**Lab Taks-6**

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 include resources for all the section in the table

|  |
| --- |
| **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 position1 = 0.0f;**  **GLfloat position2 = 0.0f;**  **GLfloat speed = 0.1f;**  **void dis();**  **void display();**  **void update1(int value) {**  **if(position1 <-1.0)**  **position1 = 1.0f;**  **position1 -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void update2(int value) {**  **if(position2 >1.0)**  **position2 = -1.0f;**  **position2 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update2,0);**  **}**  **void disback(int val)**  **{**  **glutDisplayFunc(display);**  **}**  **void display5()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(255.0f, 0.0f, 0.0f, 1.0f);//red**  **glutTimerFunc(2000,disback,0);**  **glFlush();**  **}**  **void display4(int val) {**  **glutDisplayFunc(display5);**  **}**  **void display3()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(0.0f, 0.0f, 255.0f, 1.0f);//blue**  **glutTimerFunc(2000,display4,0);**  **glFlush();**  **}**  **void display2(int val)**  **{**  **glutDisplayFunc(display3);**  **}**  **void display()**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glClearColor(255.0f, 255.0f, 0.0f, 1.0f);//yellow**  **glLoadIdentity();**  **glutTimerFunc(2000,display2,0);**  **glFlush();**  **}**  **void dis()**  **{**  **glutDisplayFunc(display);**  **}**  **int main(int argc, char\*\* argv) {**  **glutInitWindowSize(320, 320);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Animation");**  **glutDisplayFunc(dis);**  **glutTimerFunc(100, update1, 0);**  **glutTimerFunc(100, update2, 0);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Develop an animation that will call four objects separately, each after 20 ms. |
| **Code-**  **#include<cstdio>**  **#include <GL/gl.h>**  **#include <GL/glut.h>**  **GLfloat position1 = 0.0f;**  **GLfloat position2 = 0.0f;**  **GLfloat speed = 0.1f;**  **void dis();**  **void display();**  **void update1(int value) {**  **if(position1 <-1.5)**  **position1 = 1.5f;**  **position1 -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void update2(int value) {**  **if(position2 >1.5)**  **position2 = -1.5f;**  **position2 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update2,0);**  **}**  **void init() {**  **glClearColor(1.0f, 1.0f, 1.0f, 1.0f);**  **}**  **void disback(int val)**  **{**  **glutDisplayFunc(display);**  **}**  **void box1(){**  **glBegin(GL\_QUADS);**  **glColor3f(255,0,0);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f( 0.2f, -0.2f);**  **glVertex2f( 0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glEnd();**  **}**  **void box2(){**  **glBegin(GL\_QUADS);**  **glColor3f(0,0,255);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f( 0.2f, -0.2f);**  **glVertex2f( 0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glEnd();**  **}**  **void box3(){**  **glBegin(GL\_QUADS);**  **glColor3f(255,255,0);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f( 0.2f, -0.2f);**  **glVertex2f( 0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glEnd();**  **}**  **void box4(){**  **glBegin(GL\_QUADS);**  **glColor3f(0,255,0);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f( 0.2f, -0.2f);**  **glVertex2f( 0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glEnd();**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glLoadIdentity();**  **glPushMatrix();**  **glTranslatef(position1,0.0f, 0.0f);**  **box1();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position2,0.0f, 0.0f);**  **box2();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(0.0f,position1, 0.0f);**  **box3();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(0.0f,position2, 0.0f);**  **box4();**  **glPopMatrix();**  **glFlush();**  **}**  **void dis(){**  **glutDisplayFunc(display);**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitWindowSize(320, 320);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Animation");**  **glutDisplayFunc(dis);**  **init();**  **glutTimerFunc(100, update1, 0);**  **glutTimerFunc(100, update2, 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-** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Develop a code that will have four different objects (keep it simple). Four different keys will be dedicated each objects. The objects will move to the left, right, up and down in a loop as the keys are pressed individually. |
