**Lab Taks-4**

Submission Guidelines-

* Rename the file with your serial number only
* Must submit within time that will be discussed in class VUES
* Must include resources for all the section in the table

|  |
| --- |
| **Question- 1**  Design the given scenario |
| **Graph Plot (Picture)-** |
| **Code-**  #include <windows.h> // for MS Windows  #include <GL/glut.h> // GLUT, include glu.h and gl.h  #include <math.h>  /\* Handler for window-repaint event. Call back when the window first appears and  whenever the window needs to be re-painted. \*/  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f); // Set background color to black and opaque  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)  glLineWidth(7.5);  glBegin(GL\_POLYGON);//black circle  for(int i=0;i<200;i++)  {  glColor3f(0.0,0,0.0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 1.6619;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+1.6631418618947,y+3.2477542517588);  }  glEnd();  glBegin(GL\_POLYGON);//white circle  for(int i=0;i<200;i++)  {  glColor3f(1,1,1);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 1.5094;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+1.6631418618947,y+3.2477542517588);  }  glEnd();  glBegin(GL\_LINES); // line down to the hill  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.314796803497, 1.717804111741); // x, y  glVertex2f(4.7503626801448, 1.717804111741); // x, y  glEnd();  glBegin(GL\_LINES); // left tree  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.7000846936685, 1.7451246499556); // x, y  glVertex2f(-0.7000846936685, 2.8789269858616); // x, y  glEnd();  glBegin(GL\_LINES);//L1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.0279311522437, 2.4827791817499); // x, y  glVertex2f(-0.7000846936685, 2.8789269858616); // x, y  glEnd();  glBegin(GL\_LINES);//L2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.0279311522437, 2.1959135304966); // x, y  glVertex2f(-0.7000846936685, 2.4826721201342); // x, y  glEnd();  glBegin(GL\_LINES);//L3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.0279311522437, 1.8407465337067); // x, y  glVertex2f(-0.7000846936685, 2.1528564845004); // x, y  glEnd();  glBegin(GL\_LINES);//R1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.3449176968786, 2.5100997199645); // x, y  glVertex2f(-0.7000846936685, 2.8789269858616); // x, y  glEnd();  glBegin(GL\_LINES);//R2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.3449176968786, 2.1975343207013); // x, y  glVertex2f(-0.7000846936685, 2.4826721201342); // x, y  glEnd();  glBegin(GL\_LINES);//R3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.3449176968786, 1.8817273410286); // x, y  glVertex2f(-0.7000846936685, 2.1528564845004); // x, y  glEnd();  glBegin(GL\_LINES); // Right tree  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.2209034913981, 2.9089979773388); // x, y  glVertex2f(4.2209034913981, 1.7602744326932); // x, y  glEnd();  glBegin(GL\_LINES);//L1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8544647979448, 2.5343247065046); // x, y  glVertex2f(4.2209034913981, 2.9089979773388); // x, y  glEnd();  glBegin(GL\_LINES);//L2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8544647979448, 2.2131761886466); // x, y  glVertex2f(4.2209034913981, 2.5219728404331); // x, y  glEnd();  glBegin(GL\_LINES);//L3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8544647979448, 1.9414351350745); // x, y  glVertex2f(4.2209034913981,2.2); // x, y  glEnd();  glBegin(GL\_LINES);//R1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.5585211640179, 2.5260901291236); // x, y  glVertex2f(4.2209034913981, 2.9089979773388); // x, y  glEnd();  glBegin(GL\_LINES);//R2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.5585211640179, 2.2296453434086); // x, y  glVertex2f(4.2209034913981, 2.5219728404331); // x, y  glEnd();  glBegin(GL\_LINES);//R3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.5585211640179, 1.9167314029316); // x, y  glVertex2f(4.2209034913981,2.2); // x, y  glEnd();  glBegin(GL\_POLYGON);//black 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.7108891312295, 1.7201070288594);  glVertex2f(0.3483331632028, 2.2849421143346);  glVertex2f(0.7258866147959, 1.7574016751223);  glEnd();  glBegin(GL\_POLYGON);//2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.3483331632028, 2.2849421143346);  glVertex2f(0.7362305449766, 2.6573236008374);  glVertex2f(1.4787694897357, 1.7496747243715);  glVertex2f(0.7258866147959, 