GRAFIKA KOMPUTER

PRAKTIKUM 5

Disusun Oleh:

Sultan Alif Novana Putra 1805055

PROGRAM STUDI D4 REKAYASA PERANGKAT LUNAK

JURUSAN TEKNIK INFORMATIKA

POLITEKNIK NEGERI INDRAMAYU

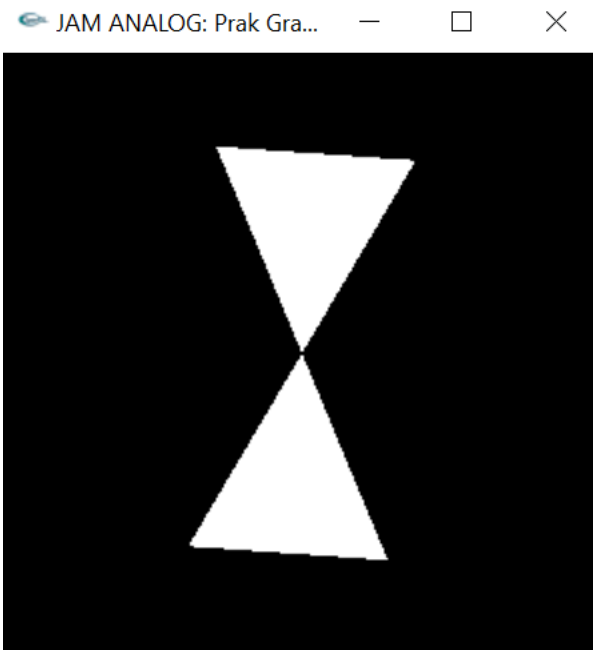
2020/2021

1. Membuat sebuah baling-baling yang bisa perputar secara clockwise kemudian unclockwise yang pusatnya berada pada pusat koordinat

```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
int x=0;
int zputer=0;
int zbalik=180;
void Timer(int value){
if (zputer <= 360){
x = 1;
zputer += 1;
}
if (zputer>360){
x = -1;
zbalik -= 1;
}
if (zbalik<0){
x = 1;
zputer = 0;
zbalik = 360;
}
glutPostRedisplay();
glutTimerFunc(5, Timer, 0);
}
void Draw(){
glBegin(GL_TRIANGLES);
glClear(GL_COLOR_BUFFER_BIT);
glVertex2d(0, 0);
glVertex2d(-100, 200);
glVertex2d(100, 200);
glVertex2d(0, 0);
glVertex2d(-100, -200);
glVertex2d(100, -200);
glEnd();
}
void display()
{
glClear(GL_COLOR_BUFFER_BIT);
```

```
glPopMatrix();
glRotatef(x, 0., 0., 1.);
Draw();
glPushMatrix();
glFlush();
}
void main(int argc, char **argv){
glutInit(&argc, argv);
glutInitWindowPosition(100, 100);
glutInitWindowSize(300, 300);
glutCreateWindow("JAM ANALOG: Prak Grafkom");
gluOrtho2D(-300., 300., -300., 300.);
glutDisplayFunc(display);
glutTimerFunc(60, Timer, 0);
glutMainLoop();
}
```

Animasi 2D menggunakan OpenGL dapat dilakukan dengan melakukan perpindahan objek menggunakan glTranslatef, juga dapat dilakukan dengan melakukan perputaran atau pergerakan objek yang berporos pada sumbu-sumbu koordinat dari sumbu pembentuk objek tersebut menggunakan glRotatef. Kemudian waktu yang digunakan dalam pergerakan animasi juga dapat diatur sesuai keinginan dengan menggunakan glutTimerFunc. Ouputnya :



```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
int c, d;
void triAngles(){
glBegin(GL_TRIANGLES);
glVertex2i(280, 0);
glVertex2i(250, 60);
glVertex2i(220, 0);
glEnd();
}
void mySpecialKeyboard(int key, int x, int y){
switch (key){
case
GLUT_KEY_LEFT: c += -4;
d += 0;
break;
case
GLUT_KEY_RIGHT: c += 4;
d += 0;
break;
case
GLUT_KEY_UP: c += 0; d += 4;
break;
case
GLUT_KEY_DOWN: c += 0;
d += -4;
break;
}
}
void timer(int value){
glutPostRedisplay();
glutTimerFunc(50, timer, 0);
}
void renderScene(void){
glClear(GL_COLOR_BUFFER_BIT);
glPushMatrix();
glTranslatef(c, d, 0);
triAngles();
glPopMatrix();
```

```
glFlush();
}
void main(int argc, char **argv){
glutInit(&argc, argv);
glutInitWindowPosition(100, 100);
glutInitWindowSize(640, 480);
glutCreateWindow("uji keyfunc");
gluOrtho2D(-320., 320., -240., 240.);
glutTimerFunc(50, timer, 0);
glutDisplayFunc(renderScene);
glutSpecialFunc(mySpecialKeyboard);
glutMainLoop();
}
```

