**Lab Taks-2**

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-2
* Must include resources for all the section in the table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Question- 1**  Draw a Rainbow Flag   |  | | --- | |  | |  | |  | |  | |  | |  | |  | |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <windows.h> // for MS Windows**  **#include <GL/glut.h> // GLUT, include glu.h and gl.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(10.0);**  **// Draw a Red 1x1 Square centered at origin**  **glColor3f(1.0f,0.0f,0.0f);//For RED**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,1.0f);**  **glVertex2f(5.0f,1.0f);**  **glVertex2f(5.0f,2.0f);**  **glVertex2f(1.0f,2.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,0.0f);//For Yellow**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,2.0f);**  **glVertex2f(5.0f,2.0f);**  **glVertex2f(5.0f,3.0f);**  **glVertex2f(1.0f,3.0f);**  **glEnd();**  **glColor3ub(255, 165, 0);//For Orange**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,3.0f);**  **glVertex2f(5.0f,3.0f);**  **glVertex2f(5.0f,4.0f);**  **glVertex2f(1.0f,4.0f);**  **glEnd();**  **glColor3f(0.0f,1.0f,0.0f);//For Green**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,4.0f);**  **glVertex2f(5.0f,4.0f);**  **glVertex2f(5.0f,5.0f);**  **glVertex2f(1.0f,5.0f);**  **glEnd();**  **glColor3ub(0,191,255);//For Sky**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,5.0f);**  **glVertex2f(5.0f,5.0f);**  **glVertex2f(5.0f,6.0f);**  **glVertex2f(1.0f,6.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,1.0f);//For Blue**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,6.0f);**  **glVertex2f(5.0f,6.0f);**  **glVertex2f(5.0f,7.0f);**  **glVertex2f(1.0f,7.0f);**  **glEnd();**  **glColor3ub(135,0,130);//For Purple**  **glBegin(GL\_QUADS);**  **glVertex2f(1.0f,7.0f);**  **glVertex2f(5.0f,7.0f);**  **glVertex2f(5.0f,8.0f);**  **glVertex2f(1.0f,8.0f);**  **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("Rainbow Flag"); // Create a window with the given title**  **glutInitWindowSize(150, 150);**  **gluOrtho2D(-10.0,10.0,-10.0,10.0); //resize the axis size**  **glutDisplayFunc(display); // Register display callback handler for window re-paint**  **glutMainLoop(); // Enter the event-processing loop**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 2**  Draw 8X8 Chess Board |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <windows.h> // for MS Windows**  **#include <GL/glut.h> // GLUT, include glu.h and gl.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(0.0f, 5.0f, .0f, 0.0f); // Set background color to black and opaque**  **glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)**  **glLineWidth(10.0);**  **// Draw a Red 1x1 Square centered at origin**  **//first column**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(0.0f,0.0f);**  **glVertex2f(2.0f,0.0f);**  **glVertex2f(2.0f,2.0f);**  **glVertex2f(0.0f,2.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(0.0f,2.0f);**  **glVertex2f(2.0f,2.0f);**  **glVertex2f(2.0f,4.0f);**  **glVertex2f(0.0f,4.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(0.0f,4.0f);**  **glVertex2f(2.0f,4.0f);**  **glVertex2f(2.0f,6.0f);**  **glVertex2f(0.0f,6.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(0.0f,6.0f);**  **glVertex2f(2.0f,6.0f);**  **glVertex2f(2.0f,8.0f);**  **glVertex2f(0.0f,8.0f);**  **glEnd();**  **//second column**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(2.0f,0.0f);**  **glVertex2f(4.0f,0.0f);**  **glVertex2f(4.0f,2.0f);**  **glVertex2f(2.0f,2.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(2.0f,2.0f);**  **glVertex2f(4.0f,2.0f);**  **glVertex2f(4.0f,4.0f);**  **glVertex2f(2.0f,4.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(2.0f,4.0f);**  **glVertex2f(4.0f,4.0f);**  **glVertex2f(4.0f,6.0f);**  **glVertex2f(2.0f,6.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(2.0f,6.0f);**  **glVertex2f(4.0f,6.0f);**  **glVertex2f(4.0f,8.0f);**  **glVertex2f(2.0f,8.0f);**  **glEnd();**  **//Third column**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(4.0f,0.0f);**  **glVertex2f(6.0f,0.0f);**  **glVertex2f(6.0f,2.0f);**  **glVertex2f(4.0f,2.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(4.0f,2.0f);**  **glVertex2f(6.0f,2.0f);**  **glVertex2f(6.0f,4.0f);**  **glVertex2f(4.0f,4.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(4.0f,4.0f);**  **glVertex2f(6.0f,4.0f);**  **glVertex2f(6.0f,6.0f);**  **glVertex2f(4.0f,6.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(4.0f,6.0f);**  **glVertex2f(6.0f,6.0f);**  **glVertex2f(6.0f,8.0f);**  **glVertex2f(4.0f,8.0f);**  **glEnd();**  **//Fourth Column**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(6.0f,0.0f);**  **glVertex2f(8.0f,0.0f);**  **glVertex2f(8.0f,2.0f);**  **glVertex2f(6.0f,2.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(6.0f,2.0f);**  **glVertex2f(8.0f,2.0f);**  **glVertex2f(8.0f,4.0f);**  **glVertex2f(6.0f,4.0f);**  **glEnd();**  **glColor3f(1.0f,1.0f,1.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(6.0f,4.0f);**  **glVertex2f(8.0f,4.0f);**  **glVertex2f(8.0f,6.0f);**  **glVertex2f(6.0f,6.0f);**  **glEnd();**  **glColor3f(0.0f,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glVertex2f(6.0f,6.0f);**  **glVertex2f(8.0f,6.0f);**  **glVertex2f(8.0f,8.0f);**  **glVertex2f(6.0f,8.0f);**  **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("Chess Board"); // Create a window with the given title**  **glutInitWindowSize(320, 320);**  **gluOrtho2D(-15.0,15.0,-15.0,15.0); //resize the axis size**  **glutDisplayFunc(display); // Register display callback handler for window re-paint**  **glutMainLoop(); // Enter the event-processing loop**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 3**  Create the batman logo given below- |
| **Graph Plot (Picture)-**  **(Not Needed)** |
| **Code-**  #include <windows.h>  #include <GL/glut.h>  void Display(void)  {  glClear (GL\_COLOR\_BUFFER\_BIT);  glClearColor(0.0f, 5.0f, .0f, 0.0f);  glColor3ub (255, 255, 255);  glBegin(GL\_QUADS);  glVertex2i(0, 0);  glVertex2i(640, 0);  glVertex2i(640, 480);  glVertex2i(0, 480);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(427, 440);  glVertex2i(427, 460);  glVertex2i(212, 460);  glVertex2i(212, 440);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(212, 414);  glVertex2i(212, 440);  glVertex2i(148, 440);  glVertex2i(148, 414);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(148, 390);  glVertex2i(148, 414);  glVertex2i(118, 414);  glVertex2i(118, 390);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(118, 362);  glVertex2i(118,390);  glVertex2i(90, 390);  glVertex2i(90, 362);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(90, 337);  glVertex2i(90,362);  glVertex2i(55, 362);  glVertex2i(55, 337);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(55, 156);  glVertex2i(55,337);  glVertex2i(25, 337);  glVertex2i(25, 156);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(82, 130);  glVertex2i(82,156);  glVertex2i(55, 156);  glVertex2i(55, 130);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(114, 105);  glVertex2i(114,130);  glVertex2i(82, 130);  glVertex2i(82, 105);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(145, 80);  glVertex2i(145,105);  glVertex2i(114, 105);  glVertex2i(114, 80);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(210, 55);  glVertex2i(210,80);  glVertex2i(145, 80);  glVertex2i(145, 55);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(427, 30);  glVertex2i(427,55);  glVertex2i(212, 55);  glVertex2i(212, 30);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(490, 55);  glVertex2i(490,80);  glVertex2i(427, 80);  glVertex2i(427, 55);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(520, 80);  glVertex2i(520,102);  glVertex2i(490, 102);  glVertex2i(490, 80);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(550, 102);  glVertex2i(550,128);  glVertex2i(520, 128);  glVertex2i(520, 102);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(580, 128);  glVertex2i(580,156);  glVertex2i(550, 156);  glVertex2i(550, 128);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(610, 156);  glVertex2i(610,337);  glVertex2i(580, 337);  glVertex2i(580, 156);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(580, 337);  glVertex2i(580,360);  glVertex2i(555, 360);  glVertex2i(555, 337);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(555, 360);  glVertex2i(555,385);  glVertex2i(523, 385);  glVertex2i(523, 360);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(523, 385);  glVertex2i(523,410);  glVertex2i(490, 410);  glVertex2i(490, 385);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(490, 410);  glVertex2i(490,440);  glVertex2i(427, 440);  glVertex2i(427, 410);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(580, 156);  glVertex2i(580,337);  glVertex2i(555, 337);  glVertex2i(550, 156);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(550, 128);  glVertex2i(556,360);  glVertex2i(523, 360);  glVertex2i(520, 128);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(520, 102);  glVertex2i(524,385);  glVertex2i(490, 385);  glVertex2i(490, 102);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(490, 80);  glVertex2i(490,410);  glVertex2i(427, 410);  glVertex2i(427, 80);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(427, 440);  glVertex2i(427,55);  glVertex2i(212, 55);  glVertex2i(212, 440);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(212, 80);  glVertex2i(212,414);  glVertex2i(145, 414);  glVertex2i(145, 80);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(145, 105);  glVertex2i(148,390);  glVertex2i(118, 390);  glVertex2i(114, 105);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(115, 130);  glVertex2i(118,362);  glVertex2i(90, 362);  glVertex2i(82, 130);  glEnd();  glColor3ub (255, 255, 0);  glBegin(GL\_QUADS);  glVertex2i(83, 156);  glVertex2i(90,337);  glVertex2i(55, 337);  glVertex2i(55, 156);  glEnd();  //BODY 1  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(550, 182);  glVertex2i(550, 306);  glVertex2i(523, 306);  glVertex2i(523, 182);  glEnd();  //BODY 2  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(523, 157);  glVertex2i(523, 334);  glVertex2i(490, 334);  glVertex2i(490, 157);  glEnd();  //BODY 3  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(490, 130);  glVertex2i(490, 360);  glVertex2i(460, 360);  glVertex2i(460, 130);  glEnd();  //BODY 4  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(460, 180);  glVertex2i(460, 310);  glVertex2i(430, 310);  glVertex2i(430, 180);  glEnd();  //BODY 5  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(430, 155);  glVertex2i(430, 280);  glVertex2i(400, 280);  glVertex2i(400, 155);  glEnd();  //BODY 6  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(400, 205);  glVertex2i(400, 310);  glVertex2i(365, 310);  glVertex2i(365, 205);  glEnd();  //BODY 7  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(365, 155);  glVertex2i(365, 360);  glVertex2i(275, 360);  glVertex2i(275, 155);  glEnd();  //BODY 8  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(275, 205);  glVertex2i(275, 310);  glVertex2i(245, 310);  glVertex2i(245, 205);  glEnd();  //BODY 9  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(245, 155);  glVertex2i(245, 280);  glVertex2i(215, 280);  glVertex2i(215, 155);  glEnd();  //BODY 10  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(215, 180);  glVertex2i(215, 310);  glVertex2i(185, 310);  glVertex2i(185, 180);  glEnd();  //BODY 11  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(185, 130);  glVertex2i(185, 360);  glVertex2i(155, 360);  glVertex2i(155, 130);  glEnd();  //BODY 12  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(155, 157);  glVertex2i(155, 334);  glVertex2i(125, 334);  glVertex2i(125, 157);  glEnd();  //BODY 13  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(125, 182);  glVertex2i(125, 306);  glVertex2i(95, 306);  glVertex2i(95, 182);  glEnd();  //BODY 14  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(350, 105);  glVertex2i(350, 360);  glVertex2i(290, 360);  glVertex2i(290, 105);  glEnd();  //Body 15  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(335, 80);  glVertex2i(335, 105);  glVertex2i(305, 105);  glVertex2i(305, 80);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(365, 360);  glVertex2i(365, 385);  glVertex2i(335, 385);  glVertex2i(335, 360);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(305, 360);  glVertex2i(305, 385);  glVertex2i(275, 385);  glVertex2i(275, 360);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(460, 345);  glVertex2i(460, 370);  glVertex2i(430, 370);  glVertex2i(430, 345);  glEnd();  glColor3ub (0, 0, 0);  glBegin(GL\_QUADS);  glVertex2i(210, 345);  glVertex2i(210, 370);  glVertex2i(180, 370);  glVertex2i(180, 345);  glEnd();  glFlush ();  }  void myInit (void)  {  glClearColor(0.0,0.0,0.0,0.0);  glMatrixMode(GL\_PROJECTION);  glLoadIdentity();  gluOrtho2D(0.0, 640.0, 0.0, 480.0);  }  int main(int argc, char\*\* argv)  {  glutInit(&argc, argv);  glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);  glutInitWindowSize (720, 520);  glutInitWindowPosition (150, 200);  glutCreateWindow ("Batman logo");  glutDisplayFunc(Display);  myInit ();  glutMainLoop();  return 0;  } |
| **Output Screenshot (Full Screen)-** |