| **Code-**  **#include<cstdio>**  **#include<stdio.h>**  **#include <GL/gl.h>**  **#include <GL/glut.h>**  **GLfloat position = 0.0f;**  **GLfloat position2 = 0.0f;**  **GLfloat position3 = 0.0f;**  **GLfloat position4 = 0.0f;**  **GLfloat speed = 0.1f;**  **GLfloat speed2 = 0.1f;**  **GLfloat speed3 = 0.1f;**  **GLfloat speed4 = 0.1f;**  **void update(int value) {**  **position -= speed;**  **if(position <-1.0)**  **{**  **position = 1.0f;**  **}**  **position2 += speed2;**  **if(position2 > 1)**  **{**  **position2 = - 1.0f ;**  **}**  **position3 -= speed3;**  **if(position3 <-1.0)**  **{**  **position3 = 1.0f;**  **}**  **position4 += speed4;**  **if(position4 > 1)**  **{**  **position4 = - 1.0f ;**  **}**  **glutPostRedisplay();**  **glutTimerFunc(100, update, 0);**  **}**  **void display() {**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glPushMatrix();**  **glTranslatef(position,0.0f, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(255,0,0);**  **glVertex2f(-0.1f, 0.8f);**  **glVertex2f( 0.1f, 0.8f);**  **glVertex2f( 0.1f, 0.6f);**  **glVertex2f(-0.1f, 0.6f);**  **glEnd();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position2,0.0f, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(255,0,0);**  **glVertex2f(-0.1f, 0.5f);**  **glVertex2f( 0.1f, 0.5f);**  **glVertex2f( 0.1f, 0.3f);**  **glVertex2f(-0.1f, 0.3f);**  **glEnd();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(0.0f,position3, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(255,0,0);**  **glVertex2f(-0.3f, -0.8f);**  **glVertex2f( -0.1f, -0.8f);**  **glVertex2f( -0.1f, -0.6f);**  **glVertex2f(-0.3f, -0.6f);**  **glEnd();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(0.0f,position4, 0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(255,0,0);**  **glVertex2f(0.1f, -0.5f);**  **glVertex2f( 0.3f, -0.5f);**  **glVertex2f( 0.3f, -0.3f);**  **glVertex2f(0.1f, -0.3f);**  **glEnd();**  **glPopMatrix();**  **glFlush();**  **}**  **void handleKeypress(unsigned char key, int x, int y) {**  **switch (key) {**  **case 'a':**  **speed = 0.0f;**  **speed2 = 0.0f;**  **speed3 = 0.0f;**  **speed4 = 0.0f;**  **break;**  **case 'b':**  **speed3 += 0.1f;**  **break;**  **case 'c':**  **speed4 += 0.1f;**  **break;**  **case 'd':**  **speed += 0.1f;**  **break;**  **case 'e':**  **speed2 += 0.1f;**  **break;**  **glutPostRedisplay();**  **}}**  **int main(int argc, char\*\* argv)**  **{**  **glutInit(&argc, argv);**  **glutInitWindowSize(1020, 680);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Keyboard Interaction Animation");**  **glutDisplayFunc(display);**  **glutTimerFunc(100, update, 0);**  **glutKeyboardFunc(handleKeypress);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **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 <GL/gl.h>**  **#include <GL/glut.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=1.0f;**  **position -= speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update,0);**  **}**  **void update1(int value)**  **{**  **if(position1>1.0)**  **position1 = -1.0f;**  **position1 += speed;**  **glutPostRedisplay();**  **glutTimerFunc(100,update1,0);**  **}**  **void display2() {**  **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(0,0,255);**  **glVertex2f(0.1f, 0.0f);**  **glVertex2f(0.5f, 0.0f);**  **glVertex2f(0.5f, 0.2f);**  **glVertex2f(0.1f, 0.2);**  **glEnd();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position1,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(0,0,255);**  **glVertex2f(-0.1f, 0.0f);**  **glVertex2f(-0.5f, 0.0f);**  **glVertex2f(-0.5f, 0.2f);**  **glVertex2f(-0.1f, 0.2);**  **glEnd();**  **glPopMatrix();**  **glFlush();**  **}**  **void init() {**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f);**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glLoadIdentity();**  **glPushMatrix();**  **glTranslatef(position,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(0,0,255);**  **glVertex2f(0.1f, 0.0f);**  **glVertex2f(0.5f, 0.0f);**  **glVertex2f(0.5f, 0.2f);**  **glVertex2f(0.1f, 0.2);**  **glEnd();**  **glPopMatrix();**  **glPushMatrix();**  **glTranslatef(position,0.0f,0.0f);**  **glBegin(GL\_QUADS);**  **glColor3f(0,0,255);**  **glVertex2f(-0.1f, 0.0f);**  **glVertex2f(-0.5f, 0.0f);**  **glVertex2f(-0.5f, 0.2f);**  **glVertex2f(-0.1f, 0.2);**  **glEnd();**  **glPopMatrix();**  **glPopMatrix();**  **glFlush();**  **glFlush();**  **}**  **void handleMouse(int button, int state, int x, int y) {**  **if (button == GLUT\_LEFT\_BUTTON)**  **{ glutDisplayFunc(display);**  **}**  **if (button == GLUT\_RIGHT\_BUTTON)**  **{glutDisplayFunc(display2);**  **}**  **glutPostRedisplay();}**  **void handleKeypress(unsigned char key, int x, int y) {**  **switch (key) {**  **case 'a':**  **speed = 0.0f;**  **break;**  **case 'w':**  **speed = 0.1f;**  **break;**  **glutPostRedisplay();**  **}}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutInitWindowSize(320, 320);**  **glutInitWindowPosition(50, 50);**  **glutCreateWindow("Basic Animation");**  **glutDisplayFunc(display);**  **init();**  **glutTimerFunc(100, update, 0); //Add a timer**  **glutTimerFunc(100, update1, 0); //Add a timer**  **glutKeyboardFunc(handleKeypress);**  **glutMouseFunc(handleMouse);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |