

Department of Computer Science and Engineering

POJECT SOURCE CODE

|  |  |  |
| --- | --- | --- |
| INFORMATIONS | | |
| COMPUTER GRAPHICS [A] | | |
| SL | **NAME** | **ID** |
| 1 | **TANVIR TANJUM SHOURAV** | **17-35860-3** |
| 2 | **ZISHAD HOSSAIN LIMON** | **18-36101-1** |
| 3 | **ASHFAQ AFZAL CHOWDHURY** | **17-35800-3** |
| 4 | **ANIKA TAHSIN TINA** | **17-35852-3** |

**///------------------\*\*\*\*------------------///**

**///TANVIR TANJUM SHOURAV (17-35860-3)**

**///ZISHAD HOSSAIN LIMON (18-36101-1)**

**///ANIKA TAHSIN TINA (17-35852-3)**

**///ASHFAQ AFZAL CHOWDHURY (17-35800-3)**

**///------------------\*\*\*\*------------------///**

**///GIT REPOSITORY:** [**https://github.com/tanvirtanjum/Dynamic-Sea-Beach-View**](https://github.com/tanvirtanjum/Dynamic-Sea-Beach-View)

**///CLONE FROM GITHUB REPOSITORY FOR EXPERIENCING SOUND EFFECTS**

#include <cstdio>

#include <windows.h>

#include <GL/gl.h>

#include <GL/glut.h>

#include <math.h>

#define PI 3.14159265358979323846

GLfloat i = 0.0f;

GLfloat position\_sun = 0.0f;

GLfloat position\_sun\_down = 0.0f;

GLfloat speed\_sun = 0.0175f;

GLfloat position\_moon = 0.0f;

GLfloat position\_moon\_down = 0.0f;

GLfloat speed\_moon = 0.0175f;

GLfloat position\_cloud = 0.0f;

GLfloat speed\_cloud = 0.02f;

GLfloat position\_balloon = 0.0f;

GLfloat speed\_balloon = 0.4f;

GLfloat position\_bird = 0.0f;

GLfloat speed\_bird = 0.03f;

GLfloat rain\_position1 = 0.0f;

GLfloat rain\_speed1 = 0.02f;

GLfloat rain\_position2 = 0.0f;

GLfloat rain\_speed2 = 0.01f;

GLfloat position\_ship = 0.0f;

GLfloat speed\_ship = 0.01f;

GLfloat position\_sea\_wave = 0.0f;

GLfloat speed\_sea\_wave = 0.05f;

boolean RainController = false;

GLfloat ship\_position = 0.0f;///ship

void load\_Start();

void load\_DayBack();

void Idle()

{

glutPostRedisplay();

}

void update\_sun(int value)

{

if(position\_sun <-1.0)

{

position\_sun = 1.0f;

position\_sun\_down = 1.0f;

}

position\_sun -= speed\_sun;

position\_sun\_down -= speed\_sun/3;

glutTimerFunc(135, update\_sun, 0);

}

void update\_ship(int value)

{

if(position\_ship <-1.0)

{

position\_ship = 1.0f;

}

position\_ship -= speed\_ship;

glutTimerFunc(100, update\_ship, 0);

}

void update\_sea\_wave(int value)

{

position\_sea\_wave -= 50;

if(position\_sea\_wave <= -80)

position\_sea\_wave = 0;

glutPostRedisplay();

glutTimerFunc(1500,update\_sea\_wave,0);

}

void update\_moon(int value)

{

if(position\_moon <-1.0)

{

position\_moon = 1.0f;

position\_moon\_down = 1.0f;

}

position\_moon -= speed\_moon;

position\_moon\_down -= speed\_moon/3;

glutTimerFunc(80, update\_moon, 0);

}

void update\_cloud(int value)

{

if(position\_cloud <-1.0)

{

position\_cloud = 1.0f;

}

position\_cloud -= speed\_sun;

glutTimerFunc(100, update\_cloud, 0);

}

void update\_balloon(int value)

{

if(position\_balloon <-1.0)

{

position\_balloon = 1.0f;

}

position\_balloon -= speed\_balloon;

glutTimerFunc(50, update\_balloon, 0);

}

void update\_bird(int value)

{

if(position\_bird <-1.0)

{

position\_bird = 1.0f;

}

position\_bird -= speed\_bird;

glutTimerFunc(100, update\_bird, 0);

}

void update\_rain(int value)

{

if(rain\_position1 <-1.0)

rain\_position1 = 0.2f;

rain\_position1 -= rain\_speed1;

if(rain\_position2 >1.0)

rain\_position2 = -1.0f;

rain\_position2 += rain\_speed2;

glutTimerFunc(30, update\_rain, 0);

}

void load\_Restart(int x)

{

glutDisplayFunc(load\_Start);

}

void Ship()

{

glScalef(.2,.2,0);

glTranslatef(.3,.9,0);

glPushMatrix();

glTranslatef(-position\_ship,0.0, 0.0f);

glBegin(GL\_POLYGON);

glColor3ub(52, 73, 94 );

glVertex2f(-0.3f, -0.2f);

glVertex2f(-0.3f, -0.4f);

glVertex2f(0.1f, -0.4f);

glVertex2f(0.3f, -0.2f);

glEnd();

/// 1st floor

glBegin(GL\_POLYGON);

glColor3ub(51, 0, 128);

glVertex2f(-0.25f, -0.07f);

glVertex2f(-0.25f, -0.2f);

glVertex2f(0.15f, -0.2f);

glVertex2f(0.1f, -0.07f);

glEnd();

glBegin(GL\_QUADS); ///window1

glColor3ub(255, 0, 255);

glVertex2f(-0.2f, -0.09f);

glVertex2f(-0.2f, -0.18f);

glVertex2f(-0.15f, -0.18f);

glVertex2f(-0.15f, -0.09f);

glEnd();

glBegin(GL\_QUADS); ///window2

glColor3ub(241, 196, 15);

glVertex2f(-0.1f, -0.09f);

glVertex2f(-0.1f, -0.18f);

glVertex2f(-0.05f, -0.18f);

glVertex2f(-0.05f, -0.09f);

glEnd();

glBegin(GL\_QUADS); ///window3

glColor3ub(255, 0, 255);

glVertex2f(0.0f, -0.09f);

glVertex2f(0.0f, -0.18f);

glVertex2f(0.05f, -0.18f);

glVertex2f(0.05f, -0.09f);

glEnd();

///2nd floor

glBegin(GL\_POLYGON);

glColor3ub(33, 97, 140);

glVertex2f(-0.2f, 0.05f);

glVertex2f(-0.2f, -0.07f);

glVertex2f(0.06f, -0.07f);

glVertex2f(0.02f, 0.05f);

glEnd();

glBegin(GL\_QUADS); ///window1

glColor3ub(0, 153, 51);

glVertex2f(-0.18f, 0.03f);

glVertex2f(-0.18f, -0.061f);

glVertex2f(-0.13f, -0.061f);

glVertex2f(-0.13f, 0.03f);

glEnd();

/\* glBegin(GL\_QUADS); ///window2

glColor3ub(241, 196, 15);

glVertex2f(-0.12f, 0.03f);

glVertex2f(-0.12f, -0.061f);

glVertex2f(-0.07f, -0.061f);

glVertex2f(-0.07f, 0.03f);

glEnd();

\*/

glBegin(GL\_QUADS); ///window3

glColor3ub(0, 153, 51);

glVertex2f(-0.06f, 0.03f);

glVertex2f(-0.06f, -0.061f);

glVertex2f(-0.01f, -0.061f);

glVertex2f(-0.01f, 0.03f);

glEnd();

glPopMatrix();

glLoadIdentity();

}

void Ship\_2()

{

glPushMatrix();

glTranslatef(-position\_ship,0.0f, 0.0f);

glScalef(0.5,0.5,0.0);

glTranslatef(-0.7,0.6,0.0);

glBegin(GL\_POLYGON);

glColor3ub(10, 10, 30);

glVertex2f(-0.6f,-0.7f);

glVertex2f(-0.5f,-0.8f);

glVertex2f(0.3f,-0.8f);

glVertex2f(0.4f,-0.7f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(227, 218, 209);

glVertex2f(-0.48f,-0.6f);

glVertex2f(-0.5f,-0.7f);

glVertex2f(0.2f,-0.7f);

glVertex2f(0.15f,-0.6f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(255, 255, 255);

glVertex2f(-0.48f,-0.65f);

glVertex2f(-0.48f,-0.68f);

glVertex2f(0.15f,-0.68f);

glVertex2f(0.15f,-0.65f);

glEnd();

///chimni

glBegin(GL\_POLYGON);

glColor3ub(217, 185, 24);

glVertex2f(-0.4f,-0.5f);

glVertex2f(-0.4f,-0.6f);

glVertex2f(-.3f,-0.6f);

glVertex2f(-.3f,-0.5f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(217, 185, 24);

glVertex2f(-0.2f,-0.5f);

glVertex2f(-0.2f,-0.6f);

glVertex2f(-.1f,-0.6f);

glVertex2f(-.1f,-0.5f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(217, 185, 24);

glVertex2f(-0.0f,-0.5f);

glVertex2f(-0.0f,-0.6f);

glVertex2f(.1f,-0.6f);

glVertex2f(.1f,-0.5f);

glEnd();

///chimni border

glBegin(GL\_POLYGON);

glColor3ub(8, 7, 6);

glVertex2f(-0.4f,-0.5f);

glVertex2f(-0.4f,-0.48f);

glVertex2f(-.3f,-0.48f);

glVertex2f(-.3f,-0.5f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(8, 7, 6);

glVertex2f(-0.2f,-0.5f);

glVertex2f(-0.2f,-0.48f);

glVertex2f(-.1f,-0.48f);

glVertex2f(-.1f,-0.5f);

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(8, 7, 6);

glVertex2f(-0.0f,-0.5f);

glVertex2f(-0.0f,-0.48f);

glVertex2f(.1f,-0.48f);

glVertex2f(.1f,-0.5f);

glEnd();

glLoadIdentity();

glPopMatrix();

}

void Tower()

{

glTranslatef(0.7,-0.5,0.0);

glScalef(0.8,0.8,0.0);

glBegin(GL\_QUADS);

glColor3ub(27, 38, 49);

glVertex2f(-0.2f, 0.1f);

glVertex2f(0.2f, 0.1f);

glVertex2f(0.2f, -0.1f);

glVertex2f(-0.2f, -0.1f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(110, 44, 0);

glVertex2f(-0.15f, 0.28f);

glVertex2f(-0.18f, 0.28f);

glVertex2f(-0.18f, 0.1f);

glVertex2f(-0.15f, 0.1f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(110, 44, 0);

glVertex2f(0.15f, 0.28f);

glVertex2f(0.18f, 0.28f);

glVertex2f(0.18f, 0.1f);

glVertex2f(0.15f, 0.1f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(27, 38, 49);

glVertex2f(-0.2f, 0.32f);

glVertex2f(0.2f, 0.32f);

glVertex2f(0.2f, 0.28f);

glVertex2f(-0.2f, 0.28f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(27, 38, 49);

glVertex2f(-0.2, 0.32);

glVertex2f(0.0, 0.52);

glVertex2f(0.2, 0.32);

glEnd();

glBegin(GL\_LINES);

glVertex2f(0.0, 0.5);

glVertex2f(0.0, 0.65);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(188, 0, 0);

glVertex2f(0.0, 0.6);

glVertex2f(0.05, 0.625);

glVertex2f(0.0, 0.65);

glEnd();

glLineWidth(6);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(-0.2, -0.1);

glVertex2f(-0.3, -0.5);

glEnd();

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.2, -0.1);

glVertex2f(0.3, -0.5);

glEnd();

glLineWidth(4);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(-0.2, -0.1);

glVertex2f(0.28, -0.4);

glEnd();

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.2, -0.1);

glVertex2f(-0.28, -0.4);

glEnd();

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(0.08, -0.5);

glEnd();

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(-0.08, -0.1);

glVertex2f(-0.08, -0.5);

glEnd();

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

//glLoadIdentity();

glTranslatef(0.0, -0.05, 0.0);

glBegin(GL\_LINES);

glColor3ub(94, 0, 0);

glVertex2f(0.08, -0.1);

glVertex2f(-0.08, -0.1);

glEnd();

glLoadIdentity();

}

void Rain()

{

glPushMatrix();

glTranslatef(rain\_position2,rain\_position1, 0.0f);

///glLineWidth(1.5); /\*DON'T USE LINE WIDTH

// Rain line1

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glLoadIdentity();

glPopMatrix();

glTranslatef(0,0.4,0);

glPushMatrix();

glTranslatef(rain\_position2,rain\_position1, 0.0f);

