**Lab Taks-4**

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.
* Must submit within time that will be discussed in class VUES to the section named Lab Tak-4
* Must include resources for all the section in the table

|  |
| --- |
| **Question- 1**  Draw the scenario of a traffic signal using function to represent each object |
| **Graph Plot (Picture)-** |
| **Code-**  #include <windows.h>  #include <GL/glut.h>  #include <math.h>  void circle(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(253, 2, 25);  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+cX,y+cY);  }  glEnd();  }  void circle1(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(253, 245, 2 );  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+cX,y+cY);  }  glEnd();  }  void circle2(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 167, 25 );  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+cX,y+cY);  }  glEnd();  }  void circle3(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(86, 101, 115 );  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+cX,y+cY);  }  glEnd();  }  void circle4(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(86, 101, 115 );  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+cX,y+cY);  }  glEnd();  }  void display() {  glClear(GL\_COLOR\_BUFFER\_BIT);  glBegin(GL\_POLYGON);  glColor3ub(235, 152, 78 );  glVertex2f(-8.0,1.50);  glVertex2f(8.0,1.50);  glVertex2f(8.0,0.0);  glVertex2f(-8.0,0.0);  glVertex2f(-8.0,1.50);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(151, 154, 154 );  glVertex2f(-8.0,0.0);  glVertex2f(8.0,0.0);  glVertex2f(8.0,-6.0);  glVertex2f(-8.0,-6.0);  glVertex2f(-8.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42 );  glVertex2f(-8.0,0.0);  glVertex2f(-6.0,0.0);  glVertex2f(-6.0,-6.0);  glVertex2f(-8.0,-6.0);  glVertex2f(-8.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241 );  glVertex2f(-6.0,0.0);  glVertex2f(-4.0,0.0);  glVertex2f(-4.0,-6.0);  glVertex2f(-6.0,-6.0);  glVertex2f(-6.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42 );  glVertex2f(-4.0,0.0);  glVertex2f(-2.0,0.0);  glVertex2f(-2.0,-6.0);  glVertex2f(-4.0,-6.0);  glVertex2f(-4.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241 );  glVertex2f(-2.0,0.0);  glVertex2f(0.0,0.0);  glVertex2f(0.0,-6.0);  glVertex2f(-2.0,-6.0);  glVertex2f(-2.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42 );  glVertex2f(0.0,0.0);  glVertex2f(2.0,0.0);  glVertex2f(2.0,-6.0);  glVertex2f(0.0,-6.0);  glVertex2f(0.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241);  glVertex2f(2.0,0.0);  glVertex2f(4.0,0.0);  glVertex2f(4.0,-6.0);  glVertex2f(2.0,-6.0);  glVertex2f(2.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42);  glVertex2f(4.0,0.0);  glVertex2f(6.0,0.0);  glVertex2f(6.0,-6.0);  glVertex2f(4.0,-6.0);  glVertex2f(4.0,0.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241);  glVertex2f(6.0,0.0);  glVertex2f(8.0,0.0);  glVertex2f(8.0,-6.0);  glVertex2f(6.0,-6.0);  glVertex2f(6.0,0.0);  glEnd();  glBegin(GL\_LINES);  glColor3ub(23, 32, 42);  glVertex2f(-8.0,-6.0);  glVertex2f(8.0,-6.