

SPRAWOZDANIE

Zajęcia: Grafika Komputerowa

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium: 6

Data: 15.04.2024

Temat: "Światło i Materiały w OpenGL"

Michał Michalik
Informatyka I stopień,
stacjonarne,
4 semestr,
Gr.3a

Zadanie 1

1. Polecenie:

Celem zadania jest stworzenie piramidy z różnymi materiałami, umieszczając ją na podstawie. Użytkownik może obracać podstawę wokół osi Y, przeciągając mysz w poziomie. Scena korzysta z globalnego światła otoczenia (ambient) oraz źródła światła o kształcie kuli z możliwością animacji obrotu wokół piramidy. Aby wykonać to zadanie w JavaScript, zaleca się zapoznanie się z plikami .html: four-lights-demo.html oraz materials-demo.html.

2. Wprowadzane dane:

```
236 function drawPyramid(num) {  
237     const angleIncrement = 2 * Math.PI / num;  
238     angle = 0;  
239     var m = materialProperties[5];  
240     var s = Math.round(m[1]);  
241     var dr = m[4];  
242     var dg = m[5];  
243     var db = m[6];  
244     var sr = m[8];  
245     var sg = m[9];  
246     var sb = m[10];  
247     //glColor3d(1,1,1)  
248     glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT_AND_DIFFUSE, [dr,dg,db,1]);  
249     glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, [sr, sg, sb, 1]);  
250     glMateriali(GL_FRONT_AND_BACK, GL_SHININESS, s);  
251  
252     for (let i = 1; i <= num; i++) {  
253         glBegin(GL_TRIANGLE_FAN);  
254         x = radius * Math.cos(angle);  
255         z = radius * Math.sin(angle);  
256  
