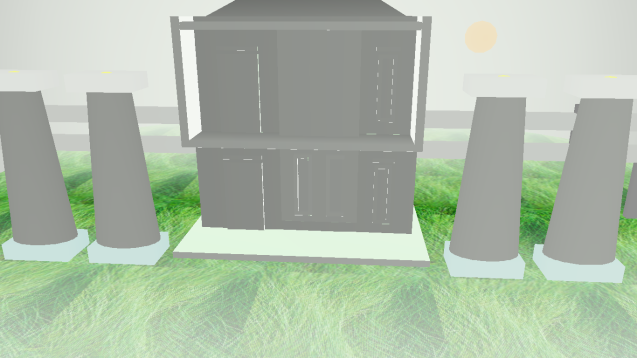
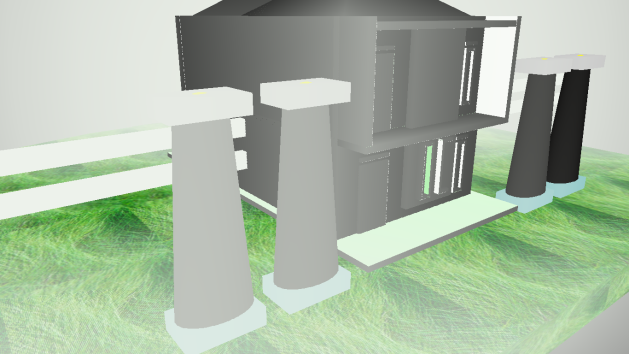
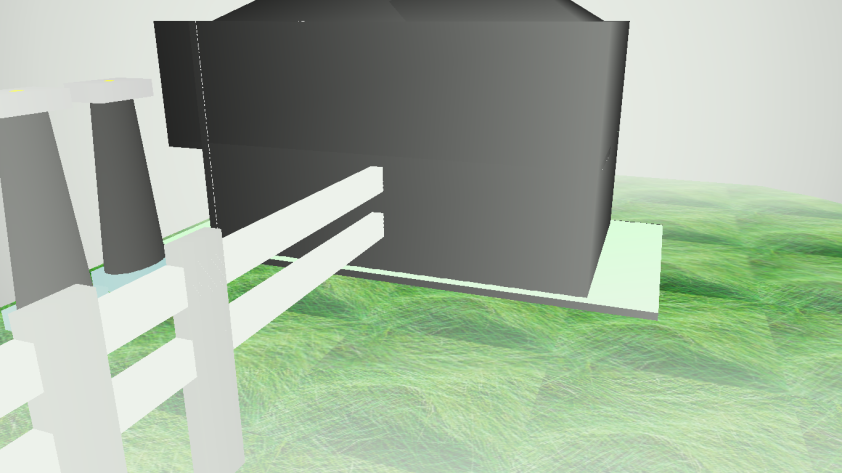
// tampilan depan



// Tampilan samping



// Tampilan Belakang

**//Deskripsi**

Pada tugas komputer grafika ini dibangun sebuah rumah dengan bangunan 2 lantai. Pagar dan 4 pilar sebagai properti pendukung. Matahari dengan pergerakan statis beserta permainan texture pada rumput dan langit serta kondisi alam menggunakan setuhan kabut dan sedikit teknik pencahayaan .

**// Listing program**

/\*

Kelompok :

- Robi JS 10107033

- Hari M 10107040

- Eka Nugraha 10107041

- Ridky B 10107862

\*/

#include <cstdlib>

#include <iostream>

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

#include <windows.h>

#include <string.h>

#include <conio2.h>

#include <GL/glut.h>

#include <GL/glext.h>

#include "island.h"

char title[] = "Perbaikan komgraf",

cmdvolvar[]="";

unsigned int windowWidth= 640,

windowHeight= 480,

windowPosX= 50,

windowPosY= 50,

vol= 300;

bool fullScreenMode= true, kedepan=false;

float sudutputar= 0.0;

unsigned int texture\_id, langit1, langit2, langit3, alasrum, pasir;

// meload textur

int load\_texture(char \*file\_name, unsigned int width, unsigned int height){

GLubyte \*imgbitmap;

FILE \*file;

unsigned short int depth=3;

if ((file = fopen(file\_name, "rb"))==NULL){

printf("File tidak ada: %s!\n",file\_name);

getch(); exit(1);

