

Nama : Shopia Mudjahidah

NIM : 24060123120010

Lab : GKV E1

Pertanyaan:

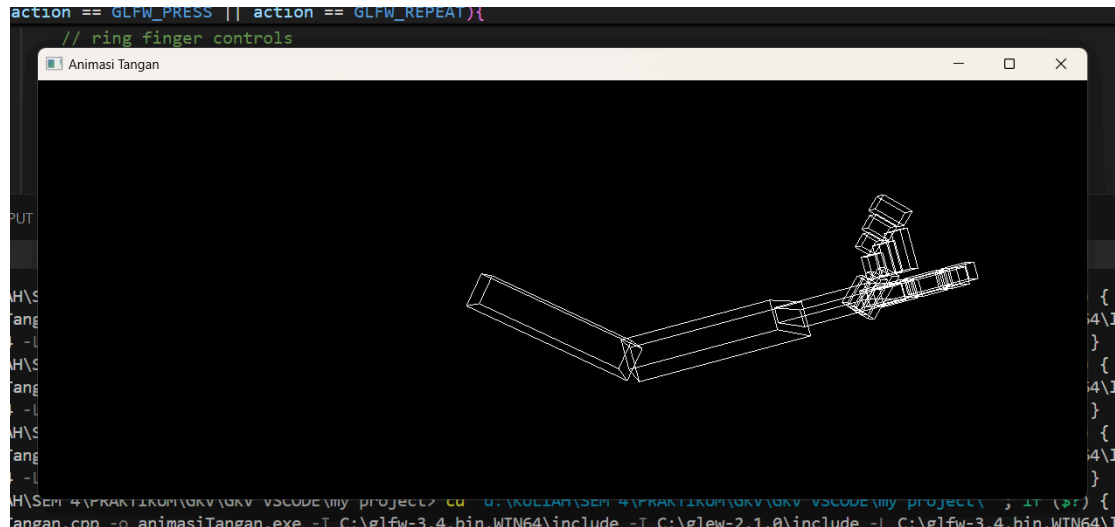
1. Jelaskan bagaimana cara kerja Pergerakan Tangan pada kode diatas?
2. Simulasikan menggunakan sumbu X, Y, Z bagaimana operasi yang dilakukan ketika keyboard diketik (dalam bentuk screenshot dan milimeter blok).
3. Jelaskan struktur dari tugas yang kalian buat dalam proyeksi, proyeksi projection dan modelview dalam!

Jawaban:

1. Kode di atas mensimulasikan pergerakan tangan. Struktur pergerakan terdiri dari dua bagian utama, yaitu bahu (shoulder) dan siku (elbow), dimana masing-masing bagian dapat berotasi secara independen pada sumbu Z. Pergerakan bahu dikontrol dengan fungsi `glRotatef(shoulder, 0.0, 0.0, 1.0)`, yang menggerakkan seluruh lengan atas, sedangkan pergerakan siku menggunakan `glRotatef(elbow, 0.0, 0.0, 1.0)`, yang menggerakkan lengan bawah secara relatif terhadap bahu. Untuk membentuk lengan, digunakan fungsi `DrawWireCube()`, yang menggambar objek berbentuk wireframe cube dengan skala yang diperbesar agar menyerupai bentuk lengan. Interaksi dengan keyboard memungkinkan pengguna menggerakkan bahu dengan tombol S dan W, serta menggerakkan siku dengan tombol E dan Q. Proses ini dikendalikan dalam fungsi `KeyCallback()`, dimana setiap penekanan tombol akan mengubah nilai rotasi bahu dan siku secara bertahap. Dengan menggunakan `glPushMatrix()` dan `glPopMatrix()`, setiap transformasi disimpan dan dipulihkan agar tidak mempengaruhi objek lain. Hasil akhirnya adalah animasi sederhana di mana lengan atas dan lengan bawah dapat bergerak secara realistis sesuai dengan input pengguna.
2. Pada program animasi tangan yang telah saya buat, setiap bagian tangan dapat digerakkan menggunakan tombol pada keyboard. Pergerakan ini berpengaruh terhadap rotasi bagian tertentu pada sumbu X, Y, atau Z, tergantung pada sendi yang bergerak. Gerakan bahu (shoulder) dipengaruhi oleh tombol S dan W, yang menyebabkan rotasi pada sumbu Z. Siku (elbow) dikontrol oleh tombol E dan Q, juga berputar pada sumbu Z tetapi dengan batas maksimal rotasi 180 derajat agar menyerupai pergerakan manusia yang sebenarnya. Pergelangan tangan (wrist) dikontrol oleh tombol D dan A, yang memungkinkan rotasi pada sumbu X dengan batasan -90 hingga 90 derajat, sesuai dengan cara manusia menekuk pergelangan tangannya. Jari-jari tangan juga memiliki tiga bagian sendi, yaitu pangkal (base), tengah (mid), dan ujung (tip). Masing-masing bagian ini dapat bergerak secara independen menggunakan kombinasi tombol tertentu. Sebagai contoh, jempol (thumb) dikendalikan oleh tombol T, G, Y, H, U, dan J, yang mempengaruhi rotasi

pada sumbu Z. Jari telunjuk (index finger), jari tengah (middle finger), jari manis (ring finger), dan kelingking (pinky finger) juga memiliki kontrol serupa.