// Rain line2

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glLoadIdentity();

glPopMatrix();

glTranslatef(0,-0.4,0);

glPushMatrix();

glTranslatef(rain\_position2,rain\_position1, 0.0f);

// Rain line3

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glLoadIdentity();

glPopMatrix();

glTranslatef(0,-0.8,0);

glPushMatrix();

glTranslatef(rain\_position2,rain\_position1, 0.0f);

// Rain line4

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(-.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,-0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glTranslatef(.2,0.1,0);

glColor3ub(174, 214, 241);

glBegin(GL\_LINES);

glVertex2f (-0.4,0.47);

glVertex2f (-0.35,0.28);

glEnd();

glPopMatrix();

glLoadIdentity();

}

void Sun()

{

glPushMatrix();

glTranslatef(-position\_sun, 0.0/\*position\_sun\_down\*/, 0.0f);

int is;

GLfloat xs=-0.8f;

GLfloat ys=0.8f;

GLfloat radiuss =.17f;

int triangleAmounts = 100;

GLfloat twicePis = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(255, 209, 26);

glVertex2f(xs, ys);

for(is = 0; is <= triangleAmounts; is++)

{

glVertex2f(xs+(radiuss\*cos(is\*twicePis/triangleAmounts)), ys+(radiuss\*sin(is\*twicePis/triangleAmounts)));

}

glEnd();

glPopMatrix();

}

void Moon()

{

glPushMatrix();

glTranslatef(-position\_moon, 0.0/\*position\_moon\_down\*/, 0.0f);

int is;

GLfloat xs=-0.8f;

GLfloat ys=0.8f;

GLfloat radiuss =.1f;

int triangleAmounts = 100;

GLfloat twicePis = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(255,255,240);

glVertex2f(xs, ys);

for(is = 0; is <= triangleAmounts; is++)

{

glVertex2f(xs+(radiuss\*cos(is\*twicePis/triangleAmounts)), ys+(radiuss\*sin(is\*twicePis/triangleAmounts)));

}

glEnd();

glTranslatef(0.04,0.05,0.0);

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(44,62,80);

glVertex2f(xs, ys);

for(is = 0; is <= triangleAmounts; is++)

{

glVertex2f(xs+(radiuss\*cos(is\*twicePis/triangleAmounts)), ys+(radiuss\*sin(is\*twicePis/triangleAmounts)));

}

glEnd();

glLoadIdentity();

glPopMatrix();

}

void Cloud()

{

glPushMatrix();

glTranslatef(position\_cloud, 0.0f, 0.0f);

float xC,yC,radiusC,triangleAmountC,twicePiC;

xC= -0.4f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(230, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.3f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.37f, yC= 0.67f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

glTranslatef(0.5f,0.1,0);

xC= -0.4f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(230, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.3f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.37f, yC= 0.67f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

glTranslatef(0.5f,-0.19,0);

xC= -0.4f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(230, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.3f, yC= 0.7f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

xC= -0.37f, yC= 0.67f, radiusC =.09f;

triangleAmountC = 20;

twicePiC = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(204, 255, 255);

glVertex2f(xC, yC);

for(int iC = 0; iC <= triangleAmountC;iC++)

{

glVertex2f(xC + (radiusC \* cos(iC \* twicePiC / triangleAmountC)),yC + (radiusC \* sin(iC \* twicePiC / triangleAmountC)));

}

glEnd();

glLoadIdentity();

glPopMatrix();

}

void Bird()

{

glPushMatrix();

glTranslatef(-position\_bird, 0.0f, 0.0f);

int i;

GLfloat mm=0.182f; GLfloat nn=.801f; GLfloat radiusmm =.01f;

int triangleAmount = 20;

GLfloat twicePi = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(225, 225, 208);

glVertex2f(mm, nn); // center of circle

for(i = 0; i <= triangleAmount;i++) {

glVertex2f(

mm + (radiusmm \* cos(i \* twicePi / triangleAmount)),

nn + (radiusmm \* sin(i \* twicePi / triangleAmount))

);

}

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(225, 225, 208 );

glVertex2f(0.1f,0.8f);

glVertex2f(0.11f,0.79f);

glVertex2f(0.12f,0.78f);

glVertex2f(0.16f,0.77f);

glVertex2f(0.19f,0.79f);

glVertex2f(0.201f,0.8f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(217, 217, 217);

glVertex2f(0.175f,0.8f);

glVertex2f(0.15f,0.8f);

glVertex2f(0.14f,0.84f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(242, 242, 242 );

glVertex2f(0.175f,0.8f);

glVertex2f(0.144f,0.8f);

glVertex2f(0.12f,0.83f);

glEnd();

/////2nd bird////

glBegin(GL\_POLYGON);

glColor3ub(225, 225, 208 );

glVertex2f(-0.02f,0.8f);

glVertex2f(-0.01f,0.79f);

glVertex2f(0.0f,0.78f);

glVertex2f(0.04f,0.77f);

glVertex2f(0.07f,0.79f);

glVertex2f(0.081f,0.8f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(217, 217, 217);

glVertex2f(0.055f,0.8f);

glVertex2f(0.03f,0.8f);

glVertex2f(0.02f,0.84f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(242, 242, 242 );

glVertex2f(0.055f,0.8f);

glVertex2f(0.024f,0.8f);

glVertex2f(0.0f,0.83f);

glEnd();

GLfloat mmm=0.062f; GLfloat nnn=.801f; GLfloat radiusmmm =.01f;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(225, 225, 208);

glVertex2f(mmm, nnn); // center of circle

for(i = 0; i <= triangleAmount;i++) {

glVertex2f(

mmm + (radiusmmm \* cos(i \* twicePi / triangleAmount)),

nnn + (radiusmmm \* sin(i \* twicePi / triangleAmount))

);

}

glEnd();

/////3rd bird/////

glBegin(GL\_POLYGON);

glColor3ub(225, 225, 208 );

glVertex2f(-0.72f,0.8f);

glVertex2f(-0.71f,0.79f);

glVertex2f(-0.7f,0.78f);

glVertex2f(-0.66f,0.77f);

glVertex2f(-0.63f,0.79f);

glVertex2f(-0.619f,0.8f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(217, 217, 217);

glVertex2f(-0.645f,0.8f);

glVertex2f(-0.67f,0.8f);

glVertex2f(-0.68f,0.84f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(242, 242, 242 );

glVertex2f(-0.645f,0.8f);

glVertex2f(-0.676f,0.8f);

glVertex2f(-0.7f,0.83f);

glEnd();