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42);  glVertex2f(4.50,0.50);  glVertex2f(5.25,0.50);  glVertex2f(5.25,5.50);  glVertex2f(4.50,5.50);  glVertex2f(4.50,0.50);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(23, 32, 42);  glVertex2f(3.00,5.50);  glVertex2f(6.50,5.50);  glVertex2f(6.50,7.0);  glVertex2f(3.00,7.0);  glVertex2f(3.00,5.50);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(253, 245, 2 );  glVertex2f(-6.5,-4.00);  glVertex2f(-6.50,-2.0);  glVertex2f(-3.75,-2.0);  glVertex2f(-3.00,-0.50);  glVertex2f(-0.50,-0.50);  glVertex2f(0.25,-2.0);  glVertex2f(3.25,-2.0);  glVertex2f(3.25,-4.00);  glVertex2f(1.75,-4.0);  glVertex2f(1.25,-3.50);  glVertex2f(0.75,-3.50);  glVertex2f(0.25,-4.00);  glVertex2f(-4.25,-4.00);  glVertex2f(-4.75,-3.50);  glVertex2f(-5.25,-3.50);  glVertex2f(-5.75,-4.00);  glVertex2f(-6.50,-4.00);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 231, 233 );  glVertex2f(-1.75,-1.00);  glVertex2f(-1.0,-1.0);  glVertex2f(-0.25,-2.00);  glVertex2f(-1.75,-2.00);  glVertex2f(-1.75,-1.00);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 231, 233 );  glVertex2f(-2.25,-1.00);  glVertex2f(-3.0,-1.0);  glVertex2f(-3.50,-2.00);  glVertex2f(-2.25,-2.00);  glVertex2f(-2.25,-1.00);  glEnd();  circle(0.5,3.75,6.25);  circle1(0.5,4.75,6.25);  circle2(0.5,5.75,6.25);  circle3(0.75,-5.00,-4.00);  circle4(0.75,1.0,-4.00);  glFlush();  }  int main(int argc, char\*\* argv) {  glutInit(&argc, argv);  glutInitWindowSize(1200,1000);  glutCreateWindow("Traffic Light");  glClearColor(1.0,1.0,1.0,1.0);  gluOrtho2D(-8.0,8.0,-8.0,8.0);  glutDisplayFunc(display);  glutMainLoop();  return 0;  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 2**  Draw two village scenarios for day and night using function to represent each object |
| **Graph Plot (Picture)-**      **Same Graph Plot use for for day and night light** |
| **Code-**  **Code For Day Light:**  #include <windows.h>  #include <GL/glut.h>  #include <math.h>  void circle1(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(253, 245, 2 );  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+cX,y+cY);  }  glEnd();  }  void circle2(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void circle3(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void circle4(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void display() {  glClear(GL\_COLOR\_BUFFER\_BIT);  glBegin(GL\_POLYGON);  glColor3ub(209, 234, 250);  glVertex2f(-30.0,9.0);  glVertex2f(-30.0,20.0);  glVertex2f(30.0,20.0);  glVertex2f(30.0,9.0);  glVertex2f(-30.0,9.0);  glEnd();  // M-1  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-30.0,9.0);  glVertex2f(-24.0,16.0);  glVertex2f(-19.0,9.0);  glVertex2f(-30.0,9.0);  glEnd();  //M-2  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-19.0,9.0);  glVertex2f(-14.0,16.0);  glVertex2f(-8.0,9.0);  glVertex2f(-19.0,9.0);  glEnd();  //M-3  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-8.0,9.0);  glVertex2f(-3.0,16.0);  glVertex2f(2.0,9.0);  glVertex2f(-8.0,9.0);  glEnd();  //M-4  