**Lab Taks-6**

|  |
| --- |
| **Question-**  Develop an animation that will change the background color of the window after 20ms. Use at least two different colors. |
| **Code-**  **#include<cstdio>**  **#include <GL/gl.h>**  **#include <GL/glut.h>**  **GLfloat colorchange = 0.0f;**  **GLfloat increment = 0.05f;**  **void update(int value) {**  **if(colorchange >1.0)**  **colorchange = 0.0f;**  **colorchange += increment;**  **glutPostRedisplay();**  **glutTimerFunc(100, update, 0);**  **}**  **void init() {**  **gluOrtho2D(-2,2,-2,2);**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glClearColor(colorchange, colorchange , colorchange , 1.0f);**  **glPopMatrix();**  **glFlush();**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitWindowSize(480, 320);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Two Box Translation Animation");**  **glutDisplayFunc(display);**  **init();**  **glutTimerFunc(100, update, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Develop a code that will have four different objects (keep it simple). The objects will move to the left, right, up and down in a loop. |
| **Code-**  **#include<cstdio>**  **#include <GL/gl.h>**  **#include <GL/glut.h>**  **#include <math.h>**  **GLfloat position = 0.0f;**  **GLfloat position1 = 0.0f;**  **GLfloat speed = 0.1f;**  **void dis();**  **void display();**  **void update(int value) {**  **if(position <-1.5)**  **position = 0.0f;**  **position -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update,0);**  **}**  **void update1(int value) {**  **if(position1 >1.5)**  **position1 = -0.0f;**  **position1 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void init() {**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **}**  **void disback(int val)**  **{**  **glutDisplayFunc(display);**  **}**  **void display7()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(0.0f,position1, 0.0f);**  **glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin**  **for(int i=0;i<200;i++)**  **{**  **glColor3f(1.0,1.0,1.0);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.10;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,disback,0);**  **glFlush();**  **}**  **void display6(int val) {**  **glutDisplayFunc(display7);**  **}**  **void display5()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(0.0f,position, 0.0f);**  **glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin**  **for(int i=0;i<200;i++)**  **{**  **glColor3f(1.0,0,1.0);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.10;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,display6,0);**  **glFlush();**  **}**  **void display4(int val) {**  **glutDisplayFunc(display5);**  **}**  **void display3()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(position1,0.0f, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 1.0f, 0.0f);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f( 0.2f, -0.2f);**  **glVertex2f( 0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,display4,0);**  **glFlush();**  **}**  **void display2(int val) {**  **glutDisplayFunc(display3);**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glLoadIdentity();**  **glPushMatrix();**  **glTranslatef(position,0.0f, 0.0f);**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(0.2f,-0.2f);**  **glVertex2f(0.5f, 0.0f);**  **glVertex2f(0.2f, 0.2f);**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,display2,0);**  **glFlush();**  **}**  **void dis()**  **{**  **glutDisplayFunc(display);**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitWindowSize(480, 300);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Fore moving object");**  **glutDisplayFunc(dis);**  **init();**  **glutTimerFunc(100, update, 0);**  **glutTimerFunc(100, update1, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-**  **Down**    **Up**  **Right**    **Left** |

|  |
| --- |
| **Question-**  Develop a code that will have four different objects (keep it simple). Four different keys will be dedicated for the objects. The objects will move to the left, right, up and down in a loop as the keys are pressed individually. |