}

imgbitmap = (GLubyte \*) malloc (width \* height \* depth \* (sizeof(GLubyte)));

if (imgbitmap == NULL){

printf("load texture gagal!\n");

fclose(file);

getch(); exit(1);

}

fread(imgbitmap, width \* height \* depth, 1, file);

fclose(file);

glGenTextures(1, &texture\_id);

glBindTexture(GL\_TEXTURE\_2D, texture\_id);

glPixelStorei(GL\_UNPACK\_ALIGNMENT,1);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_MAG\_FILTER, GL\_LINEAR);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_MIN\_FILTER, GL\_LINEAR\_MIPMAP\_NEAREST);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_S, GL\_REPEAT);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_T, GL\_REPEAT);

glTexEnvi(GL\_TEXTURE\_ENV, GL\_TEXTURE\_ENV\_MODE, GL\_REPLACE);

gluBuild2DMipmaps(GL\_TEXTURE\_2D, GL\_RGB8, width, height, GL\_BGR, GL\_UNSIGNED\_BYTE, imgbitmap);

glTexImage2D(GL\_TEXTURE\_2D, 0, GL\_RGB8, width, height, 0, GL\_BGR, GL\_UNSIGNED\_BYTE, imgbitmap);

free(imgbitmap);

return texture\_id;

}

// Load semua gambar

void load\_image(){

langit1 = load\_texture("bmp/langit0.bmp", 356, 256);

alasrum = load\_texture("bmp/alasrum.bmp", 256, 256);

printf("Inisialisasi textur sedang berlangsung.\n");

}

// ini adalah bagian Inisialisasi

void init() {

glClearColor(1.0f, 1.0f, 1.0f, 1.0f);

glClearDepth(100.0f);

glEnable(GL\_DEPTH\_TEST);

glDepthFunc(GL\_LEQUAL);

glHint(GL\_PERSPECTIVE\_CORRECTION\_HINT, GL\_NICEST);

glShadeModel(GL\_SMOOTH);

glEnable(GL\_ALPHA\_TEST);

glAlphaFunc(GL\_GREATER, 0);

load\_image();

printf("\nJalankan aplikasi.\n");

}

// ini adalah bagian load untuk texture rumput

void draw\_rumput(){

float xleft=-13, y=-1.0, zfar=8.0, xright=11, znear=-20.0, xl, zf, incre=4.0;

zf=zfar;

while (zfar>=znear){

xl=xleft;

while (xl<=xright){

//------------------------gambar design alas

glEnable(GL\_TEXTURE\_2D);

glBindTexture(GL\_TEXTURE\_2D, alasrum);

if(xl==xleft){

glPushMatrix();

glBegin(GL\_QUADS); // kiri

glTexCoord2f(1.0f, 1.0f); glVertex3f(xl, y, zfar); // Kanan Atas

glTexCoord2f(1.0f, 0.0f); glVertex3f(xl, y-(incre/4), zfar); // Kiri Atas

glTexCoord2f(0.0f, 0.0f); glVertex3f(xl, y-(incre/4), zfar-incre); // bawah Kiri

glTexCoord2f(0.0f, 1.0f); glVertex3f(xl, y, zfar-incre); // bawah atas

glEnd();

glPopMatrix();

}

if(zfar==zf){

glPushMatrix();

glBegin(GL\_QUADS); // atas

glTexCoord2f(1.0f, 0.0f); glVertex3f(xl, y, zfar);

glTexCoord2f(1.0f, 1.0f); glVertex3f(xl+incre, y, zfar);

glTexCoord2f(0.0f, 1.0f); glVertex3f(xl+incre, y-(incre/4), zfar);

glTexCoord2f(0.0f, 0.0f); glVertex3f(xl, y-(incre/4), zfar);

glEnd();

glPopMatrix();

}

if(xl==xright){

glPushMatrix();

glBegin(GL\_QUADS); // kanan

glTexCoord2f(1.0f, 1.0f); glVertex3f(xl+incre, y-(incre/4), zfar);

glTexCoord2f(1.0f, 0.0f); glVertex3f(xl+incre, y, zfar);

glTexCoord2f(0.0f, 0.0f); glVertex3f(xl+incre, y, zfar-incre);

glTexCoord2f(0.0f, 1.0f); glVertex3f(xl+incre, y-(incre/4), zfar-incre);

glEnd();

glPopMatrix();