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(2.0,9.0);  glVertex2f(7.0,16.0);  glVertex2f(12.0,9.0);  glVertex2f(2.0,9.0);  glEnd();  //M-5  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(12.0,9.0);  glVertex2f(17.0,16.0);  glVertex2f(22.0,9.0);  glVertex2f(12.0,9.0);  glEnd();  //M-6  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(22.0,9.0);  glVertex2f(26.0,16.0);  glVertex2f(30.0,9.0);  glVertex2f(22.0,9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(-14.0,9.0);  glVertex2f(-13.0,9.0);  glVertex2f(-2.0,1.0);  glVertex2f(-3.0,1.0);  glVertex2f(-14.0,9.0);  glEnd();  glBegin(GL\_LINES);  glColor3ub(135, 54, 0 );  glVertex2f(-3.0,1.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(-5.0,1.0);  glVertex2f(6.0,-10.0);  glVertex2f(5.0,-10.0);  glVertex2f(-5.0,0.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_LINES);  glColor3ub(135,54,0);  glVertex2f(5.00,-10.0);  glVertex2f(3.0,-10.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(3.0,-10.0);  glVertex2f(13.0,-20.0);  glVertex2f(12.0,-20.0);  glVertex2f(3.0,-11.0);  glVertex2f(3.0,-10.0);  glEnd();  // River  glBegin(GL\_POLYGON);  glColor3ub(52, 152, 219 );  glVertex2f(-30.0,9.0);  glVertex2f(-14.0,9.0);  glVertex2f(-3.0,1.0);  glVertex2f(-5.0,1.0);  glVertex2f(-5.0,0.0);  glVertex2f(5.0,-10.0);  glVertex2f(3.0,-10.0);  glVertex2f(3.0,-11.0);  glVertex2f(12.0,-20.0);  glVertex2f(-30.0,-20.0);  glVertex2f(-30.0,9.0);  glEnd();  // Boat-1  glBegin(GL\_POLYGON);  glColor3ub(28, 40, 51 );  glVertex2f(-26.0,-1.0);  glVertex2f(-10.0,-1.0);  glVertex2f(-12.0,-4.0);  glVertex2f(-24.0,-4.0);  glVertex2f(-26.0,-1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(231, 76, 60 );  glVertex2f(-21.0,-1.0);  glVertex2f(-18.0,3.0);  glVertex2f(-14.0,-1.0);  glVertex2f(-21.0,-1.0);  glEnd();  // Boat-2  glBegin(GL\_POLYGON);  glColor3ub(28, 40, 51 );  glVertex2f(-20.0,-14.0);  glVertex2f(-6.0,-14.0);  glVertex2f(-8.0,-17.0);  glVertex2f(-18.0,-17.0);  glVertex2f(-20.0,-14.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(17, 122, 101 );  glVertex2f(-16.0,-14.0);  glVertex2f(-13.0,-10.0);  glVertex2f(-9.0,-14.0);  glVertex2f(-16.0,-14.0);  glEnd();  // Ground  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(-13.0,9.0);  glVertex2f(30.0,9.0);  glVertex2f(30.0,1.0);  glVertex2f(-2.0,1.0);  glVertex2f(-13.0,9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(-5.0,1.0);  glVertex2f(30.0,1.0);  glVertex2f(30.0,-10.0);  glVertex2f(6.0,-10.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(3.0,-10.0);  glVertex2f(30.0,-10.0);  glVertex2f(30.0,-20.0);  glVertex2f(13.0,-20.0);  glVertex2f(3.0,-10.0);  glEnd();  // House-1  glBegin(GL\_POLYGON);  glColor3ub(246, 221, 204 );  glVertex2f(2.0,4.0);  glVertex2f(16.0,4.0);  glVertex2f(16.0,-3.0);  glVertex2f(2.0,-3.0);  glVertex2f(2.0,4.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub( 241, 196, 15 );  glVertex2f(2.0,4.0);  glVertex2f(9.0,8.0);  glVertex2f(16.0,4.0);  glVertex2f(2.0,4.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(128, 139, 150);  glVertex2f(8.0,-3.0);  glVertex2f(8.0,2.0);  glVertex2f(11.0,2.0);  glVertex2f(11.0,-3.0);  glVertex2f(8.0,-3.