Berikut ini merupakan simulasi gerakan pada bahu (shoulder), siku (elbow), jari tengah (middle finger), jari manis (ring finger), kelingking (pinky) yang menyebabkan rotasi pada sumbu Z.



Berikut ini merupakan simulasi gerakan pada pergelangan tangan (wrist) yang menyebabkan rotasi pada sumbu X.



3. Penjelasan Struktur dari Tugas yang Dikerjakan

a. Proyeksi

Proyeksi dalam program ini diatur menggunakan fungsi `gluPerspective()`, yang menentukan sudut pandang kamera dan bidang potong (clipping planes).

Berikut adalah bagian program yang menangani proyeksi:

```
198
199     glViewport(0, 0, (GLsizei)width, (GLsizei)height);
200     glMatrixMode(GL_PROJECTION);
201     glLoadIdentity();
202     gluPerspective(40.0, (GLdouble)width / (GLdouble)height, 0.5, 20.0);
203
```

`glMatrixMode(GL_PROJECTION);` mengaktifkan Projection Matrix, yang mengontrol cara objek 3D diproyeksikan ke layar 2D, `glLoadIdentity();` mengatur ulang matrix agar tidak ada transformasi sebelumnya yang mempengaruhi proyeksi, `gluPerspective(40.0, (GLdouble)width / (GLdouble)height, 0.5, 20.0);` akan mengatur perspektif dimana 40.0 merupakan sudut pandang vertikal dalam derajat, `(GLdouble)width / (GLdouble)height` merupakan rasio aspek layar, 0.5 merupakan jarak bidang dekat, dan 20.0 merupakan jarak bidang jauh.

b. Proyeksi Projection

Projection Matrix digunakan untuk mentransformasikan objek dari ruang 3D ke layar 2D. Proses ini memastikan bahwa objek yang lebih jauh dari kamera akan tampak lebih kecil sesuai dengan hukum perspektif. Dalam program ini, hanya ada satu kali pemanggilan Projection Matrix, yaitu pada bagian `Reshape()`. Dilakukan pemanggilan hanya sekali karena Projection Matrix hanya perlu diatur satu kali ketika ukuran jendela berubah, sedangkan pergerakan objek dikelola menggunakan transformasi model.

c. Modelview

Setelah proyeksi diatur, program menggunakan transformasi model untuk mengatur posisi, rotasi, dan skala objek tangan. Meskipun `glMatrixMode(GL_MODELVIEW);` tidak dipanggil secara eksplisit setelah proyeksi, OpenGL secara default menganggap semua transformasi objek setelah proyeksi sebagai bagian dari Modelview.

Berikut adalah bagian program yang menangani modelview:

```
204     glMatrixMode(GL_MODELVIEW);
205     glViewport(0, 0, width, height);
206 }
```

Transformasi dalam program ini juga dilakukan dengan menggunakan translasi (`glTranslatef()`) dan rotasi (`glRotatef()`), misalnya pada pergerakan tangan:

```
76 void Display(){
77     glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
78     glLoadIdentity();
79     glTranslatef(0.0, 0.0, -7.0);
80
81     glPushMatrix();
```

`glTranslatef(0.0, 0.0, -7.0);` memindahkan seluruh objek menjauh dari kamera agar terlihat lebih jelas di layar.

```

104
105 // wrist movement (gerakan pergelangan tangan)
106 glTranslatef(1.0, 0.0, 0.0);
107 glRotatef((GLfloat)wrist, 1.0, 0.0, 0.0);
108
109 // hand/palm (tangan/telapak)
110 glPushMatrix();
111 glTranslatef(0.6, 0.0, 0.0);
112 glScalef(1.2, 0.2, 0.8);
113 DrawWireCube();
114 glPopMatrix();
115

```

`glRotatef((GLfloat)wrist, 1.0, 0.0, 0.0);` memutar pergelangan tangan di sekitar sumbu X, sehingga tangan bisa menekuk ke atas atau ke bawah.

Selain pergelangan tangan, jari-jari juga bisa bergerak dengan cara yang sama. Misalnya, jempol (`thumbBase`, `thumbMid`, `thumbTip`) diputar menggunakan transformasi:

```

116 // thumb (jempol) - 3 segments
117 glPushMatrix();
118 glTranslatef(1.1, 0.0, 0.4);
119 glRotatef((GLfloat)thumbBase, 0.0, 0.0, 1.0);
120 DrawFingerSegment(0.4);
121
122 glTranslatef(0.4, 0.0, 0.0);
123 glRotatef((GLfloat)thumbMid, 0.0, 0.0, 1.0);
124 DrawFingerSegment(0.3);
125
126 glTranslatef(0.3, 0.0, 0.0);
127 glRotatef((GLfloat)thumbTip, 0.0, 0.0, 1.0);
128 DrawFingerSegment(0.2);
129 glPopMatrix();

```