**Lab Practice-7**

|  |
| --- |
| **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 Circle(float radius, float xc, float yc, float r, float g, float b)  {  glLineWidth(7.5);  glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin  for(int i=0;i<200;i++)  {  glColor3ub(r,g,b);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=radius;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+xc,y+yc );  }  glEnd();  }  void CircleBorder(float radius, float xc, float yc, float width)  {  glLineWidth(width);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin  for(int i=0;i<200;i++)  {  glColor3ub(0,0,0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=radius;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+xc,y+yc );  }  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void skyDay()  {  glBegin(GL\_POLYGON); //sky  glColor3ub(46, 222, 232);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  Circle(15, 140,120,234,242,0);  CircleBorder(15, 140,120,2);  glLineWidth(4);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //sky  glColor3ub(0,0,0);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void skyNight()  {  glBegin(GL\_POLYGON); //sky  glColor3ub(0, 0, 0);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  Circle(15, 140,120,255,255,255);  CircleBorder(15, 140,120,2);  glLineWidth(4);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //sky  glColor3ub(255,255,255);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  Circle(0.5, 79,110,255,255,255);  Circle(0.5, -23,90,255,255,255);  Circle(0.5, 138,70,255,255,255);  Circle(0.5, -177,130,255,255,255);  Circle(0.5, -190,67,255,255,255);  Circle(0.5, -78,142,255,255,255);  Circle(0.5, 90,100,255,255,255);  Circle(0.5, 50,159,255,255,255);  Circle(0.5, 105,89,255,255,255);  Circle(0.5, 40,100,255,255,255);  Circle(0.5, 78,82,255,255,255);  Circle(0.5, 76,99,255,255,255);  Circle(0.5, 38,134,255,255,255);  }  void backgroundDay()  {  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,100);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glBegin(GL\_POLYGON); //grass  glColor3ub(0,55,0);  glVertex2i(-200,-60);  glVertex2i(-200,40);  glVertex2f(200,40);  glVertex2f(200,-60);  glEnd();  skyDay();  //skyNight();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glLineWidth(6);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,0);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //grass  glColor3ub(0,0,0);  glVertex2i(-200,-60);  glVertex2i(-200,40);  glVertex2f(200,40);  glVertex2f(200,-60);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void backgroundNight()  {  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,100);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glBegin(GL\_POLYGON); //grass  glColor3ub(0,55,0);  glVertex2i(-200,-60);  glVertex2i(-200,60);  glVertex2f(200,60);  glVertex2f(200,-60);  glEnd();  //skyDay();  skyNight();  glLineWidth(6);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,0);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //grass  glColor3ub(0,0,0);  glVertex2i(-200,-60);  glVertex2i(-200,60);  glVertex2f(200,60);  glVertex2f(200,-60);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void house1()  {  glLineWidth(5);  glBegin(GL\_POLYGON); //house p1  glColor3ub(89, 46, 3);  glVertex2i(-100,30+20);  glVertex2i(-80,65+20);  glVertex2i(-75,60+20);  glVertex2i(-95,27+20);  glEnd();  glBegin(GL\_POLYGON); //house p2  glColor3ub(89, 46, 3);  glVertex2i(-80,65+20);  glVertex2i(-60,27+20);  glVertex2i(-55,30+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house roof  glColor3ub(89, 46, 3);  glVertex2i(-60,65+20);  glVertex2i(-10,65+20);  glVertex2i(10,27+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house  glColor3ub(89, 46, 3);  glVertex2i(-75,65+20);  glVertex2i(-60,27+20);  glVertex2i(10,27+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house body  glColor3ub(242, 121, 0);  glVertex2i(-95,27+20);  glVertex2i(-76,58+20);  glVertex2i(-60,27+20);  