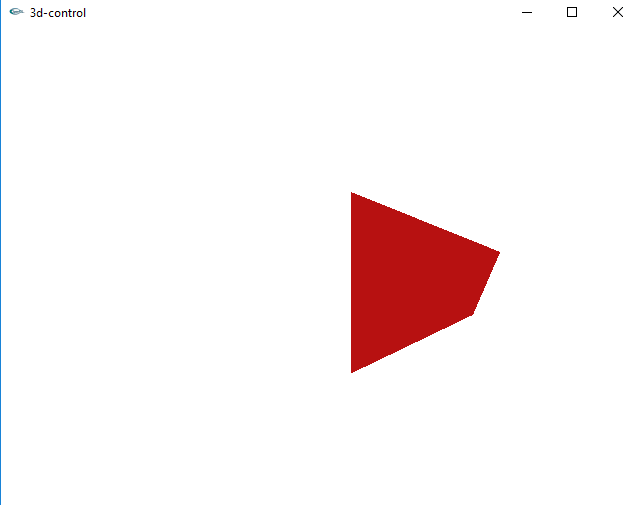
TUGAS PG 3



1. #include <windows.h>
2. #include "GL/glut.h"
3. #include "stdio.h"
5. **float** view\_rotx = 20.0f, view\_roty = 30.0f;
6. **int** oldMouseX, oldMouseY;
7. **float** R\_Z=0.0f, R\_X=0.0f, R\_Y=0.0f;
8. **float** T\_Z=-16.0f, T\_X=0.0f, T\_Y=0.0f;
10. **void** initGL(){
11. glShadeModel(GL\_FLAT);
13. **float** ambient[] = {1.0f,1.0f,1.0f,1.0f};
14. **float** diffuse[] = {1.0f,1.0f,1.0f,1.0f};
15. **float** specular[] = {0.2f,1.0f,0.2f,1.0f};
16. **float** position[] = {20.0f,30.0f,20.0f,0.0f};
18. glLightfv(GL\_LIGHT0, GL\_AMBIENT, ambient);
19. glLightfv(GL\_LIGHT0, GL\_DIFFUSE, diffuse);
20. glLightfv(GL\_LIGHT0, GL\_POSITION, position);
21. glMaterialfv(GL\_FRONT, GL\_SPECULAR, specular);
23. **float** mambient[] ={0.1745f, 0.01175f, 0.01175f, 0.55f};
24. **float** mdiffuse[] ={0.61424f, 0.04136f, 0.04136f, 0.55f };
25. **float** mspecular[] ={0.727811f, 0.626959f, 0.626959f, 0.55f };
26. **float** mshine =76.8f;
28. glMaterialfv(GL\_FRONT,GL\_AMBIENT,mambient);
29. glMaterialfv(GL\_FRONT,GL\_DIFFUSE,mdiffuse);
31. glMaterialfv(GL\_FRONT,GL\_SPECULAR,mspecular);
32. glMaterialf (GL\_FRONT,GL\_SHININESS,mshine);
34. glEnable(GL\_LIGHTING);
35. glEnable(GL\_LIGHT0);
36. glEnable(GL\_DEPTH\_TEST);
37. glEnable(GL\_NORMALIZE);
38. }
40. **void** Kubus(){
41. glBegin(GL\_POLYGON);/\* f1: Surface bagian depan \*/
42. glColor3f(1,0,1);
43. glVertex3f(0.0f,0.0f,0.0f);
44. glVertex3f(0.0f,0.0f,1.0f);
45. glVertex3f(1.0f,0.0f,1.0f);
46. glVertex3f(1.0f,0.0f,0.0f);
47. glEnd();
48. glColor3f(1,1,0);
49. glBegin(GL\_POLYGON);/\* f2: Surface bagian bawah (ABFE) \*/
50. glVertex3f(0.0f,0.0f,0.0f);
51. glVertex3f(1.0f,0.0f,0.0f);
52. glVertex3f(1.0f,1.0f,0.0f);
53. glVertex3f(0.0f,1.0f,0.0f);
54. glEnd();
55. glBegin(GL\_POLYGON);/\* f3:Surface bagian belakang (CDHG)\*/
56. glVertex3f(1.0f,1.0f,0.0f);
57. glVertex3f(1.0f,1.0f,1.0f);
58. glVertex3f(0.0f,1.0f,1.0f);
59. glVertex3f(0.0f,1.0f,0.0f);
60. glEnd();
61. glBegin(GL\_POLYGON);/\* f4: Surface bagian atas (EFGH)\*/
62. glVertex3f(1.0f,1.0f,1.0f);
63. glVertex3f(1.0f,0.0f,1.0f);
64. glVertex3f(0.0f,0.0f,1.0f);
65. glVertex3f(0.0f,1.0f,1.0f);
66. glEnd();
67. glBegin(GL\_POLYGON);/\* f5: Surface bagian kiri
68. (ADEH)\*/
69. glVertex3f(0.0f,0.0f,0.0f);
70. glVertex3f(0.0f,1.0f,0.0f);
71. glVertex3f(0.0f,1.0f,1.0f);
72. glVertex3f(0.0f,0.0f,1.0f);
73. glEnd();
74. glBegin(GL\_POLYGON);/\* f6: Surface bagian kanan (BCFG)\*/
75. glVertex3f(1.0f,0.0f,0.0f);
76. glVertex3f(1.0f,0.0f,1.0f);
77. glVertex3f(1.0f,1.0f,1.0f);
78. glVertex3f(1.0f,1.0f,0.0f);
79. glEnd();
80. }
81. **void** LimasPoli(){
82. glColor3f(1,1,0);
83. glBegin(GL\_POLYGON);/\* f1: Surface bagian depan \*/
84. glColor3f(1,0,1);
