Lab Taks-7

Md. Abdul Muneem Adnan
20-44213-3

Question-1

Create an animation using two boxes that will move in the opposite direction.

Graph Plot (Picture)-

[Not needed]

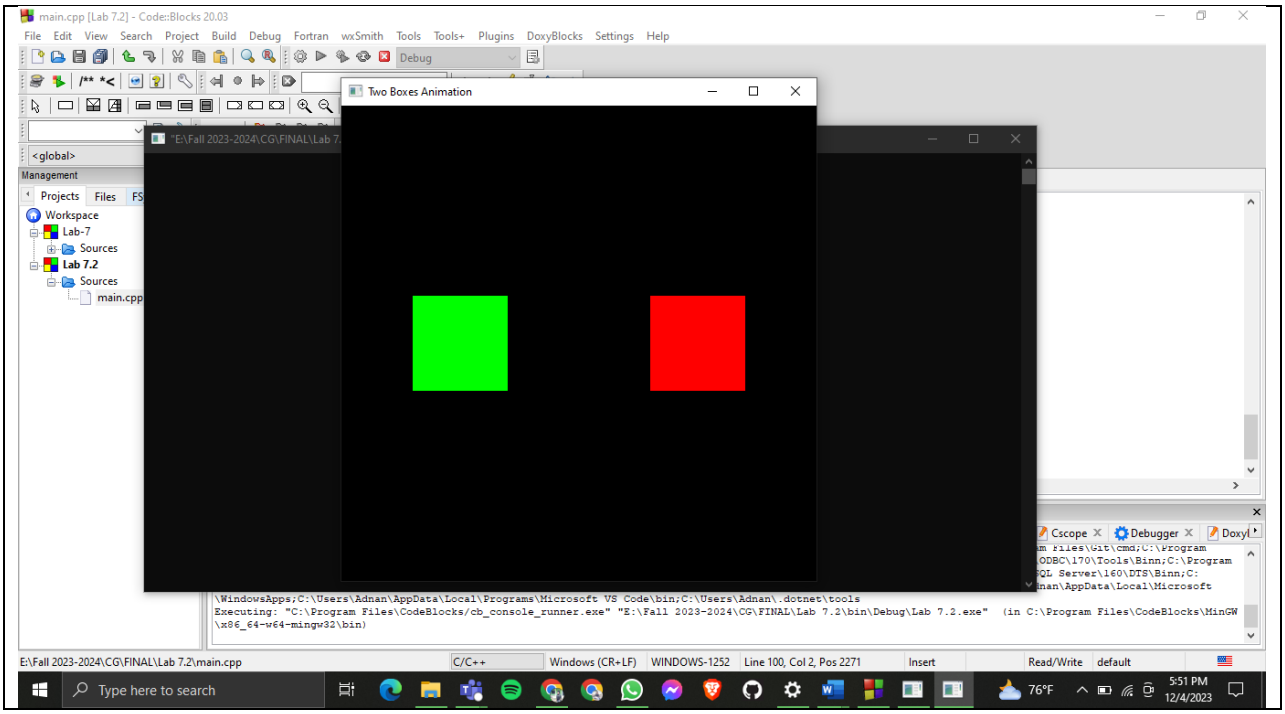
Code-

```
#include<cstdio>
#include <GL/gl.h>
#include <GL/glut.h>

GLfloat positionLeft = -1.0f;
GLfloat positionRight = 1.0f;
GLfloat speed = 0.1f;

void display();
void updateLeft(int value);
void updateRight(int value);
void contextMenu(int id);

void init() {
   glClearColor(0.0f, 0.0f, 0.0f, 1.0f);
}

void display() {
   glClear(GL_COLOR_BUFFER_BIT);
   glLoadIdentity();

   glPushMatrix();
   glTranslatef(positionLeft, 0.0f, 0.0f);
   glBegin(GL_POLYGON);
   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();

    glPushMatrix();
    glTranslatef(positionRight, 0.0f, 0.0f);
    glBegin(GL_POLYGON);
    glColor3f(1.0f, 0.0f, 0.0f);
    glVertex2f(-0.2f, -0.2f);
    glVertex2f(0.2f, -0.2f);
    glVertex2f(0.2f, 0.2f);
    glVertex2f(-0.2f, 0.2f);
    glEnd();
    glPopMatrix();

    glutSwapBuffers();
}

void updateLeft(int value) {
    if (positionLeft > 1.5) {
        positionLeft = -1.0f;
    }
    positionLeft += speed;
    glutPostRedisplay();
    glutTimerFunc(100, updateLeft, 0);
}

void updateRight(int value) {
    if (positionRight < -1.5) {
        positionRight = 1.0f;
    }
    positionRight -= speed;
    glutPostRedisplay();
    glutTimerFunc(100, updateRight, 0);
}

void contextMenu(int id) {
    switch (id) {
        case 1:
            glutTimerFunc(100, updateLeft, 0);
            break;
        case 2:
            glutTimerFunc(100, updateRight, 0);
            break;
        case 3:
            exit(0);
```

```
            break;
        default:
            break;
    }
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitWindowSize(520, 520);
    glutInitWindowPosition(50, 50);
    glutCreateWindow("Animation of two boxes that will move in the opposite direction");
    glutDisplayFunc(display);
    init();

    glutCreateMenu(contextMenu);
    glutAddMenuEntry("Move Box from Left", 1);
    glutAddMenuEntry("Move Box from Right", 2);
    glutAddMenuEntry("Exit", 3);
    glutAttachMenu(GLUT_RIGHT_BUTTON);

    glutTimerFunc(100, updateLeft, 0);
    glutTimerFunc(100, updateRight, 0);

    glutMainLoop();
    return 0;
}
```