| **Code-**  **#include <iostream>**  **#include <stdlib.h>**  **#include<GL/gl.h>**  **#include <GL/glut.h>**  **#include<math.h>**  **#include<cstring>**  **using namespace std;**  **GLfloat position = 0.0f;**  **GLfloat position1 = 0.0f;**  **GLfloat speed = 0.1f;**  **void display();**  **void bench()**  **{**  **glBegin(GL\_QUADS);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.25f, 0.15f);**  **glVertex2f(-0.25f, 0.0f);**  **glVertex2f(0.25f, 0.0f);**  **glVertex2f(0.25f, 0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.25f, 0.0f);**  **glVertex2f(-0.35f, -0.1f);**  **glVertex2f(0.35f, -0.1f);**  **glVertex2f(0.25f, 0.0f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.35f, -0.1f);**  **glVertex2f(-0.35f, -0.15f);**  **glVertex2f(0.35f, -0.15f);**  **glVertex2f(0.35f, -0.1f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.35f, -0.15f);**  **glVertex2f(-0.35f, -0.25f);**  **glVertex2f(-0.33f, -0.25f);**  **glVertex2f(-0.33f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(0.35f, -0.15f);**  **glVertex2f(0.35f, -0.25f);**  **glVertex2f(0.33f, -0.25f);**  **glVertex2f(0.33f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.25f, -0.15f);**  **glVertex2f(-0.25f, -0.2f);**  **glVertex2f(-0.23f, -0.2f);**  **glVertex2f(-0.23f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(0.25f, -0.15f);**  **glVertex2f(0.25f, -0.2f);**  **glVertex2f(0.23f, -0.2f);**  **glVertex2f(0.23f, -0.15f);**  **glEnd();**  **}**  **void tree()**  **{**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.2f);**  **glVertex2f(-0.3f, 0.0f);**  **glVertex2f(0.3f, 0.0f);**  **glEnd();**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.3f);**  **glVertex2f(-0.25f, 0.1f);**  **glVertex2f(0.25f, 0.1f);**  **glEnd();**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.4f);**  **glVertex2f(-0.15f, 0.2f);**  **glVertex2f(0.15f, 0.2f);**  **glEnd();**  **glBegin(GL\_POLYGON);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.05f, 0.02f);**  **glVertex2f(-0.05f, -0.3f);**  **glVertex2f(0.05f, -0.3f);**  **glVertex2f(0.05f, 0.02f);**  **glEnd();**  **}**  **void update(int value) {**  **if(position <-1.5)**  **position = 1.0f;**  **position -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update,0);**  **}**  **void update1(int value) {**  **if(position1 >1.5)**  **position1 = -1.0f;**  **position1 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void init() {**  **glClearColor(1.0f, 1.0f, 1.0f, 1.0f);**  **}**  **void display3() //up**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(0.0f,position1, 0.0f);**  **bench();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void display2() //down**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(0.0f,position, 0.0f);**  **glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin**  **tree();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void display1() //right**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(position1,0.0f, 0.0f);**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(0.0f,-0.2f);**  **glVertex2f(0.3f, 0.0f);**  **glVertex2f(0.0f, 0.2f);**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void display() { //left**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glLoadIdentity();**  **glPushMatrix();**  **glTranslatef(position,0.0f, 0.0f);**  **glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin**  **for(int i=0;i<200;i++)**  **{**  **glColor3f(1.0,0.0,0.0);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.10;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void specialKeys(int key, int x, int y) {**  **switch (key) {**  **case GLUT\_KEY\_UP:**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display3);**  **break;**  **case GLUT\_KEY\_DOWN:**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display2);**  **break;**  **case GLUT\_KEY\_LEFT:**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display);**  **break;**  **case GLUT\_KEY\_RIGHT:**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display1);**  **break;**  **}**  **}**  **void drawScene() {**  **glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glFlush();**  **glutSwapBuffers();**  **}**  **int main(int argc, char\*\* argv) {**  **//Initialize GLUT**  **glutInit(&argc, argv);**  **glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGB | GLUT\_DEPTH);**  **glutInitWindowSize(800, 500);**  **glutCreateWindow("Transformation");**  **glutSpecialFunc(specialKeys); //Special Key Handler**  **glutDisplayFunc(drawScene);**  **init();**  **glutTimerFunc(100, update, 0);**  **glutTimerFunc(100, update1, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-**  **OnKeyDown**    **OnKeyLeft**    **OnKeyRight**  **OnKeyUp** |

|  |
| --- |
| **Question-**  Develop a code that will have four different objects (keep it simple). Two of the objects will move to the right as the right click is made on the mouse and two of the objects will move to the left as the left key is pressed on the mouse. |