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(242, 244, 244 );  glVertex2f(3.0,-1.0);  glVertex2f(3.0,2.0);  glVertex2f(6.0,2.0);  glVertex2f(6.0,-1.0);  glVertex2f(3.0,-1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(242, 244, 244 );  glVertex2f(12.0,-1.0);  glVertex2f(12.0,2.0);  glVertex2f(15.0,2.0);  glVertex2f(15.0,-1.0);  glVertex2f(12.0,-1.0);  glEnd();  // House-2  glBegin(GL\_POLYGON);  glColor3ub(142, 68, 173 );  glVertex2f(15.0,-9.0);  glVertex2f(17.50,-4.0);  glVertex2f(27.50,-6.0);  glVertex2f(25.0,-11.0);  glVertex2f(15.0,-9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(253, 242, 233 );  glVertex2f(25.0,-11.0);  glVertex2f(27.0,-7.0);  glVertex2f(29.0,-11.0);  glVertex2f(25.0,-11.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(232, 218, 239);  glVertex2f(27.0,-7.0);  glVertex2f(27.50,-6.0);  glVertex2f(30.0,-11.0);  glVertex2f(29.0,-11.0);  glVertex2f(27.0,-7.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub( 245, 238, 248 );  glVertex2f(15.0,-9.0);  glVertex2f(25.0,-11.0);  glVertex2f(25.0,-17.50);  glVertex2f(15.0,-15.0);  glVertex2f(15.0,-9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(86, 101, 115 );  glVertex2f(19.0,-16.0);  glVertex2f(19.0,-12.0);  glVertex2f(22.0,-12.75);  glVertex2f(22.0,-16.75);  glVertex2f(19.0,-16.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(215, 189, 226 );  glVertex2f(25.0,-11.0);  glVertex2f(29.0,-11.0);  glVertex2f(29.0,-17.50);  glVertex2f(25.0,-17.50);  glVertex2f(25.0,-11.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241 );  glVertex2f(26.0,-13.50);  glVertex2f(28.25,-13.50);  glVertex2f(28.25,-15.50);  glVertex2f(26.0,-15.50);  glVertex2f(26.0,-13.50);  glEnd();  // Tree  glBegin(GL\_POLYGON);  glColor3ub(110, 44, 0 );  glVertex2f(22.0,-4.90);  glVertex2f(22.0,-1.0);  glVertex2f(23.50,-1.0);  glVertex2f(23.50,-5.20);  glVertex2f(22.0,-4.90);  glEnd();  circle1(2.45,-8.0,16.0);  circle2(2.25,21.0,0.0);  circle3(2.22,25.0,0.0);  circle4(2.22,23.0,2.00);  glFlush();  }  int main(int argc, char\*\* argv) {  glutInit(&argc, argv);  glutInitWindowSize(1300,1200);  glutCreateWindow("Village in Day");  glClearColor(1.0,1.0,1.0,1.0);  gluOrtho2D(-30.0,30.0,-20.0,20.0);  glutDisplayFunc(display);  glutMainLoop();  return 0;  }  **Code For Night:**  #include <windows.h>  #include <GL/glut.h>  #include <math.h>  void circle1(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(244, 246, 247 );  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+cX,y+cY);  }  glEnd();  }  void circle2(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void circle3(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void circle4(float radius, float cX, float cY)  {  glBegin(GL\_POLYGON);  for(int i=0;i<200;i++)  {  glColor3ub(30, 132, 73 );  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+cX,y+cY);  }  glEnd();  }  void display() {  glClear(GL\_COLOR\_BUFFER\_BIT);  glBegin(GL\_POLYGON);  glColor3ub(28, 40, 51 );  glVertex2f(-30.0,9.0);  glVertex2f(-30.0,20.0);  glVertex2f(30.0,20.0);  