Output Screenshot (Full Screen)-



Question-2

Move your car off (lab task 6) from left to right in a loop.

Graph Plot (Picture)-

[Not needed]

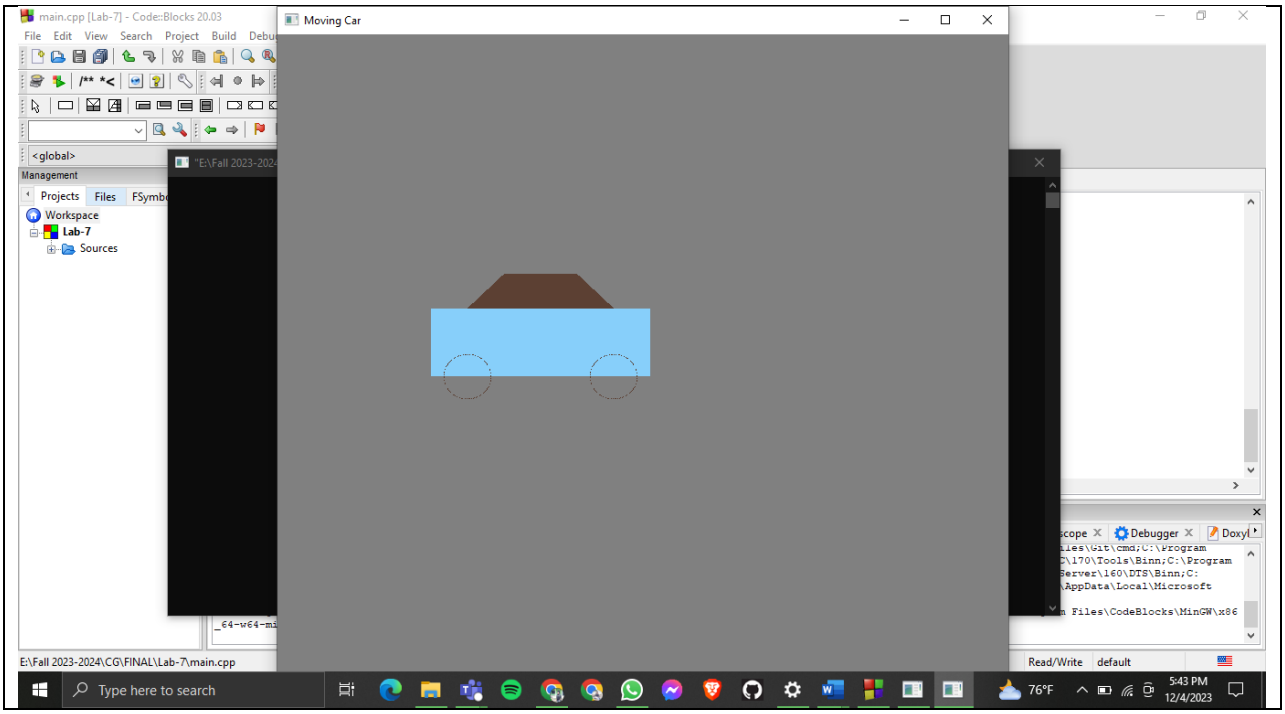
Code-

```
#include <iostream>
#include <GL/gl.h>
#include <GL/glut.h>
#include <math.h>

float _move = 0.0f;
float _angle1 = 0.0f;

void wheel(float x) {
   glLoadIdentity();
   glMatrixMode(GL_MODELVIEW);
   glPushMatrix();
   glTranslatef(x, 0, 0);
   glRotatef(_angle1, 0.0f, 0.0f, 1.0f);
```

```
   glBegin(GL_LINES);
   for (int i = 0; i < 200; i++) {
      glColor3f(0.36, 0.25, 0.20);
      float pi = 3.1416;
      float A = (i * 2 * pi) / 200;
      float r = 0.065;
      float wheel_x = r * cos(A);
      float wheel_y = r * sin(A);
      glVertex2f(wheel_x, wheel_y);
   }
   glEnd();
   glPopMatrix();
}

void drawScene() {
   glClearColor(0.5, 0.5, 0.5, 1.0);
   glClear(GL_COLOR_BUFFER_BIT);
   glColor3d(0.53, 0.81, 0.98);

   glBegin(GL_QUADS);
   glVertex2f(-0.3f + _move, 0.0f);
   glVertex2f(0.3f + _move, 0.0f);
   glVertex2f(0.3f + _move, 0.2f);
   glVertex2f(-0.3f + _move, 0.2);
   glEnd();

   glColor3d(0.36, 0.25, 0.20);

   glBegin(GL_QUADS);
   glVertex2f(-0.2f + _move, 0.2f);
