

Syahreza Abror Alvarizqi

24060123140053

GKV E1

Menggunakan Kamera dalam OpenGL

1. Jelaskan apa saja yang harus dilakukan agar dapat menggerakkan objek dengan keyboard.
 - Gunakan callback fungsi keyboard dari GLFW untuk menangkap input keyboard.
 - Membuat variable untuk menyimpan status gerakannya
 - Tambahkan algoritma untuk menggerakkan dengan menggunakan keyboard

Contoh:

```
if (action == GLFW_PRESS || action == GLFW_REPEAT) {
    switch (key) {
        case GLFW_KEY_A:
            rotAngle += 5; // rotasi ke kanan (sumbu Y)
            break;
        case GLFW_KEY_L:
            rotAngle -= 5; // rotasi ke kiri (sumbu Y)
            break;
        case GLFW_KEY_Y:
            rotAngle1 += 5; // rotasi ke atas (sumbu X)
            break;
        case GLFW_KEY_B:
            rotAngle1 -= 5; // rotasi ke bawah (sumbu X)
            break;
        case GLFW_KEY_Q:
            glfwSetWindowShouldClose(window, GLFW_TRUE); // keluar
            break;
    }
}
```

2. Dari kode di atas, terdapat dua proyeksi (Projection, dan Model View), Jelaskan cara kerjanya. Mengapa keduanya digunakan?

```
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
float aspect = (float)width / (float)height;
glFrustum(-aspect, aspect, -1.0, 1.0, 1.0, 1000.0);
```

Penjelasan:

- untuk **mengatur cara pandang kamera terhadap dunia 3D**, yaitu **mengubah koordinat 3D ke koordinat 2D** untuk ditampilkan ke layar.
- `glMatrixMode(GL_PROJECTION)` memberitahu OpenGL bahwa kita sedang mengatur **matriks proyeksi**.

- glFrustum membuat perspektif 3D dengan parameter seperti **lebar field of view**, **jarak dekat (near)**, dan **jarak jauh (far)** dari kamera.

- | |
|--|
| <ul style="list-style-type: none">• glMatrixMode(GL_MODELVIEW);• glLoadIdentity();• glTranslatef(0.0f, 0.0f, -5.0f); |
|--|

Penjelasan:

- **Transformasi View** → menentukan posisi dan orientasi kamera (mata pengamat).
- **Transformasi Model** → menentukan posisi, rotasi, dan skala objek dalam dunia 3D.
- glMatrixMode(GL_MODELVIEW) artinya kita sekarang akan mengatur transformasi objek dan kamera.
- glLoadIdentity() mengatur ulang matriks.
- glTranslatef() digunakan untuk **menggeser kamera** ke belakang, sehingga objek bisa terlihat di layar.

Scripting Dept dan Lighting

1. Jelaskan Lighting yang ada di GLFW

GLFW tidak menyediakan lighting secara langsung. GLFW hanyalah library untuk membuat window, context OpenGL, dan menangani input keyboard/mouse.

- Namun, dalam dua kode tersebut, lighting dilakukan dengan cara manual dan sangat, yaitu: Mengatur warna sisi kubus (box) menggunakan `glColor3f(...)`.

2. Perbedaan versi 1 dan 2

Versi 1:

- Untuk kamera mengelilingi box, dan selalu melihat ke box
- Mekanisme geraknya kamera dinamis dan box static

Versi 2:

- Untuk kamera bebas bergerak dan berputar
- Mekanisme geraknya kamera dan box bisa digerakkan manual

3. Analisis: Bagaimana kubus, grid, dan pencahayaan tersebut dapat dibuat

- Kubus (Box):

Dibuat dengan fungsi `glBegin(GL_QUADS)` lalu menggambar 6 sisi kubus. Masing-masing sisi menggunakan warna berbeda (pakai `glColor3f`). Kubus bisa digerakkan (versi 2) dengan mengubah posisi koordinat sebelum menggambar: `glTranslatef(boxX, boxY, boxZ)`.

- Grid

Dibuat dengan `glBegin(GL_LINES)` menggunakan perulangan (`for`) menggambar garis-garis sejajar. Contoh: 20 garis dari -10 sampai +10 untuk sumbu X dan Z. Grid digunakan sebagai acuan visual lantai.

- Pencahayaan

Tidak ada pencahayaan dinamis di kedua versi. Warna kubus ditentukan manual (`glColor3f`), jadi lebih ke **color shading statis**. Tidak ada efek bayangan, pantulan, atau pencahayaan dari sumber cahaya.