glVertex2f(30.0,9.0);  glVertex2f(-30.0,9.0);  glEnd();  // M-1  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-30.0,9.0);  glVertex2f(-24.0,16.0);  glVertex2f(-19.0,9.0);  glVertex2f(-30.0,9.0);  glEnd();  //M-2  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-19.0,9.0);  glVertex2f(-14.0,16.0);  glVertex2f(-8.0,9.0);  glVertex2f(-19.0,9.0);  glEnd();  //M-3  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(-8.0,9.0);  glVertex2f(-3.0,16.0);  glVertex2f(2.0,9.0);  glVertex2f(-8.0,9.0);  glEnd();  //M-4  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(2.0,9.0);  glVertex2f(7.0,16.0);  glVertex2f(12.0,9.0);  glVertex2f(2.0,9.0);  glEnd();  //M-5  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(12.0,9.0);  glVertex2f(17.0,16.0);  glVertex2f(22.0,9.0);  glVertex2f(12.0,9.0);  glEnd();  //M-6  glBegin(GL\_POLYGON);  glColor3ub(147, 81, 22 );  glVertex2f(22.0,9.0);  glVertex2f(26.0,16.0);  glVertex2f(30.0,9.0);  glVertex2f(22.0,9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(-14.0,9.0);  glVertex2f(-13.0,9.0);  glVertex2f(-2.0,1.0);  glVertex2f(-3.0,1.0);  glVertex2f(-14.0,9.0);  glEnd();  glBegin(GL\_LINES);  glColor3ub(135, 54, 0 );  glVertex2f(-3.0,1.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(-5.0,1.0);  glVertex2f(6.0,-10.0);  glVertex2f(5.0,-10.0);  glVertex2f(-5.0,0.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_LINES);  glColor3ub(135,54,0);  glVertex2f(5.00,-10.0);  glVertex2f(3.0,-10.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(135, 54, 0 );  glVertex2f(3.0,-10.0);  glVertex2f(13.0,-20.0);  glVertex2f(12.0,-20.0);  glVertex2f(3.0,-11.0);  glVertex2f(3.0,-10.0);  glEnd();  // River  glBegin(GL\_POLYGON);  glColor3ub(133, 193, 233 );  glVertex2f(-30.0,9.0);  glVertex2f(-14.0,9.0);  glVertex2f(-3.0,1.0);  glVertex2f(-5.0,1.0);  glVertex2f(-5.0,0.0);  glVertex2f(5.0,-10.0);  glVertex2f(3.0,-10.0);  glVertex2f(3.0,-11.0);  glVertex2f(12.0,-20.0);  glVertex2f(-30.0,-20.0);  glVertex2f(-30.0,9.0);  glEnd();  // Boat-1  glBegin(GL\_POLYGON);  glColor3ub(28, 40, 51 );  glVertex2f(-26.0,-1.0);  glVertex2f(-10.0,-1.0);  glVertex2f(-12.0,-4.0);  glVertex2f(-24.0,-4.0);  glVertex2f(-26.0,-1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(231, 76, 60 );  glVertex2f(-21.0,-1.0);  glVertex2f(-18.0,3.0);  glVertex2f(-14.0,-1.0);  glVertex2f(-21.0,-1.0);  glEnd();  // Boat-2  glBegin(GL\_POLYGON);  glColor3ub(28, 40, 51 );  glVertex2f(-20.0,-14.0);  glVertex2f(-6.0,-14.0);  glVertex2f(-8.0,-17.0);  glVertex2f(-18.0,-17.0);  glVertex2f(-20.0,-14.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(17, 122, 101 );  glVertex2f(-16.0,-14.0);  glVertex2f(-13.0,-10.0);  glVertex2f(-9.0,-14.0);  glVertex2f(-16.0,-14.0);  glEnd();  // Ground  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(-13.0,9.0);  glVertex2f(30.0,9.0);  glVertex2f(30.0,1.0);  glVertex2f(-2.0,1.0);  glVertex2f(-13.0,9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(-5.0,1.0);  glVertex2f(30.0,1.0);  glVertex2f(30.0,-10.0);  glVertex2f(6.0,-10.0);  glVertex2f(-5.0,1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(229, 152, 102 );  glVertex2f(3.0,-10.0);  glVertex2f(30.0,-10.0);  glVertex2f(30.0,-20.0);  glVertex2f(13.0,-20.0);  glVertex2f(3.0,-10.0);  glEnd();  // House-1  glBegin(GL\_POLYGON);  glColor3ub(246, 221, 204 );  glVertex2f(2.0,4.0);  glVertex2f(16.0,4.0);  glVertex2f(16.0,-3.0);  glVertex2f(2.0,-3.0);  glVertex2f(2.0,4.