1.7574016751223);  glEnd();  glBegin(GL\_POLYGON);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.7362305449766, 2.6573236008374);  glVertex2f(0.8655296722345, 2.7400750422825);  glVertex2f(1.217223298376, 3.1331443891466);  glVertex2f(2.065425573188, 1.7625736402126);  glVertex2f(1.4787694897357, 1.7496747243715);  glEnd();  glBegin(GL\_POLYGON);//4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.217223298376, 3.1331443891466);  glVertex2f(1.6051206801498, 3.1538322495078);  glVertex2f(2.6136538727615, 1.7884334656642);  glVertex2f(2.065425573188, 1.7625736402126);  glEnd();  glBegin(GL\_POLYGON);//5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.6051206801498, 3.1538322495078);  glVertex2f(1.9206105506591, 3.4124305040237);  glVertex2f(2.7170931745679, 2.6728394961084);  glVertex2f(2.1815296861592, 2.3734631336798);  glEnd();  glBegin(GL\_POLYGON);//6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.1815296861592, 2.3734631336798);  glVertex2f(2.7170931745679, 2.6728394961084);  glVertex2f(4.2250207800885, 1.7602744326932);  glVertex2f(2.6136538727615, 1.7884334656642);  glEnd();  glBegin(GL\_POLYGON);//white 1  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(-0.4016017748931, 1.7884334656642);  glVertex2f(0.4207406744672, 2.2280504983411);  glVertex2f(0.7258866147959, 1.7574016751223);  glEnd();  glBegin(GL\_POLYGON);//2  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(0.4207406744672, 2.2280504983411);  glVertex2f(0.8138100213313, 2.6056039499343);  glVertex2f(1.4787694897357, 1.7496747243715);  glVertex2f(0.7258866147959, 1.7574016751223);  glEnd();  glBegin(GL\_POLYGON);//3  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(0.8138100213313, 2.6056039499343);  glVertex2f(1.2999747398211, 3.0555649127918);  glVertex2f(2.065425573188, 1.7625736402126);  glVertex2f(1.4787694897357, 1.7496747243715);  glEnd();  glBegin(GL\_POLYGON);//4  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(1.2999747398211, 3.0555649127918);  glVertex2f(1.6827001565045, 3.0969406335144);  glVertex2f(2.6136538727615, 1.7884334656642);  glVertex2f(2.065425573188, 1.7625736402126);  glEnd();  glBegin(GL\_POLYGON);//5  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(1.6827001565045, 3.0969406335144);  glVertex2f(1.8895787601172, 3.2831313767658);  glVertex2f(2.0137059222848, 3.1641761796885);  glVertex2f(2.6343417331228, 2.6004319848439);  glVertex2f(2.1815296861592, 2.3734631336798);  glEnd();  glBegin(GL\_POLYGON);//6  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(2.1815296861592, 2.3734631336798);  glVertex2f(2.6343417331228, 2.6004319848439);  glVertex2f(2.7326090698388, 2.5021646481279);  glVertex2f(3.9221610406117, 1.7780895354836);  glVertex2f(2.6136538727615, 1.7884334656642);  glEnd();  glBegin(GL\_POLYGON);//down white  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(0.6379632082606, 1.6487904082257);  glVertex2f(2.6188258378519, 1.6694782685869);  glVertex2f(2.6136538727615, 1.540179141329);  glVertex2f(0.6224473129896, 1.5660389667806);  glEnd();  glBegin(GL\_LINES);//sun line  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.159012868044, 4.3013767700083); // x, y  glVertex2f(1.5617402462176, 4.3013767700083); // x, y  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.4761976155108, 3.9277275223401); // x, y  glVertex2f(3.6463097332087, 3.9277275223401); // x, y  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.7908496135472, 2.9641057783537); // x, y  glVertex2f(3.42015360962, 2.954272903415); // x, y  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.1885114928599, 3.1214317773719); // x, y  glVertex2f(0.6177842521084, 3.1410975272491); // x, y  glEnd();  glBegin(GL\_LINES);//mountain straight line STRAT FROM DOWNE // 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.0543195326653, 1.87147379752); // x, y  glVertex2f(1.2467288708525, 1.8733621116035); // x, y  glEnd();  glBegin(GL\_LINES);//2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.4611401790889, 2.1440257114285); // x, y  glVertex2f(1.6101540264984, 2.1457355534634); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.6917978695882, 2.4096667699606); // x, y  glVertex2f(1.8119153866089, 2.4158905949674); // x, y  glEnd();  glBegin(GL\_LINES);//4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.0014502620969, 2.6757865842624); // x, y  glVertex2f(1.8341433330618, 2.6774964262972); // x, y  glEnd();  