GLfloat mmmm=-0.638f; GLfloat nnnn=.801f;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(225, 225, 208);

glVertex2f(mmmm,nnnn); // center of circle

for(i = 0; i <= triangleAmount;i++) {

glVertex2f(

mmmm + (radiusmmm \* cos(i \* twicePi / triangleAmount)),

nnnn + (radiusmmm \* sin(i \* twicePi / triangleAmount))

);

}

glEnd();

////4th bird////

GLfloat mmmmm=-0.518f; GLfloat nnnnn=.801f;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(225, 225, 208);

glVertex2f(mmmmm, nnnnn); // center of circle

for(i = 0; i <= triangleAmount;i++) {

glVertex2f(

mmmmm + (radiusmm \* cos(i \* twicePi / triangleAmount)),

nnnnn + (radiusmm \* sin(i \* twicePi / triangleAmount))

);

}

glEnd();

glBegin(GL\_POLYGON);

glColor3ub(225, 225, 208 );

glVertex2f(-0.6f,0.8f);

glVertex2f(-0.59f,0.79f);

glVertex2f(-0.58f,0.78f);

glVertex2f(-0.54f,0.77f);

glVertex2f(-0.51f,0.79f);

glVertex2f(-0.499f,0.8f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(217, 217, 217);

glVertex2f(-0.525f,0.8f);

glVertex2f(-0.55f,0.8f);

glVertex2f(-0.56f,0.84f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(242, 242, 242 );

glVertex2f(-0.525f,0.8f);

glVertex2f(-0.556f,0.8f);

glVertex2f(-0.58f,0.83f);

glEnd();

glPopMatrix();

}

void Tree()

{

glScalef(0.5,0.5,0.0);

glTranslatef(1.0,-0.9,0.0);

glBegin(GL\_QUADS);

glColor3ub(156, 139, 102);

glVertex2f(0.8, -0.3f);

glVertex2f(0.75, -0.3f);

glVertex2f(0.75, -0.9f);

glVertex2f(0.8, -0.9f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(36, 211, 47);

glVertex2f(0.65,-0.4);

glVertex2f(0.9,-0.4);

glVertex2f(0.775,-0.1);

glEnd();

glTranslatef(0,0.06,0);

glBegin(GL\_TRIANGLES);

glColor3ub(36, 211, 47);

glVertex2f(0.65,-0.4);

glVertex2f(0.9,-0.4);

glVertex2f(0.775,-0.1);

glEnd();

glTranslatef(0.0f,0.1f,0);

glBegin(GL\_TRIANGLES);

glColor3ub(36, 211, 47);

glVertex2f(0.65,-0.4);

glVertex2f(0.9,-0.4);

glVertex2f(0.775,-0.1);

glEnd();

glLoadIdentity();

}

void Umbrella()

{

glTranslatef(-0.75, 0.00, 0.0);

glBegin(GL\_QUADS);

glColor3ub(156, 156, 161);

glVertex2f(-0.01, -0.6f);

glVertex2f(0.01, -0.6f);

glVertex2f(0.01, -0.9f);

glVertex2f(-0.01, -0.9f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(245, 203, 66);

glVertex2f(-0.05,-0.65);

glVertex2f(0.05,-0.65);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(245, 105, 66);

glVertex2f(0.13,-0.64);

glVertex2f(0.05,-0.65);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(24, 161, 26);

glVertex2f(-0.13,-0.64);

glVertex2f(-0.05,-0.65);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(82, 235, 212);

glVertex2f(-0.18,-0.63);

glVertex2f(-0.13,-0.64);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(232, 9, 9);

glVertex2f(0.18,-0.63);

glVertex2f(0.13,-0.64);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(179, 2, 219);

glVertex2f(0.21,-0.62);

glVertex2f(0.18,-0.63);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(18, 51, 219);

glVertex2f(-0.21,-0.62);

glVertex2f(-0.18,-0.63);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(142, 136, 148);

glVertex2f(-0.23,-0.61);

glVertex2f(-0.21,-0.62);

glVertex2f(0.0,-0.45);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(117, 118, 128);

glVertex2f(0.23,-0.61);

glVertex2f(0.21,-0.62);

glVertex2f(0.0,-0.45);

glEnd();

glLoadIdentity();

}

void Seat()

{

glScalef(1.8,1.8, 0);

glTranslatef(-0.42, 0.42, 0.0);

glBegin(GL\_QUADS);

glColor3ub(150, 124, 45);

glVertex2f(0.13, -0.83f);

glVertex2f(0.15, -0.83f);

glVertex2f(0.15, -0.86f);

glVertex2f(0.13, -0.866f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(171, 151, 89);

glVertex2f(0.09, -0.83f);

glVertex2f(0.16, -0.83f);

glVertex2f(0.08, -0.88f);

glVertex2f(0.02, -0.88f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(150, 124, 45);

glVertex2f(0.02, -0.8f);

glVertex2f(0.08, -0.8f);

glVertex2f(0.08, -0.9f);

glVertex2f(0.02, -0.9f);

glEnd();

glLoadIdentity();

}

void Balloon()

{

glPushMatrix();

glTranslatef(-position\_cloud, 0.0f, 0.0f);

glTranslatef(-0.8,0.5,0.0);

glScalef(0.3,0.3,0.0);

glBegin(GL\_QUADS);

glColor3ub(27, 38, 49);

glVertex2f(-0.15f, 0.1f);

glVertex2f(0.15f, 0.1f);

glVertex2f(0.15f, -0.1f);

glVertex2f(-0.15f, -0.1f);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(-0.28f, 0.4f);

glVertex2f(-0.15f, 0.1f);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(0.28f, 0.4f);

glVertex2f(0.15f, 0.1f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(110, 44, 0);

glVertex2f(-0.1f, 0.2f);

glVertex2f(0.1f, 0.2f);

glVertex2f(0.1f, 0.15f);

glVertex2f(-0.1f, 0.15f);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(-0.1f, 0.2f);

glVertex2f(-0.1f, 0.4f);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(0.1f, 0.2f);

glVertex2f(0.1f, 0.4f);

glEnd();

int i;

GLfloat x=0.0f;

GLfloat y=0.6f;

GLfloat radius =0.35f;

int triangleAmount = 100;

GLfloat twicePi = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(20, 143, 119);

glVertex2f(x, y);

for(i = 0; i <= triangleAmount; i++)