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub( 241, 196, 15 );  glVertex2f(2.0,4.0);  glVertex2f(9.0,8.0);  glVertex2f(16.0,4.0);  glVertex2f(2.0,4.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(128, 139, 150);  glVertex2f(8.0,-3.0);  glVertex2f(8.0,2.0);  glVertex2f(11.0,2.0);  glVertex2f(11.0,-3.0);  glVertex2f(8.0,-3.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(242, 244, 244 );  glVertex2f(3.0,-1.0);  glVertex2f(3.0,2.0);  glVertex2f(6.0,2.0);  glVertex2f(6.0,-1.0);  glVertex2f(3.0,-1.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(242, 244, 244 );  glVertex2f(12.0,-1.0);  glVertex2f(12.0,2.0);  glVertex2f(15.0,2.0);  glVertex2f(15.0,-1.0);  glVertex2f(12.0,-1.0);  glEnd();  // House-2  glBegin(GL\_POLYGON);  glColor3ub(142, 68, 173 );  glVertex2f(15.0,-9.0);  glVertex2f(17.50,-4.0);  glVertex2f(27.50,-6.0);  glVertex2f(25.0,-11.0);  glVertex2f(15.0,-9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(253, 242, 233 );  glVertex2f(25.0,-11.0);  glVertex2f(27.0,-7.0);  glVertex2f(29.0,-11.0);  glVertex2f(25.0,-11.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(232, 218, 239);  glVertex2f(27.0,-7.0);  glVertex2f(27.50,-6.0);  glVertex2f(30.0,-11.0);  glVertex2f(29.0,-11.0);  glVertex2f(27.0,-7.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub( 245, 238, 248 );  glVertex2f(15.0,-9.0);  glVertex2f(25.0,-11.0);  glVertex2f(25.0,-17.50);  glVertex2f(15.0,-15.0);  glVertex2f(15.0,-9.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(86, 101, 115 );  glVertex2f(19.0,-16.0);  glVertex2f(19.0,-12.0);  glVertex2f(22.0,-12.75);  glVertex2f(22.0,-16.75);  glVertex2f(19.0,-16.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(215, 189, 226 );  glVertex2f(25.0,-11.0);  glVertex2f(29.0,-11.0);  glVertex2f(29.0,-17.50);  glVertex2f(25.0,-17.50);  glVertex2f(25.0,-11.0);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(236, 240, 241 );  glVertex2f(26.0,-13.50);  glVertex2f(28.25,-13.50);  glVertex2f(28.25,-15.50);  glVertex2f(26.0,-15.50);  glVertex2f(26.0,-13.50);  glEnd();  // Tree  glBegin(GL\_POLYGON);  glColor3ub(110, 44, 0 );  glVertex2f(22.0,-4.90);  glVertex2f(22.0,-1.0);  glVertex2f(23.50,-1.0);  glVertex2f(23.50,-5.20);  glVertex2f(22.0,-4.90);  glEnd();  circle1(2.45,-8.0,16.0);  circle2(2.25,21.0,0.0);  circle3(2.22,25.0,0.0);  circle4(2.22,23.0,2.00);  glFlush();  }  int main(int argc, char\*\* argv) {  glutInit(&argc, argv);  glutInitWindowSize(1300,1200);  glutCreateWindow("Village in Night");  glClearColor(1.0,1.0,1.0,1.0);  gluOrtho2D(-30.0,30.0,-20.0,20.0);  glutDisplayFunc(display);  glutMainLoop();  return 0;  } |
| **Output Screenshot (Full Screen)-**    **Village in day light**    **Village In Night** |