}

if(zfar==znear){

glPushMatrix();

glBegin(GL\_QUADS); // bawah

glTexCoord2f(1.0f, 0.0f); glVertex3f(xl, y, zfar-incre);

glTexCoord2f(1.0f, 1.0f); glVertex3f(xl+incre, y, zfar-incre);

glTexCoord2f(0.0f, 1.0f); glVertex3f(xl+incre, y-(incre/4), zfar-incre);

glTexCoord2f(0.0f, 0.0f); glVertex3f(xl, y-(incre/4), zfar-incre);

glEnd();

glPopMatrix();

}

glDisable(GL\_TEXTURE\_2D);

//bagian

glPushMatrix();

glEnable(GL\_TEXTURE\_2D);

glBindTexture(GL\_TEXTURE\_2D, alasrum);

glBegin(GL\_QUADS);

glTexCoord2f(1.0f, 0.0f); glVertex3f(xl, y, zfar);

glTexCoord2f(1.0f, 1.0f); glVertex3f(xl+incre, y, zfar);

glTexCoord2f(0.0f, 1.0f); glVertex3f(xl+incre, y, zfar-incre);

glTexCoord2f(0.0f, 0.0f); glVertex3f(xl, y, zfar-incre);

glEnd();

glDisable(GL\_TEXTURE\_2D);

glPopMatrix();

xl+=incre;

}

zfar-=incre;

}

}

// langit

void draw\_langit(){

glPushMatrix();

glColor3ub(125, 125, 125);

glEnable(GL\_TEXTURE\_2D);

glBindTexture(GL\_TEXTURE\_2D, langit1);

GLUquadricObj\* q = gluNewQuadric();

gluQuadricDrawStyle(q, GLU\_FILL);

gluQuadricNormals(q, GLU\_SMOOTH);

gluQuadricTexture(q, GL\_TRUE);

glRotatef(90.0, 1.0, 0.0, 0.0);

glScalef(16.0, 16.0, 16.0);

gluSphere(q, 1.0, 30, 30);

glDisable(GL\_TEXTURE\_2D);

glPopMatrix();

}

// Draw pulau kotak

void draw\_rumah(){

glPushMatrix();

GLfloat pos[4]={5.0,18.0,-15.0,70.0};

glLightfv(GL\_LIGHT0, GL\_POSITION, pos);

glEnable(GL\_LIGHTING);

glEnable(GL\_LIGHT0);

//atap

glPushMatrix();

glScaled(0.8, 1.0, 0.8);

glTranslatef(0.0, 4.85, -1.9);

glRotated(45, 0, 1, 0);

glRotated(-90, 1, 0, 0);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3d(0.803921568627451, 0.5215686274509804, 0.2470588235294118);

glutSolidCone(4.2, 1.5, 4, 1);

glPopMatrix();

//atap

glPushMatrix();

glScaled(0.8, 1.0, 0.8);

glTranslatef(0.0, 4.85, 2.1);

glRotated(45, 0, 1, 0);

glRotated(-90, 1, 0, 0);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3d(0.803921568627451, 0.5215686274509804, 0.2470588235294118);

glutSolidCone(4.2, 1.5, 4, 1);

glPopMatrix();

//lantai 1

glPushMatrix();

glScaled(1.115, 0.03, 2.2);

glTranslatef(0.0, 0, 0.0);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//lantai 2 depan

glPushMatrix();

glScaled(1.015, 0.03, 1.2);

glTranslatef(0.0,80, 1.7);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//lantai 2 belakang

glPushMatrix();

glScaled(0.5, 0.03, 0.8);

glTranslatef(2.5,80, -2.8);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//lantai 3

glPushMatrix();

glScaled(1.015, 0.03, 1.8);

glTranslatef(0.0,160, 0.3);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Kiri Bawah

glPushMatrix();

glScaled(0.035, 0.5, 1.6);

glTranslatef(-70.0, 2.45, 0.0);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Kanan Bawah

glPushMatrix();

glScaled(0.035, 0.5, 1.6);

glTranslatef(70.0, 2.45, 0.0);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Kiri Atas

glPushMatrix();

glScaled(0.035, 0.5, 1.8);

glTranslatef(-70.0, 7.45, 0.3);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Kanan Atas

glPushMatrix();

glScaled(0.035, 0.5, 1.8);

glTranslatef(70.0, 7.45, 0.3);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Belakang bawah

glPushMatrix();