{

glVertex2f(x+(radius\*cos(i\*twicePi/triangleAmount)), y+(radius\*sin(i\*twicePi/triangleAmount)));

}

glEnd();

glLoadIdentity();

glPopMatrix();

//glLoadIdentity();

}

void Day\_Sky()

{

glBegin(GL\_QUADS);

glColor3ub(0, 153, 255);

glVertex2f(-1.0, 1.0f);

glVertex2f(1.0f, 1.0f);

glVertex2f(1.0f, .95f);

glVertex2f(-1.0f, .95f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(26, 163, 255);

glVertex2f(-1.0, .95f);

glVertex2f(1.0f, .95f);

glVertex2f(1.0f, .90f);

glVertex2f(-1.0f, .90f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(51, 173, 255);

glVertex2f(-1.0, .90f);

glVertex2f(1.0f, .90f);

glVertex2f(1.0f, .85f);

glVertex2f(-1.0f, .85f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(77, 184, 255);

glVertex2f(-1.0, .85f);

glVertex2f(1.0f, .85f);

glVertex2f(1.0f, .80f);

glVertex2f(-1.0f, .80f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(102, 194, 255);

glVertex2f(-1.0, .80f);

glVertex2f(1.0f, .80f);

glVertex2f(1.0f, .70f);

glVertex2f(-1.0f, .70f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(128, 204, 255);

glVertex2f(-1.0, .70f);

glVertex2f(1.0f, .70f);

glVertex2f(1.0f, .60f);

glVertex2f(-1.0f, .60f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(153, 214, 255);

glVertex2f(-1.0, .60f);

glVertex2f(1.0f, .60f);

glVertex2f(1.0f, .40f);

glVertex2f(-1.0f, .40f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(179, 224, 255);

glVertex2f(-1.0, .40f);

glVertex2f(1.0f, .40f);

glVertex2f(1.0f, .15f);

glVertex2f(-1.0f, .15f);

glEnd();

glEnable(GL\_LIGHTING);

GLfloat sun\_shine[] = {77, 77, 0, 7.5};

glLightModelfv(GL\_LIGHT\_MODEL\_AMBIENT, sun\_shine);

Sun();

glDisable(GL\_LIGHTING);

Cloud();

if(RainController == false)

{

Bird();

glTranslatef(0.2, -0.1, 0.0);

Bird();

glLoadIdentity();

Balloon();

}

}

void Evening\_Sky()

{

glBegin(GL\_QUADS);

glColor3ub(255, 255, 153);

glVertex2f(-1.0, 1.0f);

glVertex2f(1.0f, 1.0f);

glVertex2f(1.0f, .95f);

glVertex2f(-1.0f, .95f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 0);

glVertex2f(-1.0, .95f);

glVertex2f(1.0f, .95f);

glVertex2f(1.0f, .90f);

glVertex2f(-1.0f, .90f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 77);

glVertex2f(-1.0, .90f);

glVertex2f(1.0f, .90f);

glVertex2f(1.0f, .85f);

glVertex2f(-1.0f, .85f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 102);

glVertex2f(-1.0, .85f);

glVertex2f(1.0f, .85f);

glVertex2f(1.0f, .80f);

glVertex2f(-1.0f, .80f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 128);

glVertex2f(-1.0, .80f);

glVertex2f(1.0f, .80f);

glVertex2f(1.0f, .70f);

glVertex2f(-1.0f, .70f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 153);

glVertex2f(-1.0, .70f);

glVertex2f(1.0f, .70f);

glVertex2f(1.0f, .60f);

glVertex2f(-1.0f, .60f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 179);

glVertex2f(-1.0, .60f);

glVertex2f(1.0f, .60f);

glVertex2f(1.0f, .40f);

glVertex2f(-1.0f, .40f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(255, 255, 204);

glVertex2f(-1.0, .40f);

glVertex2f(1.0f, .40f);

glVertex2f(1.0f, .15f);

glVertex2f(-1.0f, .15f);

glEnd();

Sun();

Cloud();

if(RainController == false)

{

Bird();

glTranslatef(0.2, -0.1, 0.0);

Bird();

glLoadIdentity();

Balloon();

}

}

void drawCircle(float x1, float y1,double radius)

{

float x2, y2;

float angle;

glBegin(GL\_POLYGON);

for (angle = .1f; angle<361.0f; angle += 1.0f)

{

x2 = x1 + sin(angle)\*radius;

y2 = y1 + cos(angle)\*radius;

glVertex2f(x2, y2);

}

glEnd();

}

void Stars()

{

glPushMatrix();

glBegin(GL\_QUADS);

glColor3ub(255,255,240);

drawCircle(0.2,0.9,0.006);

drawCircle(0.4,0.9,0.006);

drawCircle(0.6,0.9,0.006);

drawCircle(0.8,0.9,0.006);

drawCircle(-0.2,0.9,0.006);

drawCircle(-0.4,0.9,0.006);

drawCircle(-0.6,0.9,0.006);

drawCircle(-0.8,0.9,0.006);

drawCircle(-0.9,0.8,0.006);

drawCircle(-0.7,0.8,0.006);

drawCircle(-0.5,0.8,0.006);

drawCircle(-0.3,0.8,0.006);

drawCircle(-0.0,0.9,0.006);

drawCircle(-0.1,0.8,0.006);

drawCircle(0.1,0.8,0.006);

drawCircle(0.3,0.8,0.006);

drawCircle(0.5,0.8,0.006);

drawCircle(0.7,0.8,0.006);

drawCircle(0.9,0.8,0.006);

drawCircle(0.2,0.7,0.006);

drawCircle(0.4,0.7,0.006);

drawCircle(0.6,0.7,0.006);

drawCircle(0.8,0.7,0.006);

drawCircle(-0.2,0.7,0.006);

drawCircle(-0.4,0.7,0.006);

drawCircle(-0.6,0.7,0.006);

drawCircle(-0.8,0.7,0.006);

drawCircle(-0.9,0.6,0.006);

drawCircle(-0.7,0.6,0.006);

drawCircle(-0.5,0.6,0.006);

drawCircle(-0.3,0.6,0.006);

drawCircle(-0.0,0.7,0.006);

drawCircle(-0.1,0.6,0.006);

drawCircle(0.1,0.6,0.006);

drawCircle(0.3,0.6,0.006);

drawCircle(0.5,0.6,0.006);

