

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Grafik Komputer
Kelas : 3IA24
Praktikum ke- : 3
Tanggal : 25 Oktober 2023
Materi : Transformasi pada OpenGL
NPM : 50421704
Nama : Juan Samuel Christopher
Ketua Asisten : Ayu Fara Paramitha
Nama Asisten :
Paraf Asisten :
Jumlah Lembar : 5



LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2023

LISTING

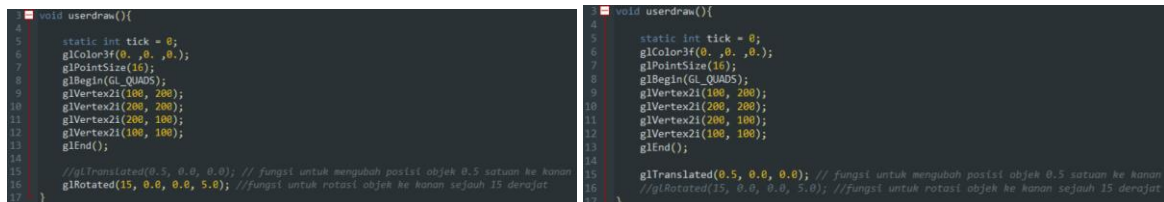
grafkompert3.cpp

```
1  #include <Gl/glut.h>
2
3  void userdraw(){
4
5      static int tick = 0;
6      glColor3f(0. ,0. ,0.);
7      glPointSize(16);
8      glBegin(GL_QUADS);
9      glVertex2i(100, 200);
10     glVertex2i(200, 200);
11     glVertex2i(200, 100);
12     glVertex2i(100, 100);
13     glEnd();
14
15     glTranslated(0.5, 0.0, 0.0);
16     //glRotated(15, 0.0, 0.0, 5.0);
17 }
18
19 void display(void){
20     glClear(GL_COLOR_BUFFER_BIT);
21     userdraw();
22     glutSwapBuffers();
23 }
24
25 int main(int argc, char** argv) {
26     glutInit(&argc, argv);
27     glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
28     glutInitWindowPosition (-100, 100);
29     glutInitWindowSize (640, 480);
30     glutCreateWindow("JuanSamuelChristopher-50421704");
31     glClearColor(1.0, 1.0, 1.0, 0.0);
32     gluOrtho2D (0., 640., -240., 500.);
33
34     glutIdleFunc(display);
35     glutDisplayFunc(display);
36     glutMainLoop();
37
38     return 0;
39 }
```

LOGIKA

```
1 #include <Gl/glut.h>
```

Pada pertemuan ke 2 ini kita akan belajar cara menggambar Gambar Primitice pada OpenGL. Kode program ini diawali dengan memasukkan library GLUT (OpenGL Utility Toolkit) dengan `#include <Gl/glut.h>` yang akan digunakan untuk membuat console application menggunakan OpenGL

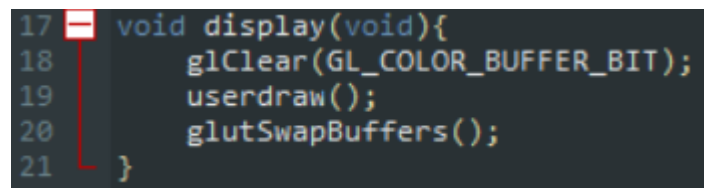


```
void userdraw(){
4
5     static int tick = 0;
6     glColor3f(0. ,0. ,0.);
7     glPointSize(16);
8     glBegin(GL_QUADS);
9     glVertex2i(100, 200);
10    glVertex2i(200, 200);
11    glVertex2i(200, 100);
12    glVertex2i(100, 100);
13    glEnd();
14
15    //glTranslated(0.5, 0.0, 0.0); // fungsi untuk mengubah posisi objek 0.5 satuan ke kanan
16    glRotated(15, 0.0, 0.0, 5.0); //fungsi untuk rotasi objek ke kanan sejauh 15 derajat
17 }
```

```
void userdraw(){
4
5     static int tick = 0;
6     glColor3f(0. ,0. ,0.);
7     glPointSize(16);
8     glBegin(GL_QUADS);
9     glVertex2i(100, 200);
10    glVertex2i(200, 200);
11    glVertex2i(200, 100);
12    glVertex2i(100, 100);
13    glEnd();
14
15    glTranslated(0.5, 0.0, 0.0); // fungsi untuk mengubah posisi objek 0.5 satuan ke kanan
16    glRotated(15, 0.0, 0.0, 5.0); //fungsi untuk rotasi objek ke kanan sejauh 15 derajat
17 }
```

Pada blok program diatas terdapat fungsi **void** yang bernama **Userdraw** yang akan menggambar objek persegi. Lalu ada **static int tick = 0**, adalah deklarasi variable bernama **tick** yang digunakan untuk menghitung waktu, lalu **glColor3f** adalah fungsi yang mengatur warna gambar dalam mode RGB dimana disini R=0, G=0,B=0, maka warna diatur ke warna hitam. Kemudian **glPointSize(16)**; adalah fungsi yang mengatur ukuran titik yang akan digunakan pada gambar disini diset ukuran 16.