   glVertex2f(0.2f + _move, 0.2f);
   glVertex2f(0.1f + _move, 0.3f);
   glVertex2f(-0.1f + _move, 0.3);
   glEnd();

   wheel(-0.2 + _move);  // Left wheel
   wheel(0.2 + _move);   // Right wheel

   glutSwapBuffers();
}

void update(int value) {
   _move -= 0.02; // Change direction to move from right to left
   if (_move < -1.3) {
```

```
        _move = 1.0;
    }
    glutPostRedisplay();
    glutTimerFunc(20, update, 0);
}

void update1(int value) {
    _angle1 += 2.0f;
    if (_angle1 > 360.0) {
        _angle1 -= 360;
    }
    glutPostRedisplay();
    glutTimerFunc(20, update1, 0);
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
    glutInitWindowSize(800, 800);
    glutCreateWindow("Moving Car");
    glutDisplayFunc(drawScene);
    gluOrtho2D(-2, 2, -2, 2);
    glutTimerFunc(20, update, 0);
    glutTimerFunc(20, update1, 0);
    glutMainLoop();
    return 0;
}
```

Output Screenshot (Full Screen)-


main.cpp [Lab-7] - Code::Blocks 20.03
File Edit View Search Project Build
Moving Car
Management
Projects Files
Workspace
Lab-7
Sources
Debugger
E:\Fall 2023-2024\CG\FINAL\Lab-7\main.cpp
Read/Write
default
Type here to search
76°F
5:43 PM
12/4/2023