drawCircle(0.7,0.6,0.006);

drawCircle(0.9,0.6,0.006);

drawCircle(0.2,0.5,0.006);

drawCircle(0.4,0.5,0.006);

drawCircle(0.6,0.5,0.006);

drawCircle(0.8,0.5,0.006);

drawCircle(-0.2,0.5,0.006);

drawCircle(-0.4,0.5,0.006);

drawCircle(-0.6,0.5,0.006);

drawCircle(-0.8,0.5,0.006);

drawCircle(-0.9,0.4,0.006);

drawCircle(-0.7,0.4,0.006);

drawCircle(-0.5,0.4,0.006);

drawCircle(-0.3,0.4,0.006);

drawCircle(-0.0,0.5,0.006);

drawCircle(-0.1,0.4,0.006);

drawCircle(0.1,0.4,0.006);

drawCircle(0.3,0.4,0.006);

drawCircle(0.5,0.4,0.006);

drawCircle(0.7,0.4,0.006);

drawCircle(0.9,0.4,0.006);

glPopMatrix();

glEnd();

}

void Night\_Sky()

{

glBegin(GL\_QUADS);

glColor3ub(44,62,80);

glVertex2f(-1.0, 1.0f);

glVertex2f(1.0f, 1.0f);

glVertex2f(1.0f, .15f);

glVertex2f(-1.0f, .15f);

glEnd();

Moon();

Stars();

}

void Sea\_Texute(int r, int g, int b)

{

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(-0.95, 0.13);

glVertex2f(-0.90, 0.13);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(-0.75, 0.124);

glVertex2f(-0.70, 0.124);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(-0.55, 0.134);

glVertex2f(-0.50, 0.134);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(-0.35, 0.120);

glVertex2f(-0.30, 0.120);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(-0.15, 0.110);

glVertex2f(-0.10, 0.110);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(0.05, 0.130);

glVertex2f(0.10, 0.130);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(0.25, 0.1);

glVertex2f(0.20, 0.1);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(0.45, 0.13);

glVertex2f(0.40, 0.13);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(0.65, 0.1);

glVertex2f(0.60, 0.1);

glEnd();

glBegin(GL\_LINES);

glColor3ub(r, g, b);

glVertex2f(0.85, 0.1);

glVertex2f(0.80, 0.1);

glEnd();

}

void Day\_Sea()

{

glBegin(GL\_QUADS);

glColor3ub(102, 153, 255);

glVertex2f(-1.0, .15f); // top left

glVertex2f(1.0f, .15f); // top right

glVertex2f(1.0, -0.30f); // top left

glVertex2f(-1.0f, -0.30f); // top right

glEnd();

///SEA TEXURE

Sea\_Texute(61, 117, 227);

glTranslatef(0.02,-0.075,0.0);

Sea\_Texute(61, 117, 227);

glLoadIdentity();

glTranslatef(0.07,-0.15,0.0);

Sea\_Texute(61, 117, 227);

glLoadIdentity();

glTranslatef(-0.05,-0.2275,0.0);

Sea\_Texute(61, 117, 227);

glLoadIdentity();

glTranslatef(0.03,-0.35,0.0);

Sea\_Texute(61, 117, 227);

glLoadIdentity();

}

void Evening\_Sea()

{

glBegin(GL\_QUADS);

glColor3ub(77, 136, 255);

glVertex2f(-1.0, .15f); // top left

glVertex2f(1.0f, .15f); // top right

glVertex2f(1.0, -0.30f); // top left

glVertex2f(-1.0f, -0.30f); // top right

glEnd();

///SEA TEXURE

Sea\_Texute(51, 114, 242);

glTranslatef(0.02,-0.075,0.0);

Sea\_Texute(51, 114, 242);

glLoadIdentity();

glTranslatef(0.07,-0.15,0.0);

Sea\_Texute(51, 114, 242);

glLoadIdentity();

glTranslatef(-0.05,-0.2275,0.0);

Sea\_Texute(51, 114, 242);

glLoadIdentity();

glTranslatef(0.03,-0.35,0.0);

Sea\_Texute(51, 114, 242);

glLoadIdentity();

}

void Night\_Sea()

{

glBegin(GL\_QUADS);

glColor3ub(0, 34, 102);

glVertex2f(-1.0, .15f); // top left

glVertex2f(1.0f, .15f); // top right

glVertex2f(1.0, -0.30f); // top left

glVertex2f(-1.0f, -0.30f); // top right

glEnd();

///SEA TEXURE

Sea\_Texute(17, 63, 156);

glTranslatef(0.02,-0.075,0.0);

Sea\_Texute(17, 63, 156);

glLoadIdentity();

glTranslatef(0.07,-0.15,0.0);

Sea\_Texute(17, 63, 156);

glLoadIdentity();

glTranslatef(-0.05,-0.2275,0.0);

Sea\_Texute(17, 63, 156);

glLoadIdentity();

glTranslatef(0.03,-0.35,0.0);

Sea\_Texute(17, 63, 156);

glLoadIdentity();

}

void Sea\_Wave(int r, int g, int b)

{

glPushMatrix();

glTranslatef(position\_sea\_wave, 0.0f, 0.0f);

glTranslatef(0.0f, -0.14f, 0.0f);

glBegin(GL\_POLYGON);

glColor3ub(r, g, b);

glVertex2f(-1.0, -0.14f);

glVertex2f(-0.9f, -0.2f);

glVertex2f(-0.8, -0.14f);

glVertex2f(-0.7f, -0.2f);

glVertex2f(-0.6f, -0.14f);

glVertex2f(-0.5f, -0.2f);

glVertex2f(-0.4f, -0.14f);

glVertex2f(-0.3f, -0.2f);

glVertex2f(-0.2f, -0.14f);

glVertex2f(-0.1f, -0.2f);

glVertex2f(0.0f, -0.14f);

glVertex2f(0.1f, -0.1f);

glVertex2f(0.2f, -0.14f);

glVertex2f(0.3f, -0.2f);

glVertex2f(0.4f, -0.14f);

glVertex2f(0.5f, -0.2f);

glVertex2f(0.6f, -0.14f);

glVertex2f(0.7f, -0.2f);

glVertex2f(0.8f, -0.14f);

glVertex2f(0.9f, -0.2f);

glVertex2f(1.0f, -0.14f);

glEnd();

glTranslatef(0.1f, -0.01f, 0.0f);

glBegin(GL\_POLYGON);