85. glVertex3f(0.0f,0.0f,0.0f); //B
86. glVertex3f(1.0f,0.0f,0.0f); //C
87. glVertex3f(0.5f,0.5f,1.0f); //A
88. glEnd();
89. glBegin(GL\_POLYGON);
90. glVertex3f(1.0f,0.0f,0.0f); //C
91. glVertex3f(0.5f,1.0f,0.0f); //D
92. glVertex3f(0.5f,0.5f,1.0f); //A
93. glEnd();
94. glBegin(GL\_POLYGON);
95. glVertex3f(0.5f,0.5f,1.0f); //A
96. glVertex3f(0.5f,1.0f,0.0f); //D
97. glVertex3f(0.0f,0.0f,0.0f); //B
98. glEnd();
99. glBegin(GL\_POLYGON);/\* f2: Surface bagian bawah (ABFE) \*/
100. glVertex3f(0.0f,0.0f,0.0f); //B
101. glVertex3f(1.0f,0.0f,0.0f); //C
102. glVertex3f(0.5f,1.0f,0.0f); //D
103. glEnd();
104. }
105. **void** LimasLine(){

108. glBegin(GL\_LINES);/\* f2: Surface bagian bawah (ABFE) \*/
109. glVertex3f(0.0f,0.0f,0.0f); //B
110. glVertex3f(1.0f,0.0f,0.0f); //C
112. glVertex3f(1.0f,0.0f,0.0f); //C
113. glVertex3f(0.5f,1.0f,0.0f); //D
115. glVertex3f(0.5f,1.0f,0.0f); //D
116. glVertex3f(0.0f,0.0f,0.0f); //B
117. glEnd();
119. glBegin(GL\_LINES);/\* f1: Surface bagian depan \*/
120. glVertex3f(0.5f,0.5f,1.0f); //A
121. glVertex3f(0.0f,0.0f,0.0f); //B
122. glVertex3f(0.5f,0.5f,1.0f); //A
123. glVertex3f(1.0f,0.0f,0.0f); //C
124. glVertex3f(0.5f,0.5f,1.0f); //A
125. glVertex3f(0.5f,1.0f,0.0f); //D
126. glEnd();
127. }
128. **void** Prisma(){
129. glBegin(GL\_POLYGON);/\* f1: Surface bagian depan \*/
130. glVertex3f(0.0f,0.0f,0.0f);
131. glVertex3f(0.0f,0.0f,1.0f);
132. glVertex3f(1.0f,0.0f,1.0f);
133. glVertex3f(1.0f,0.0f,0.0f);
134. glEnd();
135. glBegin(GL\_POLYGON);/\* f2: Surface bagian bawah (ABFE) \*/
136. glVertex3f(0.0f,0.0f,0.0f);
137. glVertex3f(1.0f,0.0f,0.0f);
138. glVertex3f(0.5f,1.0f,0.0f);
139. glEnd();
140. glBegin(GL\_POLYGON);/\* f4: Surface bagian atas (EFGH)\*/
141. glVertex3f(0.5f,1.0f,1.0f);
142. glVertex3f(1.0f,0.0f,1.0f);
143. glVertex3f(0.0f,0.0f,1.0f);
144. glEnd();
145. glBegin(GL\_POLYGON);/\* f5: Surface bagian kiri
146. (ADEH)\*/
147. glVertex3f(0.0f,0.0f,0.0f);
148. glVertex3f(0.5f,1.0f,0.0f);
149. glVertex3f(0.5f,1.0f,1.0f);
150. glVertex3f(0.0f,0.0f,1.0f);
151. glEnd();
152. glBegin(GL\_POLYGON);/\* f6: Surface bagian kanan (BCFG)\*/
153. glVertex3f(1.0f,0.0f,0.0f);
154. glVertex3f(1.0f,0.0f,1.0f);
155. glVertex3f(0.5f,1.0f,1.0f);
156. glVertex3f(0.5f,1.0f,0.0f);
157. glEnd();
159. }
160. **void** keyFunction(unsigned **char** key, **int** x, **int** y){
161. **switch**(key){
162. **case** 85: // Rotasi sumbu Z dengan tombol U
163. R\_Z = R\_Z + -30.0f;
164. **break**;
165. **case** 75: // Rotasi sumbu Y dengan tombol K
166. R\_Y = R\_Y + -30.0f;
167. **break**;
168. **case** 74: // Rotasi sumbu X dengan tombol J
169. R\_X = R\_X + -30.0f;
170. **break**;
171. **case** 81: // Translasi sumbu Z dengan tombol Q
172. T\_Z = T\_Z + -1.0f;