//glScaled(0.035, 0.5, 0.8);

glScaled(1.015, 0.5, 0.07);

glTranslatef(0, 2.45,-58);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Belakang atas

glPushMatrix();

//glScaled(0.035, 0.5, 0.8);

glScaled(1.015, 0.5, 0.07);

glTranslatef(0, 7.45,-58);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Depan bawah

glPushMatrix();

glScaled(1.015, 0.5, 0.035);

glTranslatef(0, 2.45,116);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//Dinding Depan atas

glPushMatrix();

glScaled(1.015, 0.5, 0.035);

glTranslatef(0, 7.45,116);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.4613, 0.4627, 0.4174);

glutSolidCube(5.0);

glPopMatrix();

//list hitam atas

glPushMatrix();

glScaled(0.35, 0.5, 0.035);

glTranslatef(1, 7.2,124);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.1412, 0.1389, 0.1356);

glutSolidCube(5.0);

glPopMatrix();

//list hitam atas

glPushMatrix();

glScaled(0.35, 0.43, 0.035);

glTranslatef(1, 3.5,124);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.1412, 0.1389, 0.1356);

glutSolidCube(5.0);

glPopMatrix();

//pintu atas

glPushMatrix();

glScaled(0.18, 0.35, 0.035);

glTranslatef(-8, 9.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.0980, 0.0608, 0.0077);

glutSolidCube(5.0);

glPopMatrix();

//pintu bawah

glPushMatrix();

glScaled(0.18, 0.35, 0.035);

glTranslatef(-8, 2.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.0980, 0.0608, 0.0077);

glutSolidCube(5.0);

glPopMatrix();

//alis

glPushMatrix();

glScaled(0.18, 0.017, 0.035);

glTranslatef(-8, 110.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0, 0, 0);

glutSolidCube(5.0);

glPopMatrix();

//alis atas kiri

glPushMatrix();

glScaled(0.18, 0.017, 0.035);

glTranslatef(-8, 254,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

//glColor3f(0.3402, 0.3412, 0.3117);

glColor3f(0, 0, 0);

glutSolidCube(5.0);

glPopMatrix();

//alis atas kanan

glPushMatrix();

glScaled(0.10, 0.017, 0.035);

glTranslatef(18, 254,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0, 0, 0);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela atas

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(22.5, 245,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela bawah

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(22.5, 165,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kiri

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(96.6, 12.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kanan

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(115.1, 12.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//jendela bawah (kanan Bawah)

//alis atas kanan (kanan Bawah)

glPushMatrix();

glScaled(0.10, 0.017, 0.035);

glTranslatef(18, 110.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0, 0, 0);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela atas (kanan Bawah)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(22.5, 101.5,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela bawah (kanan Bawah)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(22.5, 22.0,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kiri (kanan Bawah)

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(96.6, 3.8,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kanan (kanan Bawah)

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(115.1, 3.8,118);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(0.3402, 0.3412, 0.3117);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela atas (tengah1)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(0, 119.5,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela bawah (tengah1)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(0, 40.0,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kiri (tengah1)

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(-9.6, 4.8,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kanan (tengah1)

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(9.5, 4.8,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela atas (tengah2)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(9, 119.5,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(204, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela bawah (tengah2)

glPushMatrix();

glScaled(0.08, 0.017, 0.035);

glTranslatef(9, 40.0,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

//alis jedela kiri (tengah2)

glPushMatrix();

glScaled(0.017,0.28, 0.035);

glTranslatef(33, 4.8,128);

glColorMaterial(GL\_FRONT\_AND\_BACK, GL\_AMBIENT\_AND\_DIFFUSE);

glColor3f(1.0000, 0.5252, 0.0157);

glutSolidCube(5.0);

glPopMatrix();

glDisable(GL\_LIGHT0);

glDisable(GL\_LIGHTING);

glPopMatrix();

}

void draw\_pilar1(){

GLUquadricObj \*pObj; //

pObj =gluNewQuadric();

gluQuadricNormals(pObj, GLU\_SMOOTH);

glPushMatrix();

glTranslatef(4.0, 3.7, 5.0);

glScalef(0.3, 0.19, 0.19);