glColor3ub(r, g, b);

glVertex2f(-1.0, -0.14f);

glVertex2f(-0.9f, -0.2f);

glVertex2f(-0.8, -0.14f);

glVertex2f(-0.7f, -0.2f);

glVertex2f(-0.6f, -0.14f);

glVertex2f(-0.5f, -0.2f);

glVertex2f(-0.4f, -0.14f);

glVertex2f(-0.3f, -0.2f);

glVertex2f(-0.2f, -0.14f);

glVertex2f(-0.1f, -0.2f);

glVertex2f(0.0f, -0.14f);

glVertex2f(0.1f, -0.1f);

glVertex2f(0.2f, -0.14f);

glVertex2f(0.3f, -0.2f);

glVertex2f(0.4f, -0.14f);

glVertex2f(0.5f, -0.2f);

glVertex2f(0.6f, -0.14f);

glVertex2f(0.7f, -0.2f);

glVertex2f(0.8f, -0.14f);

glVertex2f(0.9f, -0.2f);

glVertex2f(1.0f, -0.14f);

glEnd();

glPopMatrix();

}

void Day\_Mountain()

{

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 0);

glVertex2f(0.0f, 0.45f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glScalef(0.8,0.8,0.0);

glTranslatef(0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 0);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

glScalef(0.8,0.8,0.0);

glTranslatef(-0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 0);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

}

void Evening\_Mountain()

{

glBegin(GL\_TRIANGLES);

glColor3ub(0, 77, 0);

glVertex2f(0.0f, 0.45f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glScalef(0.8,0.8,0.0);

glTranslatef(0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(0, 77, 0);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

glScalef(0.8,0.8,0.0);

glTranslatef(-0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(0, 77, 0);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

}

void Night\_Mountain()

{

glBegin(GL\_TRIANGLES);

glColor3ub(10, 51, 10);

glVertex2f(0.0f, 0.45f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glScalef(0.8,0.8,0.0);

glTranslatef(0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(10, 51, 10);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

glScalef(0.8,0.8,0.0);

glTranslatef(-0.18,0.0335,0.0);

glBegin(GL\_TRIANGLES);

glColor3ub(10, 51, 10);

glVertex2f(0.0f, 0.35f);

glVertex2f(0.15f, 0.15f);

glVertex2f(-0.15f, 0.15f);

glEnd();

glLoadIdentity();

}

void Mill()

{

glScalef(.10,.15, 1);

glBegin(GL\_TRIANGLES);

glColor3ub(166, 166, 166);

glVertex2f(0.0f, 0.0f);

glVertex2f( 0.05f, -0.20f);

glVertex2f( -0.05f, -0.20f);

glEnd();

glBegin(GL\_QUADS);

glColor3ub(166, 166, 166);

glVertex2f(-0.05, -.20f);

glVertex2f(0.05f, -.20f);

glVertex2f(0.05f, -0.62f);

glVertex2f(-0.05f, -0.62f);

glEnd();

glPushMatrix();

glRotatef(-i,0.0,0.0,0.1);

int iw;

GLfloat xw=-0.0f;

GLfloat yw=0.0f;

GLfloat radiusw =.05f;

int triangleAmountw = 100;

GLfloat twicePiw = 2.0f \* PI;

glBegin(GL\_TRIANGLE\_FAN);

glColor3ub(0, 102, 153);

glVertex2f(xw, yw);

for(iw = 0; iw <= triangleAmountw; iw++)

{

glVertex2f(xw+(radiusw\*cos(iw\*twicePiw/triangleAmountw)), yw+(radiusw\*sin(iw\*twicePiw/triangleAmountw)));

}

glEnd();

glScalef(.7,.7, 1);

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 153);

glVertex2f(0.0f, 0.0f);

glVertex2f( 0.225f, -0.225f);

glVertex2f( 0.6f, -0.3f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 153);

glVertex2f(0.0f, 0.0f);

glVertex2f( -0.225f, -0.225f);

glVertex2f( -0.6f, -0.3f);

glEnd();

glBegin(GL\_TRIANGLES);

glColor3ub(0, 102, 153);

glVertex2f(0.0f, 0.0f);

glVertex2f( -0.1f, 0.3f);

glVertex2f( 0.0f, 0.6f);

glEnd();

glPopMatrix();

i+=-0.30f;

glLoadIdentity();

}

void Day\_Sand\_Texure()

{

///SAND TEXURE

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.95,-0.35);

glVertex2f(-0.91,-0.32);

glEnd();

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.91,-0.32);

glVertex2f(-0.91,-0.35);

glEnd();

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.91,-0.35);

glVertex2f(-0.87,-0.33);

glEnd();

}

void Evening\_Sand\_Texure()

{

///SAND TEXURE

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.95,-0.35);

glVertex2f(-0.91,-0.32);

glEnd();

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.91,-0.32);

glVertex2f(-0.91,-0.35);

glEnd();

glBegin(GL\_LINES);

glColor3ub(153, 153, 0);

glVertex2f(-0.91,-0.35);

glVertex2f(-0.87,-0.33);

glEnd();

}

void Night\_Sand\_Texure()

{

///SAND TEXURE

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(-0.95,-0.35);

glVertex2f(-0.91,-0.32);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(-0.91,-0.32);

glVertex2f(-0.91,-0.35);

glEnd();

glBegin(GL\_LINES);

glColor3ub(0, 0, 0);

glVertex2f(-0.91,-0.35);

glVertex2f(-0.87,-0.33);

glEnd();

}

void Day\_Sand()

{

glBegin(GL\_QUADS);

glColor3ub(243, 213, 120);

glVertex2f(-1.0, -0.30f);

glVertex2f(1.0f, -0.30f);

glVertex2f(1.0f, -1.0f);

glVertex2f(-1.0f, -1.0f);

glEnd();

Day\_Sand\_Texure();

glTranslatef(0.2, -0.2, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.2, -0.2, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.25, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.45, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.75, -0.45, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.85, -0.35, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.55, -0.55, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.35, -0.60, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.95, -0.60, 0.0);

Day\_Sand\_Texure();

glLoadIdentity();

///ADD MORE TEXURE

///ADD SEATS AND UMBRELLA

Umbrella();

Seat();

glTranslatef(0.9,0.0,0.0);

Umbrella();

glLoadIdentity();

glTranslatef(0.9,0.0,0.0);

Seat();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Umbrella();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Seat();

glLoadIdentity();

}

void Night\_Sand()