glBegin(GL\_LINES);//5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.2767348297054, 2.9476514678012); // x, y  glVertex2f(1.6905166021356, 2.9425219416967); // x, y  glEnd();  glBegin(GL\_LINES);//mountain curve line STRAT FROM DOWNE // 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.3204072031623, 1.7420071311657); // x, y  glVertex2f(1.5902658937848, 1.9896518442065); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.7788532542827, 2.1690711691871); // x, y  glVertex2f(1.9636736286288, 2.3889461790621); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.9430492037842, 2.390695258662); // x, y  glVertex2f(1.9147395573184, 2.5095957738182); // x, y  glEnd();  glBegin(GL\_LINES);//4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.9306783885064, 2.5074818599221); // x, y  glVertex2f(1.9505770284396, 2.7322627925005); // x, y  glEnd();  glBegin(GL\_LINES);//5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.9625546755684, 2.7337353376726); // x, y  glVertex2f(2.031058950769, 2.9046913980385); // x, y  glEnd();  glBegin(GL\_LINES);//6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.0185882159822, 2.9129767451845); // x, y  glVertex2f(1.9309122056877, 2.9947197923764); // x, y  glEnd();  glBegin(GL\_LINES);//7  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.9203647157275, 2.9934013561314); // x, y  glVertex2f(1.7924763999595, 3.0672337858531); // x, y  glEnd();  glBegin(GL\_LINES);//8  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.7979619662524, 3.0679165640454); // x, y  glVertex2f(1.8432679239643, 3.2389194154161); // x, y  glEnd();  glBegin(GL\_LINES);//M 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.3902342314114, 0.5199976271435); // x, y  glVertex2f(-1.4, 1.4); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.4, 1.4); // x, y  glVertex2f(-1.0666245394825, 0.6053946291803); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.0666245394825, 0.6053946291803); // x, y  glVertex2f(-0.7430148475537, 1.4054297008932); // x, y  glEnd();  glBegin(GL\_LINES);//4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.7430148475537, 1.4054297008932); // x, y  glVertex2f(-0.72953111039, 0.5379759433618); // x, y  glEnd();  glBegin(GL\_POLYGON);//O lower  for(int i=0;i<200;i++)  {  glColor3f(0.0,0,0.0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 0.2770;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x-0.0659522082077,y+0.8172065484871);  }  glEnd();  glBegin(GL\_POLYGON);//O uper  for(int i=0;i<200;i++)  {  glColor3f(0.0,0,0.0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 0.2770;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x-0.0583282631912,y+1.1202583628909);  }  glEnd();  glBegin(GL\_POLYGON);// upper white part  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(-0.0506548592977, 1.3485020455436);  glVertex2f(0.0278810085095, 1.334122238762);  glVertex2f(0.0989878567282, 1.2722473845355);  glVertex2f(0.1369736200835, 1.1866686154126);  glVertex2f(-0.2529032564416, 1.1897224812862);  glVertex2f(-0.2091897584634, 1.272944618959);  glVertex2f(-0.1380398389705, 1.3363345167284);  glEnd();  glBegin(GL\_POLYGON);// lower white part  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(-0.0554199299511, 0.6003263676883);  glVertex2f(-0.1419461297021, 0.6155956970561);  glVertex2f(-0.219451668134, 0.6724298990118);  glVertex2f(-0.2644715247502, 0.7770706468225);  glVertex2f(0.1401443747187, 0.7757058783698);  glVertex2f(0.1054170060567, 0.6807448356921);  glVertex2f(0.0270344486352, 0.620685473512);  glEnd();  glBegin(GL\_POLYGON);// middle white part  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(-0.4, 1.2);  glVertex2f(0.2852370718326, 1.1953429212995);  glVertex2f(0.2813400700638, 0.7686212276114);  glVertex2f(-0.4084292430211, 0.7686212276114);  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.3061885751198, 1.2065660873114);  glVertex2f(-0.3061885751198, 0.7542366696799);  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.1807183626261, 1.2065660873114);  glVertex2f(0.1807183626261, 0.7542366696799);  glEnd();  glBegin(GL\_POLYGON);//U  for(int i=0;i<200;i++)  {  glColor3f(0.0,0,0.0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 0.2770;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+0.8069894961754,y+0.8267364797577);  }  glEnd();  glBegin(GL\_POLYGON);// middle white part  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(0.4794537268311, 1.1574288994963);  glVertex2f(1.1425207524176, 