//bagian atas

glPushMatrix();

// tentukan posisi baru dan arah dalam koordinat ptr

glColor3ub(255,255,0);

gluSphere(pObj, 1.0f, 26, 13);

glPopMatrix();

//alas atas cube

glPushMatrix();

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(200, 200, 200);

glutSolidCube(4.5);

glPopMatrix();

//bagian silinder

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glRotatef(-90, 1.0f, 0.0f, 0.0f);

glColor3ub(1, 0, 0);

gluCylinder(pObj, 2.50f, 1.50f, 20.0f, 26, 1);

glPopMatrix();

//bawahnya

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(400, 200, 200);

glutSolidCube(5);

glPopMatrix();

glPopMatrix();

}

void draw\_pilar2(){

GLUquadricObj \*pObj; //

pObj =gluNewQuadric();

gluQuadricNormals(pObj, GLU\_SMOOTH);

glPushMatrix();

glTranslatef(-4.0, 3.7, 5.0);

glScalef(0.3, 0.19, 0.19);

//bagian atas

glPushMatrix();

// tentukan posisi baru dan arah dalam koordinat ptr

glColor3ub(255,255,0);

gluSphere(pObj, 1.0f, 26, 13);

glPopMatrix();

//alas atas cube

glPushMatrix();

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(200, 200, 200);

glutSolidCube(4.5);

glPopMatrix();

//bagian silinder

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glRotatef(-90, 1.0f, 0.0f, 0.0f);

glColor3ub(1, 0, 0);

gluCylinder(pObj, 2.50f, 1.50f, 20.0f, 26, 1);

glPopMatrix();

//bawahnya

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(400, 200, 200);

glutSolidCube(5);

glPopMatrix();

glPopMatrix();

}

void draw\_pilar3(){

GLUquadricObj \*pObj; //

pObj =gluNewQuadric();

gluQuadricNormals(pObj, GLU\_SMOOTH);

glPushMatrix();

glTranslatef(-6.0, 3.7, 5.0);

glScalef(0.3, 0.19, 0.19);

//bagian atas

glPushMatrix();

// tentukan posisi baru dan arah dalam koordinat ptr

glColor3ub(255,255,0);

gluSphere(pObj, 1.0f, 26, 13);

glPopMatrix();

//alas atas cube

glPushMatrix();

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(200, 200, 200);

glutSolidCube(4.5);

glPopMatrix();

//bagian silinder

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glRotatef(-90, 1.0f, 0.0f, 0.0f);

glColor3ub(1, 0, 0);

gluCylinder(pObj, 2.50f, 1.50f, 20.0f, 26, 1);

glPopMatrix();

//bawahnya

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(400, 200, 200);

glutSolidCube(5);

glPopMatrix();

glPopMatrix();

}

void draw\_pilar4(){

GLUquadricObj \*pObj; //

pObj =gluNewQuadric();

gluQuadricNormals(pObj, GLU\_SMOOTH);

glPushMatrix();

glTranslatef(6.0, 3.7, 5.0);

glScalef(0.3, 0.19, 0.19);

//bagian atas

glPushMatrix();

// tentukan posisi baru dan arah dalam koordinat ptr

glColor3ub(255,255,0);

gluSphere(pObj, 1.0f, 26, 13);

glPopMatrix();

//alas atas cube

glPushMatrix();

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(200, 200, 200);

glutSolidCube(4.5);

glPopMatrix();

//bagian silinder

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glRotatef(-90, 1.0f, 0.0f, 0.0f);

glColor3ub(1, 0, 0);

gluCylinder(pObj, 2.50f, 1.50f, 20.0f, 26, 1);

glPopMatrix();

//bawahnya

glPushMatrix();

glTranslatef(0.0f, -20.0f, 0.0f);

glScalef(1.0f, 0.4f, 1.0f);

glColor3ub(400, 200, 200);

glutSolidCube(5);

glPopMatrix();

glPopMatrix();

}

void draw\_matahari(){

glPushMatrix();

//glTranslatef(-3.0, -0.8, 5.0);

glTranslatef(7.0, 4.0, -10.0);

glScalef(0.15, 0.15, 0.15);

glColor3ub(255, 147, 0);

glutSolidSphere(5.0f, 13, 26);

glPopMatrix();