glVertex2i(-60,-15+20);  glVertex2i(-95,-15+20);  glEnd();  glBegin(GL\_POLYGON); //house body  glColor3ub(242, 121, 0);  glVertex2i(-60,27+20);  glVertex2i(5,27+20);  glVertex2i(5,-15+20);  glVertex2i(-60,-15+20);  glEnd();  glBegin(GL\_POLYGON); //house floor  glColor3ub(122, 63, 4);  glVertex2i(-100,-15+20);  glVertex2i(10,-15+20);  glVertex2i(10,-20+20);  glVertex2i(-100,-20+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house p1  glColor3ub(0,0,0);  glVertex2i(-100,30+20);  glVertex2i(-80,65+20);  glVertex2i(-75,60+20);  glVertex2i(-95,27+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house p2  glColor3ub(0,0,0);  glVertex2i(-80,65+20);  glVertex2i(-60,27+20);  glVertex2i(-55,30+20);  glVertex2i(-75,65+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house roof  glColor3ub(0,0,0);  glVertex2i(-75,65+20);  glVertex2i(-10,65+20);  glVertex2i(10,27+20);  glVertex2i(-53,27+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house body  glColor3ub(0,0,0);  glVertex2i(-95,27+20);  glVertex2i(-76,58+20);  glVertex2i(-60,27+20);  glVertex2i(-60,-15+20);  glVertex2i(-95,-15+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house body  glColor3ub(0,0,0);  glVertex2i(-60,27+20);  glVertex2i(5,27+20);  glVertex2i(5,-15+20);  glVertex2i(-60,-15+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house floor  glColor3ub(0,0,0);  glVertex2i(-100,-15+20);  glVertex2i(10,-15+20);  glVertex2i(10,-20+20);  glVertex2i(-100,-20+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glBegin(GL\_POLYGON); //house window  glColor3ub(245, 49, 5);  glVertex2i(17-100,25);  glVertex2i(29-100,25);  glVertex2i(29-100,35);  glVertex2i(17-100,35);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house window border  glColor3ub(0,0,0);  glVertex2i(17-100,25);  glVertex2i(29-100,25);  glVertex2i(29-100,35);  glVertex2i(17-100,35);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glBegin(GL\_POLYGON); //house door  glColor3ub(245, 49, 5);  glVertex2i(16-50,25-20);  glVertex2i(30-50,25-20);  glVertex2i(30-50,45-20);  glVertex2i(16-50,45-20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house door  glColor3ub(0,0,0);  glVertex2i(16-50,25-20);  glVertex2i(30-50,25-20);  glVertex2i(30-50,45-20);  glVertex2i(16-50,45-20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void tree(float x, float y)  {  glLineWidth(5);  glBegin(GL\_POLYGON);  glColor3ub(165,99,60);  glVertex2i(60+x,-80+y);  glVertex2i(60+x,00+y);  glVertex2i(80+x,00+y);  glVertex2f(80+x,-80+y);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON);  glColor3ub(0,0,00);  glVertex2i(60+x,-80+y);  glVertex2i(60+x,00+y);  glVertex2i(80+x,00+y);  glVertex2f(80+x,-80+y);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  Circle(18, 40+x,7+y,0,176,65);  CircleBorder(18,40+x, 7+y,7);  Circle(18, 100+x,7+y,0,176,65);  CircleBorder(18,100+x, 7+y,7);  Circle(30, 70+x,28+y,0,176,65);  CircleBorder(30,70+x, 28+y,7);  Circle(10, 63+x,00+y,0,176,65);  CircleBorder(10,63+x, 00+y,7);  Circle(10, 77+x,00+y,0,176,65);  CircleBorder(10,77+x, 00+y,7);  Circle(12, 70+x,5+y,0,176,65);  Circle(15, 48+x,11+y,0,176,65);  Circle(15, 91+x,12+y,0,176,65);  }  void displayDay(int value)  {  backgroundDay();  tree(-170,90);  house1();  glFlush();  }  void displayNight(int value)  {  backgroundNight();  tree(-170,90);  house1();  glFlush();  }  void display()  {  displayDay(1);  glutTimerFunc(2000, displayNight, 0);  }  int main(int argc, char\*\* argv)  {  glutInit(&argc, argv);  glutInitWindowSize(1200, 1200);  glutCreateWindow("lab task 7 [22-47226-1]");  glutDisplayFunc(display);  gluOrtho2D(-200,200,-150,150);  glFlush();  glutMainLoop();  return 0;  } |
