)
104. i=6’dan i=s5 olana kadar i’yi arttır
105. fscanf(dosya5,"%f %f %f",&n2.x[i], &n2.y[i], &n2.z[i])
106. YAZ ("%d\n",s5)
107. fgets(buf, 100, dosya6) değil iken NULL
108. s6++
109. s++
110. YAZ ("6- %d. satir: \"%s\"\n", s6, buf)
111. i=6’dan i=s6 olana kadar i’yi arttır
112. fscanf(dosya6,"%f %f %f",&n2.x[i], &n2.y[i], &n2.z[i])
113. YAZ ("%d\n",s6)
114. fgets(buf, 100, dosya7) değil iken NULL
115. s7++
116. s++
117. YAZ ("7- %d. satir: \"%s\"\n", s7, buf)
118. i=6’dan i=s2 olana kadar i’yi arttır
119. fscanf(dosya7,"%f %f %f",&n2.x[i], &n2.y[i], &n2.z[i])
120. YAZ ("%d\n",s7)
121. YAZ "--------------------------------\n"
122. YAZ("%d \n", s1);
123. YAZ("%d \n", s2);
124. YAZ("%d \n", s3);
125. YAZ("%d \n", s4);
126. YAZ("%d \n", s5);
127. YAZ("%d \n", s6);
128. YAZ("%d \n", s7);
129. YAZ("%d \n", s);
130. BREAK
131. CASE 2
132. BREAK
133. CASE 3
134. i=6’dan i=s1 olana kadar i’yi arttır
135. fscanf (dosya1,"%f %f %f %d %d %d",&n1.x[i], &n1.y[i], &n1.z[i], &n1.r[i], &n1.g[i], &n1.b[i])
136. BREAK
137. CASE 4
138. YAZ Koordinat ve yaricap bilgilerini giriniz: (x y z r formatinda)\n
139. OKU cx
140. OKU cy
141. OKU cz
142. OKU cr
143. BREAK
144. CASE 5
145. BREAK
146. DEFAULT
147. YAZ "Hatali bir numara girdiniz!\n"
148. fclose(dosya1);
149. fclose(dosya2);
150. fclose(dosya3);
151. fclose(dosya4);
152. fclose(dosya5);
153. fclose(dosya6);
154. fclose(dosya7);

**4.Sonuç**

Kullanıcıya girdiği seçime göre üç boyutlu nokta ve dosya kontrolü sağlayan program Yazılmıştır. Kullanıcı istediği işlemin komutunu vererek algoritma kısmında tanımlanan işlemlerin çıktısını gösterir.

**5.Ekran Görüntüleri**



**6. Kaynakça**

[1]<https://drive.google.com/drive/folders/1EF1e6nB3I4Z_PEFCYKMCQ4mzkeqmdFDE>

[2]<https://stackoverflow.com/questions/28266965/fscanf-strings-and-arrays-in-c>

[3]<https://stackoverflow.com/questions/41139220/how-to-convert-binary-data-to-float-value>

[4]<https://stackoverflow.com/questions/1422817/how-to-read-a-float-from-binary-file-in-c>

[5]<https://www.codeproject.com/Questions/257842/How-Read-Float-value-from-Binary-File>

[6]<https://stackoverflow.com/questions/16519056/convert-ascii-char-to-hexadecimal-char-in-c>

[7]<https://www.includehelp.com/c/convert-ascii-string-to-hexadecimal-string-in-c.aspx>

[8]<https://stackoverflow.com/questions/1422817/how-to-read-a-float-from-binary-file-in-c>