}

// bagan untuk menampilkan pagar

void draw\_pagar()

{

//Atas

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(12.0f, 2.0f, 0.0f);

glScaled(297.0, 1.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

//Bawah

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(12.0f, 1.05f, 0.0f);

glScaled(297.0, 1.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

//lurus

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(8.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(68.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(94.8f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(-52.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(-102.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(152.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

glPushMatrix();

glColor4f(0.5f, 0.5f, 0.5f, 1.0f);

glTranslatef(10.0f, 0.0f, 0.0f);

glScaled(1.5, 10.0 , 0.5);

glutSolidCube(0.5f);

glPopMatrix();

}

// ini adalah bagian untuk merubah warna untuk pilar

void display(void) {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT | GL\_STENCIL\_BUFFER\_BIT);

GLfloat pos0[4] ={1.0, 1.0, 1.0, 0.0};

GLfloat ambient[4] = { 0.0,0.0, 0.0, 0.0 };

GLfloat diffuse[4] = { 1.0, 2.0, 1.0, 1.0 };

GLfloat specular[4] = { 3.0, 4.0,6.0, 1.0 };

GLfloat shininess[1] = { 60.0 };

glLightfv(GL\_LIGHT0, GL\_AMBIENT, ambient);

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, diffuse);

glLightfv(GL\_LIGHT0, GL\_POSITION, pos0);

glEnable(GL\_BLEND);

glBlendFunc(GL\_ONE, GL\_ONE);

GLfloat kabut[4]={0.93,0.95,0.92,0.88};

glFogfv(GL\_FOG\_COLOR, kabut); // kabut

glFogf(GL\_FOG\_DENSITY, 0.2f);

glFogi(GL\_FOG\_MODE, GL\_EXP);

glEnable(GL\_FOG);

glDisable(GL\_BLEND);

glLoadIdentity();

glRotatef(sudutputar, 0.0f, 0.1f, 0.0f); // sudut pandang camera

draw\_rumput();

draw\_langit();

draw\_rumah();

draw\_pilar1();

draw\_pilar2();

draw\_pilar3();

draw\_pilar4();

draw\_pagar();

draw\_matahari();

glutSwapBuffers();

sudutputar -= 0.1f;

if(sudutputar<-360) sudutputar=360;

}

// mengatur ukuran tampilan

void reshape(GLsizei width, GLsizei height) {

if (height == 0) height = 1;

glViewport(0, 0, width, height);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

//glOrtho(-10,10,-10,10,-20,20);

gluPerspective(40.0, (float)width / (float)height, 0.1, 100.0);

gluLookAt(0.0, 5.0, 15.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);

glMatrixMode(GL\_MODELVIEW);

glLoadIdentity();

}

// kontrol

void keyboard(unsigned char key, int x, int y) {

switch (key) {

case 27: exit(0); break; // ESC: keluar

default: break;

}

}

// ini adalah bagian untuk mengatur besar kecilnya secreen

void specialKey(int key, int x, int y) {

switch (key) {

case GLUT\_KEY\_F1: // F1: mengatur ukuran gambar window

fullScreenMode = !fullScreenMode;

if (fullScreenMode) {

windowPosX = glutGet(GLUT\_WINDOW\_X);

windowPosY = glutGet(GLUT\_WINDOW\_Y);

windowWidth = glutGet(GLUT\_WINDOW\_WIDTH);

windowHeight = glutGet(GLUT\_WINDOW\_HEIGHT);

glutFullScreen();

} else {

glutReshapeWindow(windowWidth, windowHeight);

glutPositionWindow(windowPosX, windowPosX);

}

break;

}

}

GLfloat xangle=0.0, yangle=0.0;

// Fungsi utama: GLUT berjalan sebagai aplikasi konsol

int main(int argc, char \*argv[]) {

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_RGBA | GLUT\_DOUBLE | GLUT\_DEPTH);

glutInitWindowSize(windowWidth, windowHeight);

glutInitWindowPosition(windowPosX, windowPosY);

glutCreateWindow(title);

glutFullScreen();

glutDisplayFunc(display);

glutReshapeFunc(reshape);

glutKeyboardFunc(keyboard);

glutSpecialFunc(specialKey);

glutIdleFunc(display);

init();

glutMainLoop();

return 0;

}