Lanjut lagi ke **glBegin(GL_QUADS)**; adalah fungsi yang menandakan permulaan gambar yang berarti saya akan menggambar objek persegi empat yang ditandai oleh adanya **GL_QUADS**. Kemudian pada line 9 – 12 ada fungsi **glVertex2f** adalah fungsi yang menggambar titik , yang berarti ada 4 titik dimana masing masing titik mempunyai koordinat x dan y yang diatur sedemikian rupa seperti blok program diatas yang akan menggambar persegi yang diinginkan. Dan pada line 15 & 16 ada 2 fungsi yang masing masing harus di running satu per satu , Digambar kiri ada **glRotated** yang berguna untuk merotasi gambar ke arah kanan 15 derajat, dan di gambar kanan ada **glTranslated** yang akan menggeser objek 0.5 satuan ke kanan. Lalu terakhir ada **glEnd()**; yang menandakan akhir dari gambar yang berarti seluruh strip persegi empat telah Digambar.



```
17 void display(void){
18     glClear(GL_COLOR_BUFFER_BIT);
19     userdraw();
20     glutSwapBuffers();
21 }
```

Selanjutnya ada **void display (void); {** adalah deklarasi fungsi bernama **display** yang akan dipanggil untuk menampilkan objek/gambar pada layar. Lalu **glClear(GL_COLOR_BUFFER_BIT)** adalah fungsi yang akan membersihkan layar dan menghapus gambar sebelumnya. Dan **userdraw()** yang berfungsi untuk memanggil **void userdraw()** untuk menggambar objek persegi yang kita atur tadi. Kemudian **glutSwapBuffers()**; adalah fungsi yang bertanggungjawab untuk menukar dua buffer gambar, ini diperlukan karena program ini menggunakan mode double buffer.

```

25 int main(int argc, char** argv) {
26     glutInit(&argc, argv);
27     glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
28     glutInitWindowPosition (-100, 100);
29     glutInitWindowSize (640, 480);
30     glutCreateWindow("JuanSamuelChristopher-50421704");
31     glClearColor(1.0, 1.0, 1.0, 0.0);
32     gluOrtho2D (0., 640., -240., 500.);
33
34     glutIdleFunc(display);
35     glutDisplayFunc(display);
36     glutMainLoop();
37
38     return 0;
39 }

```

Pada blok program terakhir **int main(int argc, char** argv) {** Ini adalah fungsi main(), yang merupakan titik masuk utama program. **glutInit(&argc, argv);** Fungsi ini menginisialisasi GLUT dengan argumen baris perintah yang diteruskan dari command line. **glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);** Fungsi ini mengatur mode tampilan GLUT dengan double buffer dan mode warna RGB.

Lanjut ada **glutInitWindowPosition (100, 100);** Fungsi ini mengatur posisi jendela aplikasi. **glutInitWindowSize (640, 480);** Fungsi ini mengatur ukuran jendela aplikasi. **glutCreateWindow("JuanSamuelChristopher - 50421704");** Fungsi ini membuat jendela dengan judul yang diberikan. **glClearColor** adalah fungsi untuk warna background yang berarti disini (1.0, 1.0, 1.0, 0.0) adalah warna putih, dan **gluOrtho2D** ini mengatur parameter proyeksi ortografik untuk jendela.

Terakhir ada **glutIdleFunc(display);** Fungsi ini menentukan fungsi display() sebagai fungsi yang akan dipanggil ketika aplikasi berada dalam mode idle. **glutDisplayFunc(display);** Fungsi ini menentukan fungsi display() sebagai fungsi yang akan digunakan untuk menampilkan objek pada layar. **glutMainLoop();** Fungsi ini memulai loop utama GLUT, yang akan menangani interaksi pengguna dan menampilkan objek pada layar. **Return 0;** Ini adalah nilai kembali dari fungsi main() yang menunjukkan bahwa program selesai dengan sukses.

OUTPUT

- **Translated**



- **Rotated**