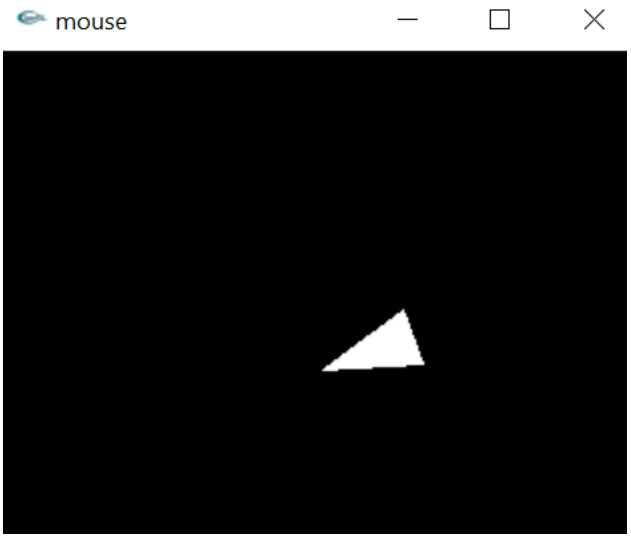
Uji KeyFunc Ouputnya:

CONTOH PROGAM PENERAPAN MOUSE PADA GLUT

```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
void Triangles(){
glBegin(GL_TRIANGLES);
glVertex2i(0, 0);
glVertex2i(100, -30);
glVertex2i(100, 30);
glEnd();
}
void mouseku(int button, int state, int x, int y){
if (button == GLUT_LEFT_BUTTON){
printf("tombol KIRI (%d,%d)\n", x, y);
}
else if (button == GLUT_MIDDLE_BUTTON){
printf("tombol TENGAH (%d,%d)\n", x, y);
}
else{
printf("tombol KANAN (%d,%d)\n", x, y);
}
if (state == GLUT_DOWN){
printf("tombol DITEKAN\n");
}
else{
printf("tombol DILEPAS\n");
}
} void display(){
glClear(GL_COLOR_BUFFER_BIT); glRotatef(10, 0., 0.,
1.);
Triangles();
glFlush();
}
void Timer(int value){
glutPostRedisplay();
glutTimerFunc(30, Timer, 0);
}
void main(int argc, char **argv){
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_DEPTH | GLUT_SINGLE |
```

```
GLUT_RGBA);
glutInitWindowPosition(100, 100);
glutInitWindowSize(320, 320);
glutCreateWindow("mouse");
glutDisplayFunc(display);
gluOrtho2D(-320.0, 320.0, -320.0, 320.0);
glutTimerFunc(10, Timer, 0);
glutMouseFunc(mouseku);
glutMainLoop();
}
```

Ouputnya :



Interaksi Drag

```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
void Triangles(){
glBegin(GL_TRIANGLES);
glVertex2i(0, 0);
glVertex2i(100, -30);
glVertex2i(100, 30);
glEnd();
}
void mouseku(int button, int state, int x, int y){
if (button == GLUT_LEFT_BUTTON){
printf("tombol KIRI (%d,%d)\n", x, y);
}
else if (button == GLUT_MIDDLE_BUTTON){
printf("tombol TENGAH (%d, %d)\n", x, y);
}
else{
printf("tombol KANAN (%d,%d)\n", x, y);
}
if (state == GLUT_DOWN){
printf("tombol DITEKAN\n");
}
else{
printf("tombol DILEPAS\n");
}
}
void motionku(int x, int y){
printf("posisi printer mouse (%d, %d)\n", x, y);
}
void display(){
glClear(GL_COLOR_BUFFER_BIT); glRotatef(89,
0., 0., 1.);
Triangles();
glFlush();
}
void Timer(int value){
glutPostRedisplay();
glutTimerFunc(10, Timer, 0);
```

```
}
void main(int argc, char **argv){
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_DEPTH | GLUT_SINGLE |
GLUT_RGBA);
glutInitWindowPosition(100, 100);
glutInitWindowSize(320, 320);
glutCreateWindow("GLUT");
glutDisplayFunc(display);
gluOrtho2D(-320.0, 320.0, -320.0, 320.0);
glutTimerFunc(10, Timer, 0);
glutMouseFunc(mouseku);
glutMotionFunc(motionku);
glutMainLoop();
}
```

Output :

Contoh Implementasi :

```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
float x = 1, y = 1, z = 0;
int w = 480, h = 480, flag = 0, flg;
void drawQuad() {
glBegin(GL_POLYGON);
glColor3f(0, 0, 1);
glVertex2f(-20., -20.);
glVertex2f(20., -20.);
glVertex2f(20., 20.);
glVertex2f(-20., 20.);
glEnd();
}
void mouse(int button, int state, int xmouse,
int ymouse){
if (flg == 0)
{
if (state == GLUT_DOWN)
{
if (button == GLUT_LEFT_BUTTON)
{
flag++; if (flag == 2)
{
flg = 3; x = 3; y = 3;
printf("%d", flg);
}
}
}
}
if (flg == 3)
{
if (state == GLUT_DOWN)
{
if (button == GLUT_LEFT_BUTTON)
{
flag--; if (flag == 0)
{
x = 1; y = 1; flg =
```

```
0;
}
}
}
}
}
void renderScene(void){
glClear(GL_COLOR_BUFFER_BIT);
glClearColor(1, 1, 1, 1);
glPushMatrix();
glScalef(x, y, z);
drawQuad();
glPopMatrix();
glFlush();
}
void timer(int value){
glutPostRedisplay();
glutTimerFunc(50, timer, 0);
}
void main(int argc, char **argv){
glutInit(&argc, argv);
glutInitWindowPosition(100, 100);
glutInitWindowSize(w, h);
glutCreateWindow("Interaksi Mouse");
gluOrtho2D(-w / 2, w / 2, -h / 2, h / 2);
glutDisplayFunc(renderScene);
glutMouseFunc(mouse);
glutTimerFunc(1, timer, 0);
glutMainLoop();
 }
```

Outputnya :