**Lab Practice-7**

Submission Guidelines-

* Rename the file to your id only. If your id is 18-XXXXX-1, then the file name must be 18-XXXXX-1.docx.

|  |
| --- |
| **Question-**  Create a simple day and night scenario that will automatically change from day to night  **Code**  #include <windows.h>  #include <GL/glut.h>  #include <math.h>  void update(int val)  {  glutPostRedisplay();  }  //void disback(int val)  //  //{  //  // glutDisplayFunc(display);  //  //}  void display2() //night  {  glClear(GL\_COLOR\_BUFFER\_BIT);  glClearColor(0.0f, 0.0f, 0.0f, 0.0f);  glLoadIdentity();  ///building  glBegin(GL\_QUADS);  glColor3ub(105, 105, 105);  glVertex2f(-0.4f,0.2f);  glVertex2f(-0.4f,-0.5f);  glVertex2f(0.4f,-0.5f);  glVertex2f(0.4f,0.2f);  glEnd();  ///building line  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(0.0f,0.1f);  glVertex2f(0.0f,-0.5f);  glEnd();  //tri  glBegin(GL\_TRIANGLES);  glColor3f(0.0f, 0.0f, 1.0f);  glVertex2f(-0.4f,0.2f);  glVertex2f(0.4f, 0.2f);  glVertex2f(0.0f,0.6f);  glEnd();  ///Ground  glBegin(GL\_QUADS);  glColor3ub(0,154,23);  glVertex2f(-1.0f,-0.5f);  glVertex2f(-1.0f,-1.0f);  glVertex2f(1.0f,-1.0f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///road  glBegin(GL\_QUADS);  glColor3ub(0, 0, 0);  glVertex2f(-1.0f,-0.5f);  glVertex2f(-1.0f,-0.7f);  glVertex2f(1.0f,-0.7f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///roadline  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(-1.0f,-0.5f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///roadline2  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(-1.0f,-0.7f);  glVertex2f(1.0f,-0.7f);  glEnd();  //moon  glTranslatef(0.7f,0.5f,0.0f);  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(250, 240, 230);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=0.15;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x,y);  }  glEnd();  // glutTimerFunc(2000,disback,0);  glutPostRedisplay();  glFlush();  }  void dis2(int val)  {  glutDisplayFunc(display2);  }  void display() {  glClear(GL\_COLOR\_BUFFER\_BIT);  glClearColor(0.0f, 1.0f, 1.0f, 1.0f);  glLoadIdentity();  ///building  glBegin(GL\_QUADS);  glColor3ub(255 ,127, 80);  glVertex2f(-0.4f,0.2f);  glVertex2f(-0.4f,-0.5f);  glVertex2f(0.4f,-0.5f);  glVertex2f(0.4f,0.2f);;  glEnd();  ///building line  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(0.0f,0.1f);  glVertex2f(0.0f,-0.5f);  glEnd();  //tri  glBegin(GL\_TRIANGLES);  glColor3f(0.0f, 0.0f, 1.0f);  glVertex2f(-0.4f,0.2f);  glVertex2f(0.4f, 0.2f);  glVertex2f(0.0f,0.6f);  glEnd();  ///Ground  glBegin(GL\_QUADS);  glColor3ub(0,154,23);  glVertex2f(-1.0f,-0.5f);  glVertex2f(-1.0f,-1.0f);  glVertex2f(1.0f,-1.0f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///road  glBegin(GL\_QUADS);  glColor3ub(0, 0, 0);  glVertex2f(-1.0f,-0.5f);  glVertex2f(-1.0f,-0.7f);  glVertex2f(1.0f,-0.7f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///roadline  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(-1.0f,-0.5f);  glVertex2f(1.0f,-0.5f);  glEnd();  ///roadline2  glLineWidth(0.1);  glBegin(GL\_LINES);  glColor3ub(255,255,255);  glVertex2f(-1.0f,-0.7f);  glVertex2f(1.0f,-0.7f);  glEnd();  //sun  glTranslatef(0.7f,0.5f,0.0f);  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(249,215,28);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=0.15;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x,y);  }  glEnd();  //tree  // glBegin(GL\_QUADS);  // glColor3ub(255 ,127, 80);  // glVertex2f(-1.0f,0.0f);  // glVertex2f(-0.7f,0.0f);  // glVertex2f(-0.8f,-0.3f);  // glVertex2f(-0.9f,-0.3f);;  // glEnd();  // glVertex2f(0.4f,0.0f);  //glEnd();  glutTimerFunc(2000,dis2,0);  glutPostRedisplay();  glFlush();  }  int main(int argc, char\*\* argv) {  glutInit(&argc, argv);  glutCreateWindow("OpenGL Setup Test");  glutInitWindowSize(200, 180);  glutDisplayFunc(display);  glutMainLoop();  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Create a simple day and night scenario using keyboard interaction. The key ‘D’ or ‘d’ will initiate the day mode and the key ‘N’ or ‘n’ will initiate the night mode.  **Code:**  **#include<cstdio>**  **#include<math.h>**  **#include <GL/gl.h>**  **#include <GL/glut.h>**  **void update(int val)**  **{**  **glutPostRedisplay();**  **}**  **void display2() //night**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 0.0f, 0.0f);**  **glLoadIdentity();**  **///building**  **glBegin(GL\_QUADS);**  **glColor3ub(105, 105, 105);**  **glVertex2f(-0.4f,0.2f);**  **glVertex2f(-0.4f,-0.5f);**  **glVertex2f(0.4f,-0.5f);**  **glVertex2f(0.4f,0.2f);**  **glEnd();**  **///building line**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(0.0f,0.1f);**  **glVertex2f(0.0f,-0.5f);**  **glEnd();**  **//tri**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(-0.4f,0.2f);**  **glVertex2f(0.4f, 0.2f);**  **glVertex2f(0.0f,0.6f);**  **glEnd();**  **///Ground**  **glBegin(GL\_QUADS);**  **glColor3ub(0,154,23);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(-1.0f,-1.0f);**  **glVertex2f(1.0f,-1.0f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///road**  **glBegin(GL\_QUADS);**  **glColor3ub(0, 0, 0);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(-1.0f,-0.7f);**  **glVertex2f(1.0f,-0.7f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///roadline**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///roadline2**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(-1.0f,-0.7f);**  **glVertex2f(1.0f,-0.7f);**  **glEnd();**  **//moon**  **glTranslatef(0.7f,0.5f,0.0f);**  **glBegin(GL\_POLYGON);**  **for(int i=0;i<200;i++)**  **{**  **glColor3ub(250, 240, 230);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.15;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **// glutTimerFunc(2000,disback,0);**  **glutPostRedisplay();**  **glFlush();**  **}**  **void dis2(int val)**  **{**  **glutDisplayFunc(display2);**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 1.0f, 1.0f, 1.0f);**  **glLoadIdentity();**  **///building**  **glBegin(GL\_QUADS);**  **glColor3ub(255 ,127, 80);**  **glVertex2f(-0.4f,0.2f);**  **glVertex2f(-0.4f,-0.5f);**  **glVertex2f(0.4f,-0.5f);**  **glVertex2f(0.4f,0.2f);;**  **glEnd();**  **///building line**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(0.0f,0.1f);**  **glVertex2f(0.0f,-0.5f);**  **glEnd();**  **//tri**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(-0.4f,0.2f);**  **glVertex2f(0.4f, 0.2f);**  **glVertex2f(0.0f,0.6f);**  **glEnd();**  **///Ground**  **glBegin(GL\_QUADS);**  **glColor3ub(0,154,23);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(-1.0f,-1.0f);**  **glVertex2f(1.0f,-1.0f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///road**  **glBegin(GL\_QUADS);**  **glColor3ub(0, 0, 0);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(-1.0f,-0.7f);**  **glVertex2f(1.0f,-0.7f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///roadline**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(-1.0f,-0.5f);**  **glVertex2f(1.0f,-0.5f);**  **glEnd();**  **///roadline2**  **glLineWidth(0.1);**  **glBegin(GL\_LINES);**  **glColor3ub(255,255,255);**  **glVertex2f(-1.0f,-0.7f);**  **glVertex2f(1.0f,-0.7f);**  **glEnd();**  **//sun**  **glTranslatef(0.7f,0.5f,0.0f);**  **glBegin(GL\_POLYGON);**  **for(int i=0;i<200;i++)**  **{**  **glColor3ub(249,215,28);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.15;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **// glutTimerFunc(2000,dis2,0);**  **glutPostRedisplay();**  **glFlush();**  **}**  **void handleKeypress(unsigned char key, int x, int y) {**  **switch (key) {**  **case 'd':**  **glutDisplayFunc(display);**  **glutPostRedisplay();**  **break;**  **case 'n':**  **glutDisplayFunc(display2);**  **glutPostRedisplay();**  **break;**  **}**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitWindowSize(600, 600);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Problem no 2");**  **glutDisplayFunc(display);**  **// init();**  **glutKeyboardFunc(handleKeypress);**  **glutTimerFunc(20, update, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |