

# **SPRAWOZDANIE**

Zajęcia: Grafika komputerowa

Prowadzący: mgr inż. Mikołaj Grygiel

## **Laboratorium 9**

30.04.2025

**Temat:** "Modelowanie hierarchiczne w grafice 2D"

**Wariant 6**

Bartłomiej Mędrzak

s61324

Informatyka I stopień,

stacjonarne,

4 semestr,

Gr.1A

## 1. Polecenie:

Celem jest konstruowanie modelu figury szachowej zgodnie z wariantem zadania (patrz rysunek) używając three.js w oparciu na omówione na zajęcie metody konstruowania obiektów



## 2. Wprowadzane dane:

Figura nr 6 - król

## 3. Wykorzystane komendy:

```
function createWorld() {  
  renderer.setClearColor("#ffffff");  
  scene = new THREE.Scene();  
  
  let colour = 0x505050;  
  
  camera = new THREE.PerspectiveCamera(30, canvas.width / canvas.height, 0.1, 100);  
  camera.position.set(-3, 7, 35);  
  
  let light = new THREE.DirectionalLight(0xffffff, 1.0);  
  light.position.set(5, 10, 7.5);  
  scene.add(light);  
  
  let light2 = new THREE.DirectionalLight(0xffffff, 0.5);  
  light2.position.set(-5, 5, -5);  
  scene.add(light2);  
  
  let ambientLight = new THREE.AmbientLight(0xaaaaaa, 0.4);  
  scene.add(ambientLight);  
  
  let pieceMaterial = new THREE.MeshPhongMaterial({ color: colour });  
  
  let part1 = new THREE.Mesh(  
    new THREE.CylinderGeometry(4, 4, 1, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
  );  
  part1.position.y = -7;  
  
  let part2 = new THREE.Mesh(  
    new THREE.CylinderGeometry(2.5, 4, 2, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
  );  
  part2.position.y = -6.6;  
  
  let part3 = new THREE.Mesh(  
    new THREE.CylinderGeometry(1.5, 3, 10, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
  );  
  part3.position.y = -1.5;
```

```
let part4 = new THREE.Mesh(  
    new THREE.CylinderGeometry(1.5, 2, 11, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part4.position.y = 2;  
  
let part5 = new THREE.Mesh(  
    new THREE.CylinderGeometry(2.5, 2.5, 0.5, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part5.position.y = 5;  
  
let part6 = new THREE.Mesh(  
    new THREE.CylinderGeometry(2, 2, 0.5, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part6.position.y = 5.5;  
  
let part7 = new THREE.Mesh(  
    new THREE.CylinderGeometry(1.5, 2, 0.5, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part7.position.y = 6;  
  
let part8 = new THREE.Mesh(  
    new THREE.CylinderGeometry(3, 0.1, 4, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part8.position.y = 7;  
  
let part9 = new THREE.Mesh(  
    new THREE.CylinderGeometry(1.5, 3, 1, 32, 1, false, 0, Math.PI * 2),  
    pieceMaterial  
);  
part9.position.y = 9.5;  
  
let part10 = new THREE.Mesh(  
    new THREE.BoxGeometry(1, 6, 1),  
    pieceMaterial  
);  
part10.position.y = 9.5;  
  
let part11 = new THREE.Mesh(  
    new THREE.BoxGeometry(2, 1, 1),  
    pieceMaterial  
);  
part11.position.y = 11.5;
```

```
scene.add(part1);  
scene.add(part2);  
scene.add(part3);  
scene.add(part4);  
scene.add(part5);  
scene.add(part6);  
scene.add(part7);  
scene.add(part8);  
scene.add(part9);  
scene.add(part10);  
scene.add(part11);  
  
scene.add(camera);
```

[https://github.com/castehard33/Grafika\\_Komputerowa/tree/main/9%20Konstruowanie%20obiekt%C3%B3w%20z%20u%C5%BCyciem%20Three.js](https://github.com/castehard33/Grafika_Komputerowa/tree/main/9%20Konstruowanie%20obiekt%C3%B3w%20z%20u%C5%BCyciem%20Three.js)

#### 4. Wynik działania:



**5. Wnioski:** Ćwiczenie pozwoliło na praktyczne przećwiczenie budowy złożonych obiektów 3D w bibliotece Three.js poprzez strategiczne łączenie i pozycjonowanie wielu prostszych, prymitywnych geometrii (głównie cylindrów i prostopadłościanów). Kluczowe okazało się precyzyjne ustalanie wymiarów oraz współrzędnych każdego elementu składowego, aby uzyskać zamierzony kształt finalny, taki jak figura szachowa.