1.1715667891676);  glVertex2f(1.1297966517134, 0.7417749431585);  glVertex2f(0.4907640385682, 0.7474300990271);  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.5671086427935, 1.4274625922191);  glVertex2f(0.5704623731325, 0.7345307806424);  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.0194436866079, 1.430334282154);  glVertex2f(1.032372467886, 0.7192513118592);  glEnd();  glBegin(GL\_LINES);//NNNNNNN 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.4323614148567, 0.5469651014709); // x, y  glVertex2f(1.4458451520204, 1.4189134380569); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.4458451520204, 1.4189134380569); // x, y  glVertex2f(1.9177759527499, 0.5694379967438); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.9177759527499, 0.5694379967438); // x, y  glVertex2f(1.9177759527499, 1.4368917542752); // x, y  glEnd();  glBegin(GL\_LINES);//TTT 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.231693508324, 1.4017351694352); // x, y  glVertex2f(2.8, 1.4); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(2.5152959204644, 1.3950882975576); // x, y  glVertex2f(2.5200495460619, 0.5514596805255); // x, y  glEnd();  glBegin(GL\_LINES);//AAAA 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.0161690866235, 0.5732703063713); // x, y  glVertex2f(3.2872842815822, 1.4410879003425); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.2872842815822, 1.4410879003425); // x, y  glVertex2f(3.5794311980376, 0.5605032230834); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.1048759767118, 0.826128724512); // x, y  glVertex2f(3.4869208440805, 0.826128724512); // x, y  glEnd();  glBegin(GL\_LINES);//IIII 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9370642428932, 1.4527921631623); // x, y  glVertex2f(3.9310897165329, 0.5596004723057); // x, y  glEnd();  glBegin(GL\_LINES);//NNNNNNN 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.3881409830917, 0.5715495250262); // x, y  glVertex2f(4.3761919303712, 1.4318813209015); // x, y  glEnd();  glBegin(GL\_LINES);// 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.3761919303712, 1.4318813209015); // x, y  glVertex2f(4.8631158287312, 0.5745367882063); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.8631158287312, 0.5745367882063); // x, y  glVertex2f(4.8600532730003, 1.4310613592782); // x, y  glEnd();  glFlush(); // Render now  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInit(&argc, argv); // Initialize GLUT  glutCreateWindow("OpenGL Setup Test");  gluOrtho2D(-4,7,0,6); // Create a window with the given title  glutInitWindowSize(320, 320);// Set the window's initial width & height  glutDisplayFunc(display);// Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 2**  Design the given scenario  Drawing Lighthouse Line Stock Illustrations – 1,639 Drawing Lighthouse Line  Stock Illustrations, Vectors & Clipart - Dreamstime |
| **Graph Plot (Picture)-** |
| **Code-**  #include <windows.h> // for MS Windows  #include <GL/glut.h> // GLUT, include glu.h and gl.h  #include <math.h>  /\* Handler for window-repaint event. Call back when the window first appears and  whenever the window needs to be re-painted. \*/  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f); // Set background color to black and opaque  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)  glLineWidth(7.5);  glBegin(GL\_POLYGON);//black circle  for(int i=0;i<200;i++)  {  glColor3f(0.0,0,0.0);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r= 3.3545;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+3.913401919523,y-1.5603051494142);  }  glEnd();  glBegin(GL\_POLYGON);//white circle  for(int i=0;i<200;i++)  {  glColor3f(1,1,1);  float pi=3.1416;  float A=(i\*2\*pi)/200;  float r=3.3606;  float x = r \* cos(A);  float y = r \* sin(A);  glVertex2f(x+3.911942754973,y-1.6589597723313);  }  glEnd();  glBegin(GL\_POLYGON);//white 1  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(0.8682102049834, 0.4384082214663);  glVertex2f(2.0636034308174, 1.1987407920517);  glVertex2f(2.7911740897954, -1.9479302353991);  glVertex2f(0.1022286635921, -1.867191740346);  glEnd();  glBegin(GL\_POLYGON);//white 2  glColor3f(1.0f, 1.0f, 1.0f);  glVertex2f(6.2244251955664, 0.9234596953082);  glVertex2f(7.35725103565, 0.3745286555147);  glVertex2f(7.5814230752361, -1.928329569324);  glVertex2f(5.7065296532436, -1.938519207487);  glEnd();  glBegin(GL\_LINES);// DOWNE line // 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-1.5609092084662, 