| **Code-**  **#include <iostream>**  **#include <stdlib.h>**  **#include<GL/gl.h>**  **#include <GL/glut.h>**  **#include<math.h>**  **#include<cstring>**  **using namespace std;**  **GLfloat position = 0.0f;**  **GLfloat position1 = 0.0f;**  **GLfloat speed = 0.1f;**  **void display();**  **void bench()**  **{**  **glTranslatef(0.0f,0.4f, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.25f, 0.15f);**  **glVertex2f(-0.25f, 0.0f);**  **glVertex2f(0.25f, 0.0f);**  **glVertex2f(0.25f, 0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.25f, 0.0f);**  **glVertex2f(-0.35f, -0.1f);**  **glVertex2f(0.35f, -0.1f);**  **glVertex2f(0.25f, 0.0f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.35f, -0.1f);**  **glVertex2f(-0.35f, -0.15f);**  **glVertex2f(0.35f, -0.15f);**  **glVertex2f(0.35f, -0.1f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.35f, -0.15f);**  **glVertex2f(-0.35f, -0.25f);**  **glVertex2f(-0.33f, -0.25f);**  **glVertex2f(-0.33f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(0.35f, -0.15f);**  **glVertex2f(0.35f, -0.25f);**  **glVertex2f(0.33f, -0.25f);**  **glVertex2f(0.33f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(-0.25f, -0.15f);**  **glVertex2f(-0.25f, -0.2f);**  **glVertex2f(-0.23f, -0.2f);**  **glVertex2f(-0.23f, -0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.0f, 0.0f);//black**  **glVertex2f(0.25f, -0.15f);**  **glVertex2f(0.25f, -0.2f);**  **glVertex2f(0.23f, -0.2f);**  **glVertex2f(0.23f, -0.15f);**  **glEnd();**  **}**  **void tree()**  **{**  **glTranslatef(0.2f, -0.4f , 0.0f);**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.2f);**  **glVertex2f(-0.3f, 0.0f);**  **glVertex2f(0.3f, 0.0f);**  **glEnd();**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.3f);**  **glVertex2f(-0.25f, 0.1f);**  **glVertex2f(0.25f, 0.1f);**  **glEnd();**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 1.0f, 0.0f);//green**  **glVertex2f(0.0f, 0.4f);**  **glVertex2f(-0.15f, 0.2f);**  **glVertex2f(0.15f, 0.2f);**  **glEnd();**  **glBegin(GL\_POLYGON);**  **glColor3f(0.60f, 0.40f, 0.12f);//brown**  **glVertex2f(-0.05f, 0.02f);**  **glVertex2f(-0.05f, -0.3f);**  **glVertex2f(0.05f, -0.3f);**  **glVertex2f(0.05f, 0.02f);**  **glEnd();**  **}**  **void update(int value) {**  **if(position <-1.5)**  **position = 1.0f;**  **position -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update,0);**  **}**  **void update1(int value) {**  **if(position1 >1.5)**  **position1 = -1.0f;**  **position1 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void init() {**  **glClearColor(1.0f, 1.0f, 1.0f, 1.0f);**  **}**  **void display2() //down**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void display1() //right**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glPushMatrix();**  **glTranslatef(position1,0.0f, 0.0f);**  **bench();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position1,0.0f, 0.0f);**  **glBegin(GL\_TRIANGLES);**  **glColor3f(0.0f, 0.0f, 1.0f);**  **glVertex2f(0.0f,-0.2f);**  **glVertex2f(0.3f, 0.0f);**  **glVertex2f(0.0f, 0.2f);**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void display() { //left**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glPushMatrix();**  **glTranslatef(position,0.0f, 0.0f);**  **tree();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position,0.0f, 0.0f);**  **glBegin(GL\_POLYGON);// Draw a Red 1x1 Square centered at origin**  **for(int i=0;i<200;i++)**  **{**  **glColor3f(1.0,0.0,0.0);**  **float pi=3.1416;**  **float A=(i\*2\*pi)/200;**  **float r=0.10;**  **float x = r \* cos(A);**  **float y = r \* sin(A);**  **glVertex2f(x,y);**  **}**  **glEnd();**  **glPopMatrix();**  **glutTimerFunc(1500,0,0);**  **glFlush();**  **glutSwapBuffers();**  **}**  **void handleMouse(int button, int state, int x, int y)**  **{**  **if (button == GLUT\_LEFT\_BUTTON)**  **{**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display);**  **}**  **if (button == GLUT\_RIGHT\_BUTTON)**  **{**  **position = 0.0f;**  **position1 = 0.0f;**  **speed = 0.05f;**  **glutDisplayFunc(display1);**  **}**  **glutPostRedisplay();**  **}**  **void drawScene() {**  **glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);**  **//glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glFlush();**  **glutSwapBuffers();**  **}**  **int main(int argc, char\*\* argv) {**  **//Initialize GLUT**  **glutInit(&argc, argv);**  **glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGB | GLUT\_DEPTH);**  **glutInitWindowSize(800, 500);**  **glutCreateWindow("Mouse Interaction to move object");**  **glutMouseFunc(handleMouse);**  **glutDisplayFunc(drawScene);**  **init();**  **glutTimerFunc(100, update, 0);**  **glutTimerFunc(100, update1, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-**  **Two Object Moving Right On Mouse Right Click**    **Two Object Moving Left On Mouse Left Click** |