173. **break**;
174. **case** 69: // Translasi sumbu Z dengan tombol E
175. T\_Z = T\_Z + 1.0f;
176. **break**;
177. **case** 83: // Translasi sumbu Y dengan tombol S
178. T\_Y = T\_Y + -1.0f;
179. **break**;
180. **case** 87: // Translasi sumbu Y dengan tombol W
181. T\_Y = T\_Y + 1.0f;
182. **break**;
183. **case** 65: // Translasi sumbu X dengan tombol A
184. T\_X = T\_X + -1.0f;
185. **break**;
186. **case** 68: // Translasi sumbu X dengan tombol A
187. T\_X = T\_X + 1.0f;
188. **break**;
190. }
191. }
192. **float** sudut = 0;
193. **void** display(){
194. glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);
195. glLoadIdentity();
196. //gluLookAt(4,4,4, // eye pos
197. //0,0,0, // look at
198. //0,0,1); // up
199. glClearColor(1.0f, 1.0f, 1.0f, 1.0f);
200. //glTranslatef(0.5f, 0.5f, 0.5f);
201. //glRotatef(90, 0,0,1);
202. //glRotatef(view\_rotx, 1.0f, 0.0f, 0.0f);
203. //glRotatef(view\_roty, 0.0f, 1.0f, 0.0f);
204. //glTranslatef(-0.5f, -0.5f, -0.5f);
206. //WITH BUTTON
207. glTranslatef(-T\_X, T\_Y, T\_Z);
208. glRotatef(-90.0f, 1.0f, 0.0f, 0.0f);
209. glRotatef(180.0f, 0.0f, 0.0f, 1.0f);
211. glRotatef(R\_X, 1.0f, 0.0f, 0.0f);
212. glRotatef(R\_Y, 0.0f, 1.0f, 0.0f);
213. glRotatef(R\_Z, 0.0f, 0.0f, 1.0f);
215. Prisma();
216. sudut++;
217. glFlush();
218. glutSwapBuffers();
219. }
221. **void** timer(**int** value){
222. glutPostRedisplay();
223. glutTimerFunc(15, timer, 0);
224. }
226. **void** reshape(GLsizei width, GLsizei height){
227. **if** (height == 0) height = 1;
228. GLfloat aspect = (GLfloat)width / (GLfloat)height;
229. glViewport(30, 6, width, height);
230. glMatrixMode(GL\_PROJECTION);
231. glLoadIdentity();
232. gluPerspective(45.0f, aspect, 1.0f, 20.0f);
233. glMatrixMode(GL\_MODELVIEW);
234. glLoadIdentity();
235. }
237. **void** mouseControl(**int** button, **int** state, **int** x, **int** y){
238. oldMouseX = x;
239. oldMouseY = y;
240. }
242. **void** mouseMotion(**int** x, **int** y){
243. **int** getX = x;
244. **int** getY = y;
245. **float** thetaY = 360.0f\*(getX - oldMouseX)/640;
246. **float** thetaX = 360.0f\*(getY - oldMouseY)/480;
247. oldMouseX = getX;
248. oldMouseY = getY;
249. view\_rotx += thetaX;
250. view\_roty += thetaY;
251. }
253. **int** main(**int** argc, **char** \*\*argv){
254. glutInit(&argc, argv);
255. glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_DEPTH);
256. glutInitWindowSize(640, 480);
257. glutInitWindowPosition(50, 50);
258. glutCreateWindow("3d-control");
259. glutDisplayFunc(display);
260. glutReshapeFunc(reshape);
261. glutKeyboardFunc(keyFunction);
262. initGL();
263. glutMouseFunc(mouseControl);
264. glutMotionFunc(mouseMotion);
265. glutTimerFunc(0, timer, 0);
266. glutMainLoop();
267. **return** 0;
268. }