{

glBegin(GL\_QUADS);

glColor3ub(170, 149, 8);

glVertex2f(-1.0, -0.30f);

glVertex2f(1.0f, -0.30f);

glVertex2f(1.0f, -1.0f);

glVertex2f(-1.0f, -1.0f);

glEnd();

Night\_Sand\_Texure();

glTranslatef(0.2, -0.2, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.2, -0.2, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.25, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.45, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.75, -0.45, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.85, -0.35, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.55, -0.55, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.35, -0.60, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.95, -0.60, 0.0);

Night\_Sand\_Texure();

glLoadIdentity();

///ADD MORE TEXURE

///ADD SEATS AND UMBRELLA

Umbrella();

Seat();

glTranslatef(0.9,0.0,0.0);

Umbrella();

glLoadIdentity();

glTranslatef(0.9,0.0,0.0);

Seat();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Umbrella();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Seat();

glLoadIdentity();

}

void Evening\_Sand()

{

glBegin(GL\_QUADS);

glColor3ub(249, 233, 185);

glVertex2f(-1.0, -0.30f);

glVertex2f(1.0f, -0.30f);

glVertex2f(1.0f, -1.0f);

glVertex2f(-1.0f, -1.0f);

glEnd();

Evening\_Sand\_Texure();

glTranslatef(0.2, -0.2, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.2, -0.2, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.15, -0.3, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.35, -0.4, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.55, -0.5, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.65, -0.45, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.25, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.75, -0.45, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.75, -0.45, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.85, -0.35, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.55, -0.55, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(1.35, -0.60, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

glTranslatef(0.95, -0.60, 0.0);

Evening\_Sand\_Texure();

glLoadIdentity();

///ADD MORE TEXURE

///ADD SEATS AND UMBRELLA

Umbrella();

Seat();

glTranslatef(0.9,0.0,0.0);

Umbrella();

glLoadIdentity();

glTranslatef(0.9,0.0,0.0);

Seat();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Umbrella();

glLoadIdentity();

glScalef(0.7, 0.7, 0.0);

glTranslatef(0.35,-0.05,0.0);

Seat();

glLoadIdentity();

}

///EVENT HANDLER

void handleKeypress(unsigned char key, int x, int y)

{

switch (key)

{

///RAIN START

case 'r':

RainController = true;

///Sound Effect

sndPlaySound( "Sounds\\Rain.wav", SND\_ASYNC|SND\_LOOP );

break;

case 'R':

RainController = true;

///Sound Effect

sndPlaySound( "Sounds\\Rain.wav", SND\_ASYNC|SND\_LOOP );

break;

///RAIN END

case 'd':

RainController = false;

///Sound Effect

sndPlaySound( "Sounds\\Sea Beach.wav", SND\_ASYNC|SND\_LOOP );

break;

case 'D':

RainController = false;

///Sound Effect

sndPlaySound( "Sounds\\Sea Beach.wav", SND\_ASYNC|SND\_LOOP );

break;

}

}

///VIEWS

void View\_Night() ///NIGHT TIME

{

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

glClear(GL\_COLOR\_BUFFER\_BIT);

Night\_Sea();

Night\_Sky();

Night\_Sand();

Sea\_Wave(0, 34, 102);

Night\_Mountain();

Ship();

Ship\_2();

Tower();

glTranslatef(0.05,0.1,0.0);

Tree();

glLoadIdentity();

Tree();

glTranslatef(-0.10,0.345,0.0);

Mill();

glLoadIdentity();

glTranslatef(0.10,0.345,0.0);

Mill();

glLoadIdentity();

if(RainController == true)

{

Rain();

}

glutTimerFunc(5500,load\_Restart,0);

position\_sun = 0.0f;

glFlush();

}

void load\_Night(int x)

{

glutDisplayFunc(View\_Night);

}

void View\_Evening() ///EVENING TIME

{

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

glClear(GL\_COLOR\_BUFFER\_BIT);

Evening\_Sea();

Evening\_Sky();

Evening\_Sand();

Sea\_Wave(77, 136, 255);

Evening\_Mountain();

Ship();

Ship\_2();

Tower();

glTranslatef(0.05,0.1,0.0);

Tree();

glLoadIdentity();

Tree();

glTranslatef(-0.10,0.345,0.0);

Mill();

glLoadIdentity();

glTranslatef(0.10,0.345,0.0);

Mill();

glLoadIdentity();

if(RainController == true)

{

Rain();

}

glutTimerFunc(5500,load\_Night,0);

position\_moon = 0.0f;

glFlush();

}

void load\_Evening(int x)

{

glutDisplayFunc(View\_Evening);

}

void View\_Day() ///DAY TIME

{

glClearColor(1.0f, 1.0f, 1.0f, 0.0f);

glClear(GL\_COLOR\_BUFFER\_BIT);

Day\_Sea();

Day\_Sky();

Day\_Sand();

Sea\_Wave(102, 153, 255);

Day\_Mountain();

Ship();

Ship\_2();

Tower();

glTranslatef(0.05,0.1,0.0);

Tree();

glLoadIdentity();

Tree();

glTranslatef(-0.10,0.345,0.0);

Mill();

glLoadIdentity();

glTranslatef(0.10,0.345,0.0);

Mill();

glLoadIdentity();

if(RainController == true)

{

Rain();

}

glutTimerFunc(5500,load\_Evening,0);

glFlush();

}

void load\_Day(int x)

{

glutDisplayFunc(View\_Day);

}

void load\_Start()

{

glutDisplayFunc(View\_Day);

}

int main(int argc, char\*\* argv)

{

glutInit(&argc, argv);

glutInitWindowSize(800, 640);

glutInitWindowPosition(400, 80);

glutCreateWindow("Dynamic Sea Beach View (Including Audio Effects)");

glutDisplayFunc(load\_Start);

glutTimerFunc(100, update\_sun, 0);

glutTimerFunc(100, update\_cloud, 0);

glutTimerFunc(100, update\_bird, 0);

glutTimerFunc(100, update\_moon, 0);

glutTimerFunc(100, update\_rain, 0);

glutTimerFunc(100, update\_ship, 0);

glutTimerFunc(100, update\_sea\_wave, 0);

glutKeyboardFunc(handleKeypress);

sndPlaySound( "Sounds\\Sea Beach.wav", SND\_ASYNC|SND\_LOOP );

glutIdleFunc(Idle);

glutMainLoop();

return 0;

}