| **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 <windows.h>  #include <GL/glut.h>  #include <math.h>  void Circle(float radius, float xc, float yc, float r, float g, float b)  {  glLineWidth(7.5);  glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin  for(int i=0;i<200;i++)  {  glColor3ub(r,g,b);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=radius;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+xc,y+yc );  }  glEnd();  }  void CircleBorder(float radius, float xc, float yc, float width)  {  glLineWidth(width);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin  for(int i=0;i<200;i++)  {  glColor3ub(0,0,0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=radius;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+xc,y+yc );  }  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void skyDay()  {  glBegin(GL\_POLYGON); //sky  glColor3ub(46, 222, 232);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  Circle(15, 140,120,234,242,0);  CircleBorder(15, 140,120,2);  glLineWidth(4);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //sky  glColor3ub(0,0,0);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void skyNight()  {  glBegin(GL\_POLYGON); //sky  glColor3ub(0, 0, 0);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  Circle(15, 140,120,255,255,255);  CircleBorder(15, 140,120,2);  glLineWidth(4);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //sky  glColor3ub(255,255,255);  glVertex2i(-200,40);  glVertex2i(-200,150);  glVertex2f(200,150);  glVertex2f(200,40);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  Circle(0.5, 79,110,255,255,255);  Circle(0.5, -23,90,255,255,255);  Circle(0.5, 138,70,255,255,255);  Circle(0.5, -177,130,255,255,255);  Circle(0.5, -190,67,255,255,255);  Circle(0.5, -78,142,255,255,255);  Circle(0.5, 90,100,255,255,255);  Circle(0.5, 50,159,255,255,255);  Circle(0.5, 105,89,255,255,255);  Circle(0.5, 40,100,255,255,255);  Circle(0.5, 78,82,255,255,255);  Circle(0.5, 76,99,255,255,255);  Circle(0.5, 38,134,255,255,255);  }  void backgroundDay()  {  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,100);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glBegin(GL\_POLYGON); //grass  glColor3ub(0,55,0);  glVertex2i(-200,-60);  glVertex2i(-200,40);  glVertex2f(200,40);  glVertex2f(200,-60);  glEnd();  skyDay();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glLineWidth(6);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,0);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //grass  glColor3ub(0,0,0);  glVertex2i(-200,-60);  glVertex2i(-200,40);  glVertex2f(200,40);  glVertex2f(200,-60);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void backgroundNight()  {  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,100);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glBegin(GL\_POLYGON); //grass  glColor3ub(0,55,0);  glVertex2i(-200,-60);  glVertex2i(-200,60);  glVertex2f(200,60);  glVertex2f(200,-60);  glEnd();  skyNight();  glLineWidth(6);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //river  glColor3ub(0,0,0);  glVertex2i(-200,-150);  glVertex2i(-200,-60);  glVertex2f(200,-60);  glVertex2f(200,-150);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //grass  glColor3ub(0,0,0);  glVertex2i(-200,-60);  glVertex2i(-200,60);  glVertex2f(200,60);  glVertex2f(200,-60);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void house1()  {  glLineWidth(5);  glBegin(GL\_POLYGON); //house p1  glColor3ub(89, 46, 3);  glVertex2i(-100,30+20);  glVertex2i(-80,65+20);  glVertex2i(-75,60+20);  glVertex2i(-95,27+20);  glEnd();  glBegin(GL\_POLYGON); //house p2  glColor3ub(89, 46, 3);  glVertex2i(-80,65+20);  glVertex2i(-60,27+20);  glVertex2i(-55,30+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house roof  glColor3ub(89, 46, 3);  glVertex2i(-60,65+20);  glVertex2i(-10,65+20);  glVertex2i(10,27+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house  glColor3ub(89, 46, 3);  glVertex2i(-75,65+20);  glVertex2i(-60,27+20);  glVertex2i(10,27+20);  glVertex2i(-75,65+20);  glEnd();  glBegin(GL\_POLYGON); //house body  glColor3ub(242, 121, 0);  glVertex2i(-95,27+20);  glVertex2i(-76,58+20);  glVertex2i(-60,27+20);  glVertex2i(-60,-15+20);  glVertex2i(-95,-15+20);  glEnd();  glBegin(GL\_POLYGON); //house body  