**Lab Taks-2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Question- 1**  Draw a Rainbow Flag   |  | | --- | |  | |  | |  | |  | |  | |  | |  | |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <GL/glut.h>**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glBegin(GL\_QUADS);**  **// Red**  **glColor3f(1.0f, 0.0f, 0.0f);**  **glVertex2f(5,10);**  **glVertex2f(95,10);**  **glVertex2f(95,20);**  **glVertex2f(5,20);**  **//glEnd();**  **// Yellow**  **glColor3f(1.0f, 1.0f, 0.0f);**  **glVertex2f(5,20);**  **glVertex2f(95,20);**  **glVertex2f(95,30);**  **glVertex2f(5,30);**  **// Orange**  **glColor3f(1.0f, 0.5f, 0.0f);**  **glVertex2f(5,30);**  **glVertex2f(95,30);**  **glVertex2f(95,40);**  **glVertex2f(5,40);**  **// Green**  **glColor3f(0.0f, 1.0f, 0.0f);**  **glVertex2f(5,40);**  **glVertex2f(95,40);**  **glVertex2f(95,50);**  **glVertex2f(5,50);**  **// cyan**  **glColor3f(0.0f, 1.0f,1.0f);**  **glVertex2f(5,50);**  **glVertex2f(95,50);**  **glVertex2f(95,60);**  **glVertex2f(5,60);**  **// Blue**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(5,60);**  **glVertex2f(95,60);**  **glVertex2f(95,70);**  **glVertex2f(5,70);**  **// Purple**  **glColor3f(0.5f, 0.0f, 0.5f);**  **glVertex2f(5,70);**  **glVertex2f(95,70);**  **glVertex2f(95,80);**  **glVertex2f(5,80);**  **glEnd();**  **glFlush();**  **}**  **void init() {**  **glClearColor(1.0, 1.0, 1.0, 0.0);**  **glMatrixMode(GL\_PROJECTION);**  **glLoadIdentity();**  **gluOrtho2D(0.0, 100.0,0.0, 100.0);**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitDisplayMode(GLUT\_RGB | GLUT\_SINGLE);**  **glutInitWindowSize(640, 480);**  **glutCreateWindow("Rainbow Flag");**  **glutDisplayFunc(display);**  **init();**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 2**  Draw 8X8 Chess Board |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <GL/glut.h>**  **void drawChessboard() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **int row, col;**  **bool isWhite = true;**  **for (row = 0; row < 8; ++row) {**  **isWhite = !isWhite;**  **for (col = 0; col < 8; ++col) {**  **if (isWhite) {**  **glColor3f(1.0, 1.0, 1.0);**  **} else {**  **glColor3f(0.0, 0.0, 0.0);**  **}**  **glBegin(GL\_QUADS);**  **glVertex2f(col,row);**  **glVertex2f(col+1,row);**  **glVertex2f(col+1,row+1);**  **glVertex2f(col,row+1);**  **glEnd();**  **isWhite=!isWhite;**  **}**  **}**  **glFlush();**  **}**  **void display() {**  **glClearColor(0.0, 0.0, 0.0, 1.0);**  **glMatrixMode(GL\_PROJECTION);**  **glLoadIdentity();**  **glOrtho(0, 8, 0, 8, -1, 1);**  **glMatrixMode(GL\_MODELVIEW);**  **glLoadIdentity();**  **drawChessboard();**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);**  **glutInitWindowSize(600,600);**  **glutCreateWindow("Chessboard");**  **glutDisplayFunc(display);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 3**  Create the batman logo given below- |
|  |
| **Code-**  **#include <windows.h>**  **#include <stdio.h>**  **#include <GL/glut.h>**  **#include <cmath>**  **using namespace std;**  **int m=0;**  **void draw\_line (int x, int y, int sizes, int num)**  **{**  **for (int a=x,b=y,c=0; c<num;c++,a+=sizes)**  **{**  **glBegin(GL\_POLYGON);**  **g1Vertex2i(a,b);**  **g1Vertex2i(a+sizes,b);**  **g1Vertex2i(a+sizes,b+sizes);**  **g1Vertex2i(a,b+sizes);**  **glEnd();**  **}**  **}**  **void vertical\_line (int x, int y, int sizes, int num)**  **{**  **for (int a=x,b=y,c=0; c<num;c++,b+=sizes)**  **{**  **glBegin(GL\_POLYGON);**  **g1Vertex2i(a,b);**  **g1Vertex2i(a+sizes,b);**  **g1Vertex2i(a+sizes,b+sizes);**  **g1Vertex2i(a,b+sizes);**  **glEnd();**  **}**  **}**  **void temp (void)**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glPointSize(5);**  **int sizes=25,blocks=17;**  **int x=9\*sizes,y=3\*sizes;**  **glColor3ub(0,0,0);**  **draw\_line(x,y,sizes,blocks);**  **x=9\*sizes,y=3\*sizes;**  **blocks;**  **for**  **(int a=0;a<=5;a++)**  **{**  **glColor3ub(250,250,0);**  **draw\_line(x,y,sizes,blocks);**  **glColor3ub(0,0,0);**  **draw\_line(x-sizes,y,sizes,1);**  **draw\_line(x+(blocks+sizes),y,sizes1);**  **x=x-sizes;**  **y=y+sizes;**  **blocks+=2;**  **}**  **x=3\*sizes,y=10\*sizes;**  **for (int a=0;a<7;a++)**  **{**  **glColor3ub(250,250,0);**  **draw\_line(x,y,sizes,29);**  **glColor3ub(0,0,0);**  **draw\_line(x-sizes,y,sizes,1);**  **glColor3ub(0,0,0);**  **draw\_line(x+(29\*sizes),y,sizes,1);**  **y+=sizes;**  **}**  **x=4\*sizes,y=17\*sizes,blocks=27;**  **for (int a=0; a<=5;a++)**  **{**  **glColor3ub(250,250,0);**  **draw\_line(x,y,sizes,29);**  **glColor3ub(0,0,0);**  **draw\_line(x-sizes,y,sizes,1);**  **draw\_line(x+(blocks\*sizes),y,sizes,1);**  **y+=sizes;**  **x+=sizes;**  **blocks-=2;**  **}**  **x=9\*sizes;**  **glColor3ub(0,0,0);**  **draw\_line(x,y,sizes,blocks+2);**  **x=9\*sizes,y=11\*sizes;**  **blocks=5;**  **for(int a=0;a<6;a++)**  **{**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y-=sizes;**  **blocks+=2;**  **}**  **blocks-=2;**  **y+=sizes;**  **vertical\_line(x,y,sizes,blocks);**  **y-=sizes;**  **x+=sizes;**  **blocks=12;**  **vertical\_line(x,y,sizes,blocks);**  **y=20\*sizes;**  **vertical\_line(x,y,sizes,1);**  **x+=sizes;**  **y=7\*sizes;**  **blocks=9;**  **for (int a=0;a<2;a++)**  **{**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y+=sizes;**  **blocks--;**  **}**  **blocks++;**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y-=sizes;**  **blocks=14;**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y-=(2\*sizes);**  **blocks=15;**  **vertical\_line(a,y,sizes,blocks);**  **x+=sizes;**  **y-=(sizes);**  **blocks=16;**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y+=(sizes);**  **blocks=15;**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y+=(2\*sizes);**  **blocks=14;**  **vertical\_line(x,y,sizes,blocks);**  **y=6\*sizes;**  **x=24\*sizes;**  **blocks=16;**  **for (int a=0;a<=6;a++)**  **{**  **vertical\_line(x,y,sizes,blocks);**  **x+=sizes;**  **y+=sizes;**  **blocks-=2;**  **}**  **y=6\*sizes;**  **x=22\*sizes;**  **blocks=12;**  **vertical\_line(x,y,sizes,blocks);**  **y=21\*sizes;**  **vertical\_line(x,y,sizes,1);**  **x-=sizes;**  **y-=(14\*sizes);**  **vertical\_line(x,y,sizes,9);**  **x-=sizes;**  **y+=sizes;**  **vertical\_line(x,y,sizes,9);**  **glFlush();**  **}**  **void myDisplay(void) {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glPointSize(5.0);**  **glFlush();**  **}**  **void initGL(void)**  **{**  **glClearColor(1.0f, 1.0f, 1.0f, 0.0f);**  **glMatrixMode(GL\_MODELVIEW);**  **glLoadIdentity();**  **gluOrtho2D(0.0, 900.0, 0.0, 700.0 )**  **}**  **main(int argc, char\*\* argv)**  **{**  **glutInit(&argc, argv);**  **glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);**  **glutInitWindowSize (900, 700);**  **glutInitWindowPosition (200, 150);**  **glutCreateWindow ("Batman");**  **glutDisplayFunc(myDisplay);**  **glutDisplayFunc(temp);**  **myInit ();**  **glutMainLoop();**  **}** |
| **Output Screenshot (Full Screen)-** |