0.6392666889761); // x, y  glVertex2f(0.8091646609467, 0.6616964101377); // x, y  glEnd();  glBegin(GL\_LINES);//2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.8091646609467, 0.6616964101377); // x, y  glVertex2f(2.1209814488765, 1.2417802528474); // x, y  glEnd();  glBegin(GL\_LINES);//3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(6.1960391314912, 0.8595203001907); // x, y  glVertex2f(6.8123839565026, 0.6316225110711); // x, y  glEnd();  glBegin(GL\_LINES);//4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(6.8123839565026, 0.6316225110711); // x, y  glVertex2f(9.4595604556176, 0.6916027050199); // x, y  glEnd();  glBegin(GL\_LINES);//tower main left line  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.590773832987, 1.754690941194); // x, y  glVertex2f(3.6964001478503, 4.0095831411012); // x, y  glEnd();  glBegin(GL\_LINES);//tower main right line  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.229124170639, 1.754690941194); // x, y  glVertex2f(4.1189054073034, 4.0095831411012); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper line  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6964001478503, 4.0095831411012); // x, y  glVertex2f(4.1189054073034, 4.0095831411012); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper middle box L1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6964001478503, 4.0095831411012); // x, y  glVertex2f(3.6262066118793, 4.1110985365097); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper middle box L2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6262066118793, 4.1110985365097); // x, y  glVertex2f(3.6183318709343, 4.2100952798181);  glEnd();  glBegin(GL\_LINES);//tower main upper middle box L3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6183318709343, 4.2100952798181);  glVertex2f(3.6194568339264, 4.3169667640714);  glEnd();  glBegin(GL\_LINES);//tower main upper middle LINE L1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6262066118793, 4.1110985365097); // x, y  glVertex2f(4.1853132189731, 4.1110985365097);  glEnd();  glBegin(GL\_LINES);//tower main upper middle LINE L2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6183318709343, 4.2100952798181); // x, y  glVertex2f(4.1853132189731, 4.2089703168259);  glEnd();  glBegin(GL\_LINES);//tower main upper middle LINE L3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6194568339264, 4.3169667640714); // x, y  glVertex2f(4.184188255981, 4.3180917270636);  glEnd();  glBegin(GL\_LINES);//tower main upper middle box R1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1189054073034, 4.0095831411012); // x, y  glVertex2f(4.1853132189731, 4.1110985365097); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper middle box R2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1853132189731, 4.1110985365097); // x, y  glVertex2f(4.1853132189731, 4.2089703168259);  glEnd();  glBegin(GL\_LINES);//tower main upper middle box R3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1853132189731, 4.2089703168259);  glVertex2f(4.184188255981, 4.3180917270636);  glEnd();  glBegin(GL\_LINES);//tower main upper LINE 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.7160172941043, 4.6516260355311); // x, y  glVertex2f(3.6640905360832, 4.635191071653);  glEnd();  glBegin(GL\_LINES);//tower main upper LINE 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.6640905360832, 4.635191071653); // x, y  glVertex2f(3.8709312913476, 4.8354445417179);  glEnd();  glBegin(GL\_LINES);//tower main upper LINE 3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8709312913476, 4.8354445417179); // x, y  glVertex2f(3.9, 5);  glEnd();  glBegin(GL\_LINES);//tower main upper LINE 4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9, 5); // x, y  glVertex2f(3.9460263426219, 4.820952514279); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper LINE 5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9460263426219, 4.820952514279); // x, y  glVertex2f(4.1489147267666, 4.6391434427728);  glEnd();  glBegin(GL\_LINES);//tower main upper middle LINE 6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1489147267666, 4.6391434427728);  glVertex2f(4.1, 4.65);  glEnd();  glBegin(GL\_LINES);//tower main upper middle LINE 7  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1, 4.65);  glVertex2f(3.7160172941043, 4.6516260355311);  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.7142450534603, 