

III. gyakorlat - Jegyzőkönyv

Ekart Csaba - ZWPMKP

1. feladat

Az első feladatban az egyszerű transzformációkat kellett használni, amelyeket a kis mosolygós fejen alkalmaztunk. A feladat leírásnak megfelelően a megadott függvényeket használtam, és egy-egy gombbal változóban tároltam az elfordulás szögét, a mértet, illetve a pozícióját a fejnek. Ennek megfelelően rajzolta ki a redraw.

```
104 //Mozgassuk a fejet a képernyő középpontjába!
105 glTranslated(fej_x,fej_y,0);
106
107 glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
108 glEnable(GL_POLYGON_SMOOTH);
109 glRotatef(angle, 0.0, 0.0, 1.0);
110 glScalef(scale,scale,1);
111 drawHead();
112
113 glutSwapBuffers();
114
```

A gombnyomások hatására a megfello értékek változnak.

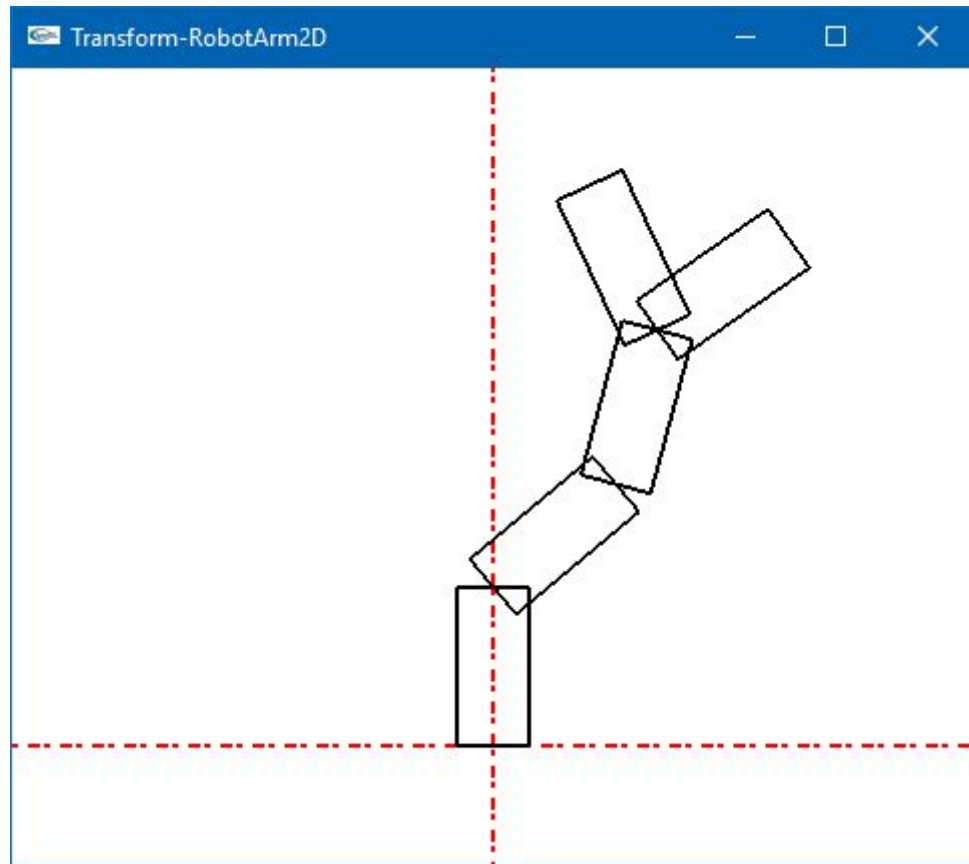
2. feladat

A feladat során egy háromcsuklójú robotkart kellett készíteni. Először a tengelyek kirajzolásához egy külön függvényt hoztam létre, melyet egyből el is helyeztem a redraw()-ban.

```
166 ///1. darabka
167 drawSegment();
168 ///2. darabka
169 glPushMatrix();
170 glTranslatef(0,segmentSizeY,0);
171 glRotatef(angle_joint1,0,0,1);
172 drawSegment();
173 //glPopMatrix();
174 ///3. darabka
175 //glPushMatrix();
176 glTranslatef(0,segmentSizeY,0);
177 glRotatef(angle_joint2,0,0,1);
178 drawSegment();
179 //glPopMatrix();
180 ///4. darabka
181 //glPushMatrix();
182 glTranslatef(0,segmentSizeY,0);
183 glRotatef(angle_joint3,0,0,1);
184 drawSegment();
185 ///5. darabka
186 glTranslatef(0,0,0);
187 glRotatef(-2*angle_joint3,0,0,1);
188 drawSegment();
189 glPopMatrix();
```

Mivel ugyanazon elemet kellett többször ábrázolni, ezért mindig ugyanazt a rajzoló függvényt hívtam meg, csak egy-egy eltolást alkalmaztam, hogy a forgatási középpont mindig új helyzetbe kerüljön.

A megfelelő értékeket a gombnyomások az előző feladathoz hasonlóan növelik.



3. feladat

A harmadik feladat az előzőhöz nagyon hasonló volt, csak 3D-s reprezentációban. Az előző módszerhez rendkívül hasonlóan hoztam létre a kódot. A keretrendszer kényelmetlenségét kiküszöbölve néhány módosítást végrehajtottam a program többi részén is.

Az alábbi képmetszeten látható, hogy miként implementáltam a `drawSegment()` függvényt. Az előzőekhez képest kicsit trükkösebb megoldásra kényszerültem az átskálázás miatt, illetve a megfelelő pont körüli forgatás elérése végett. Ezért ezeket a transzformációkat két darabban végeztem el a forgatás előtt és után. Így a forgatás a megfelelő pontok körül történik, és közben az alakzatokon se történik deformáció. Vélhetően ennél van szebb megoldás, de funkcionalitásában ez is megfelelő.

```

26 void drawSegment() {
27
28     glPushMatrix();
29     glTranslated(0,0.7,0);
30     glScalef (0.5, height, 0.5);
31
32     glRotatef(base_angle, 0, 1, 0);
33     glutWireCube (1.0);
34
35     glTranslated(0, 0.5, 0);
36     glScalef (2, 1/height, 2);
37     glRotatef(angle_joint1, 0, 0, 1);
38     glScalef (0.5, height, 0.5);
39     glTranslated(0, 0.5, 0);
40     glutWireCube (1.0);
41
42     glTranslated(0, 0.5, 0);
43     glScalef (2, 1/height, 2);
44     glRotatef(angle_joint2, 0, 0, 1);
45     glScalef (0.5, height, 0.5);
46     glTranslated(0, 0.5, 0);
47     glutWireCube (1.0);
48
49
50     glPopMatrix();
51
52
53 }

```