257         glVertex3d(x, -2.0, z);  
258         glVertex3d(0, 5, 0);  
259         angle = i * angleIncrement;  
260         glVertex3d(radius * Math.cos(angle), -2.0, radius * Math.sin(angle));  
261         glEnd();  
262     }  
263  
264 }
```

```

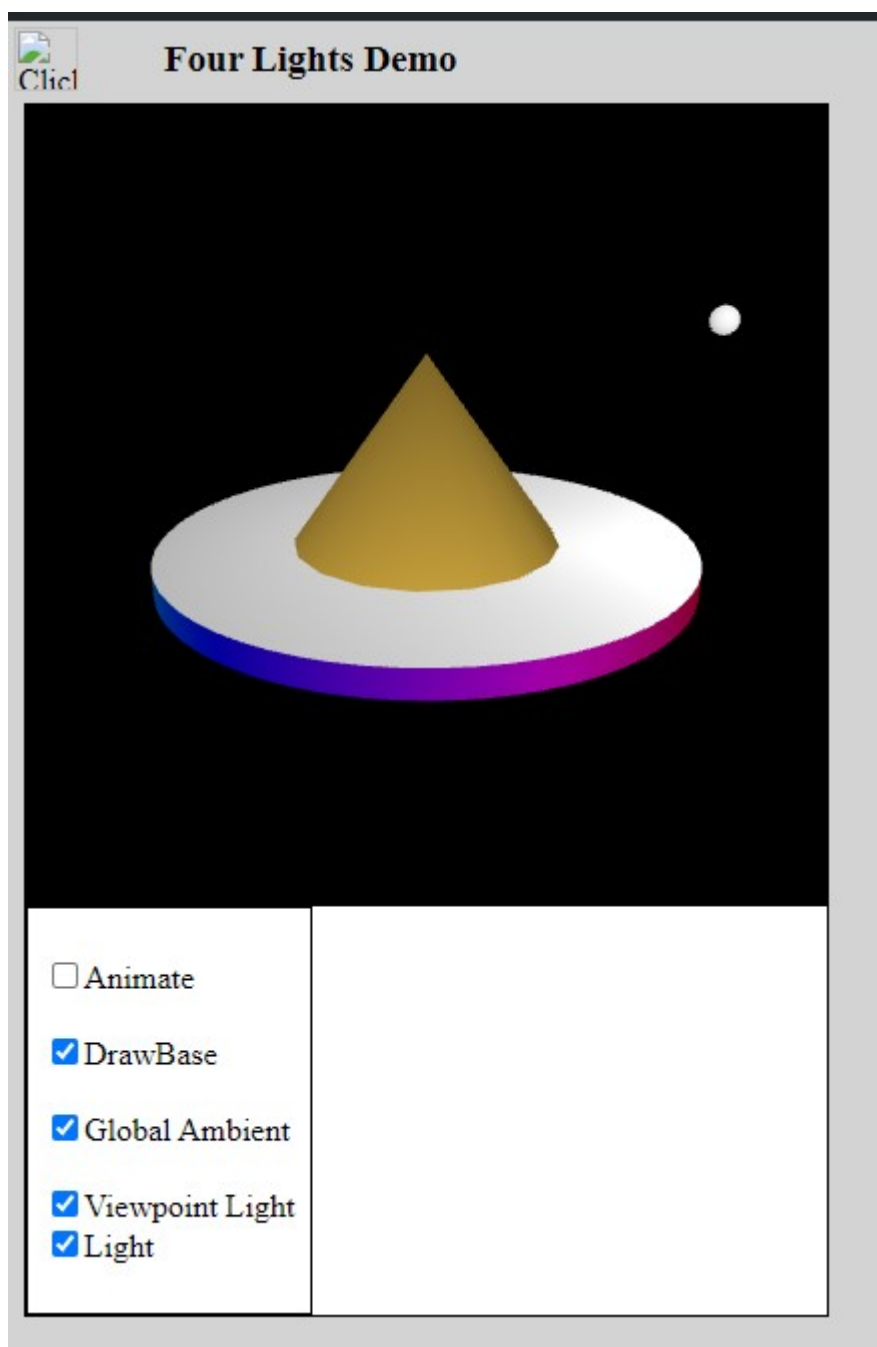
279  ✓ function initGL() {
280      glClearColor(0,0,0,1);
281      glEnable(GL_LIGHTING);
282      glEnable(GL_LIGHT0);
283  ✓  //glLightfv(GL_LIGHT0, GL_SPECULAR, [0.4,0.4,0.4,1]);
284      //glLightfv(GL_LIGHT0, GL_DIFFUSE, [1,1,1,1]);
285      glEnable(GL_NORMALIZE);
286      glEnable(GL_DEPTH_TEST);
287
288
289      glEnable(GL_COLOR_MATERIAL);
290      glLightModeli(GL_LIGHT_MODEL_LOCAL_VIEWER, 1);
291      glMateriali(GL_FRONT_AND_BACK, GL_SHININESS, 32);
292
293      var dim = [ 0.5, 0.5, 0.5, 1 ];
294      glLightfv(GL_LIGHT0, GL_DIFFUSE, dim);
295      glLightfv(GL_LIGHT0, GL_SPECULAR, dim);
296
297      var glob = [ 1.0, 1.0, 1.0, 1.0];
298      var globa = [ 0.5, 0.5, 0.5, 1];
299      glLightfv(GL_LIGHT1, GL_AMBIENT, globa);
300      glLightfv(GL_LIGHT1, GL_DIFFUSE, glob);
301      glLightfv(GL_LIGHT1, GL_SPECULAR, glob);
302  }
303

```

3. Wykorzystane komendy:

Link git dodac

4. Wynik działania:



5. Wnioski:

Dzięki funkcjom zarządzania kolorami, oświetleniem i innymi właściwościami w bibliotece OpenGL, użytkownicy mają kontrolę nad tym, co jest wizualnie wyróżnione. To pozwala na skupienie uwagi na kluczowych elementach sceny oraz tworzenie interaktywnych i estetycznych wizualizacji.