4.3199573179132); // x, y  glVertex2f(3.7160172941043, 4.6516260355311);  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8080462648546, 4.6494627298424); // x, y  glVertex2f(3.8076933534323, 4.3282235813562); // x, y  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE 3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8076933534323, 4.3282235813562); // x, y  glVertex2f(3.9960897101381, 4.3295410383961);  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE 4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9960897101381, 4.3295410383961);  glVertex2f(4.0001377270697, 4.6488918785431);  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE 5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.1, 4.65);  glVertex2f(4.1, 4.32);  glEnd();  glBegin(GL\_LINES);//tower main upper AND middle C0NNECTOR LINNE M6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9978543218725, 4.5818168508751);  glVertex2f(3.8091879674532, 4.5815314252255);  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8592614122913, 3.6659181005849); // x, y  glVertex2f(3.8572663876715, 3.8534504148505);  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8572663876715, 3.8534504148505); // x, y  glVertex2f(3.9111320524073, 3.9000009893136); // x, y  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9111320524073, 3.9000009893136); // x, y  glVertex2f(3.9623376843167, 3.8541154230572);  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9623376843167, 3.8541154230572);  glVertex2f(3.9610076679035, 3.6652530923783);  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9610076679035, 3.6652530923783);  glVertex2f(3.9137920852338, 3.6399827805269);  glEnd();  glBegin(GL\_LINES);//tower upper WINDOW LINE 6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9137920852338, 3.6399827805269);  glVertex2f(3.8592614122913, 3.6659181005849);  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8695500235334, 2.3354953200141); // x, y  glVertex2f(3.8668697711832, 2.5107403249711);  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.8668697711832, 2.5107403249711); // x, y  glVertex2f(3.9182070747175, 2.5408346063533); // x, y  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9182070747175, 2.5408346063533); // x, y  glVertex2f(3.9624633708678, 2.5133957027401);  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9624633708678, 2.5133957027401);  glVertex2f(3.9651187486368, 2.33371514037);  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9651187486368, 2.33371514037);  glVertex2f(3.9201704534532, 2.3172157203208);  glEnd();  glBegin(GL\_LINES);//tower LOWER WINDOW LINE 6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(3.9201704534532, 2.3172157203208);  glVertex2f(3.8695500235334, 2.3354953200141);  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 1  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.0847220977172, 4.9854571318171); // x, y  glVertex2f(3.437022989647, 4.5616744646026);  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 2  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.5749545415027, 4.4142717977454); // x, y  glVertex2f(3.3571798784327, 4.3712793532454); // x, y  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 3  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(1.0355878754315, 3.7079673523879); // x, y  glVertex2f(3.351038100647, 4.162458908531);  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 4  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.4759659937777, 4.5589389993202);  glVertex2f(6.8, 5);  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 5  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.5301433451216, 4.3873773867313);  glVertex2f(7.3112473807729, 4.4234956209606);  glEnd();  glBegin(GL\_LINES);//tower OUTER LINE 6  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(4.4940251108923, 4.1977566570278);  glVertex2f(6.8371955565142, 3.737249170605);  glEnd();  glFlush(); // Render now  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInit(&argc, argv); // Initialize GLUT  glutCreateWindow("OpenGL Setup Test");  gluOrtho2D(-2,10,-2,6); // Create a window with the given title  glutInitWindowSize(320, 320);// Set the window's initial width & height  glutDisplayFunc(display);// Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 3**  Design the given scenario  Traffic Light Drawing Vector Images (over 2,300) |
| **Graph Plot (Picture)-** |
| **Code-** |
| **Output Screenshot (Full Screen)-** |