

Übung 11: Kamera

1. Worin unterscheiden sich die Szenen, wenn sie mit perspektivischer bzw. Parallelprojektion dargestellt werden?
2. Wie sehen die Sichtkörper für beide Transformationsarten aus?
3. Wozu werden die near-Clipplane und far-Clipplane verwendet?
4. Welcher Unterschied ergibt sich in der Darstellung mit perspektivischer Projektion, wenn in dem einen Fall der Kamerastandpunkt in Richtung positiver z-Achse um 10 verschoben wird und in dem anderen Fall alle Objekte entlang der negativen z-Achse um 10 verschoben werden?
5. Geben Sie an, ob bei der gegebenen Einstellung der Kamera mit `THREE.OrthographicCamera (-4.0, 4.0, 4.0, -4.0, 0.1, 100.0)`; der Punkt `P(0,0,4)` sichtbar ist.

Praktischer Teil:

Gegeben ist das Testprogramm.

- Wo muss der Betrachter stehen, damit die Bäume und der Avatar von vorn sichtbar sind?
- Vergleichen Sie das Ergebnis der Darstellung mit perspektivischer und orthographischer Kamera! (`THREE.OrthographicCamera()`, `THREE.PerspectiveCamera()`)
- Testen Sie, was die Änderung der x- bzw. y-Koordinaten der Kamera bewirkt!
- Wie müssen die Parameter für den View-Point (`lookAt-Methode`) und für den up-Vektor gewählt werden, damit der Betrachter die Szene von links bzw. von oben sieht?
- Schreiben Sie eine Erweiterung für das Programm, so dass sich der Betrachter abhängig von einem Tastaturevent bewegt, also die Betrachterposition geändert wird! Setzen einer Variablen für die Bewegungsrichtung anhand des Drückens der Pfeil-Tasten (37 = Bewegen nach links, 38 = Bewegen nach hinten, 39 = Bewegen nach rechts, 40 = Bewegen nach vorn) in der Funktion `key_down` auf `true` und in `key_up` auf `false`! In der `animate-Funktion` wird dann bei der jeweiligen Richtung die Kamera verschoben (Methode `translateX()` bzw. `translateZ()`).
- Animieren Sie die eine Änderung der Objektiveneinstellung! Erhöhen des Öffnungswinkels der Kamera bis maximal 165° (Weitwinkel) bei „+“-Taste (Keycode 171) und Erniedrigen des Öffnungswinkels bis minimal 10° (Teleobjektiv) bei „-“-Taste (Keycode 173)! Nutzen Sie hierfür die perspektivische Kamera! Nach jeder Änderung der Kameraeinstellungen muss die Methode `updateProjectionMatrix()` aufgerufen werden!
- Testen Sie, was passiert, wenn Sie die Kamera mit dem Avatar verbinden und statt der Betrachterposition den Avatar bewegen!