glColor3ub(242, 121, 0);  glVertex2i(-60,27+20);  glVertex2i(5,27+20);  glVertex2i(5,-15+20);  glVertex2i(-60,-15+20);  glEnd();  glBegin(GL\_POLYGON); //house floor  glColor3ub(122, 63, 4);  glVertex2i(-100,-15+20);  glVertex2i(10,-15+20);  glVertex2i(10,-20+20);  glVertex2i(-100,-20+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house p1  glColor3ub(0,0,0);  glVertex2i(-100,30+20);  glVertex2i(-80,65+20);  glVertex2i(-75,60+20);  glVertex2i(-95,27+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house p2  glColor3ub(0,0,0);  glVertex2i(-80,65+20);  glVertex2i(-60,27+20);  glVertex2i(-55,30+20);  glVertex2i(-75,65+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house roof  glColor3ub(0,0,0);  glVertex2i(-75,65+20);  glVertex2i(-10,65+20);  glVertex2i(10,27+20);  glVertex2i(-53,27+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house body  glColor3ub(0,0,0);  glVertex2i(-95,27+20);  glVertex2i(-76,58+20);  glVertex2i(-60,27+20);  glVertex2i(-60,-15+20);  glVertex2i(-95,-15+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house body  glColor3ub(0,0,0);  glVertex2i(-60,27+20);  glVertex2i(5,27+20);  glVertex2i(5,-15+20);  glVertex2i(-60,-15+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house floor  glColor3ub(0,0,0);  glVertex2i(-100,-15+20);  glVertex2i(10,-15+20);  glVertex2i(10,-20+20);  glVertex2i(-100,-20+20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glBegin(GL\_POLYGON); //house window  glColor3ub(245, 49, 5);  glVertex2i(17-100,25);  glVertex2i(29-100,25);  glVertex2i(29-100,35);  glVertex2i(17-100,35);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house window border  glColor3ub(0,0,0);  glVertex2i(17-100,25);  glVertex2i(29-100,25);  glVertex2i(29-100,35);  glVertex2i(17-100,35);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  glBegin(GL\_POLYGON); //house door  glColor3ub(245, 49, 5);  glVertex2i(16-50,25-20);  glVertex2i(30-50,25-20);  glVertex2i(30-50,45-20);  glVertex2i(16-50,45-20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON); //house door  glColor3ub(0,0,0);  glVertex2i(16-50,25-20);  glVertex2i(30-50,25-20);  glVertex2i(30-50,45-20);  glVertex2i(16-50,45-20);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  }  void tree(float x, float y)  {  glLineWidth(5);  glBegin(GL\_POLYGON);  glColor3ub(165,99,60);  glVertex2i(60+x,-80+y);  glVertex2i(60+x,00+y);  glVertex2i(80+x,00+y);  glVertex2f(80+x,-80+y);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);  glBegin(GL\_POLYGON);  glColor3ub(0,0,00);  glVertex2i(60+x,-80+y);  glVertex2i(60+x,00+y);  glVertex2i(80+x,00+y);  glVertex2f(80+x,-80+y);  glEnd();  glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);  Circle(18, 40+x,7+y,0,176,65);  CircleBorder(18,40+x, 7+y,7);  Circle(18, 100+x,7+y,0,176,65);  CircleBorder(18,100+x, 7+y,7);  Circle(30, 70+x,28+y,0,176,65);  CircleBorder(30,70+x, 28+y,7);  Circle(10, 63+x,00+y,0,176,65);  CircleBorder(10,63+x, 00+y,7);  Circle(10, 77+x,00+y,0,176,65);  CircleBorder(10,77+x, 00+y,7);  Circle(12, 70+x,5+y,0,176,65);  Circle(15, 48+x,11+y,0,176,65);  Circle(15, 91+x,12+y,0,176,65);  }  void displayDay(int value)  {  backgroundDay();  tree(-170,90);  house1();  glFlush();  }  void displayNight(int value)  {  backgroundNight();  tree(-170,90);  house1();  glFlush();  }  void display()  {  displayDay(1);  }  void handleKeypress(unsigned char key, int x, int y)  {  switch (key)  {  case 'd':  glutTimerFunc(1, displayDay, 0);  break;  case 'n':  glutTimerFunc(1, displayNight, 0);  break;  case 'D':  glutTimerFunc(1, displayDay, 0);  break;  case 'N':  glutTimerFunc(1, displayNight, 0);  break;  glutPostRedisplay();  }  }  int main(int argc, char\*\* argv)  {  glutInit(&argc, argv);  glutInitWindowSize(1200, 1200);  glutCreateWindow("lab task 7 [22-47226-1]");  glutKeyboardFunc(handleKeypress);  glutDisplayFunc(display);  gluOrtho2D(-200,200,-150,150);  glFlush();  glutMainLoop();  return 0;  } |
| **Output Screenshot (Full Screen)-** |