



Faculty of Computers &  
Artificial Intelligence



Benha University

## CAR SIMULATION

Computer Graphics Project [ CS Dept. ]

by

Taha Ibrahim Taha (SEC.4)	Hossam Haytham Ismail (SEC.3)
Khaled Abdul Mohsen Sayed (SEC.3)	Hager Yousef Salah (SEC.9)
Gamal Adel Gamal (SEC.3)	Habiba Ramadan Mohamed(SEC.3)

Supervised by

Dr. Rasha Orban

ENG. Hisham Gamil

## Contents

<b>1. Introduction.....</b>	<b>3</b>
<b>2. project Description.....</b>	<b>3</b>
<b>3.used tools. ....</b>	<b>4</b>
<b>4. Various functions used in this programC++.....</b>	<b>4</b>
<b>5. used key .....</b>	<b>4</b>
<b>6. Screen code .....</b>	<b>5</b>
<b>7. Screen Gui.....</b>	<b>7</b>

## 1. Introduction.

As a software interface for graphics hardware, OpenGL's main purpose is to render two- and three-dimensional objects into a frame buffer. These objects are described as sequences of vertices or pixels. OpenGL performs several processing steps on this data to convert it to pixels to form the final desired image in the frame buffer.

## 2. Game Description.

The main aim of the 3D Car Simulation Computer Graphics Mini Project is to illustrate the concepts and usage of pre-built functions in OpenGL. Simulation of a 3d Car Simulation is being done using computer graphics. 3D Car Simulation is a project where we can also change the weather & color of the car and other graphics.

## 3.used tools

- C++
- Opengl
- Microsoft visual studio

## 4. Various functions used in this programC++

- glutInit() : interaction between the windowing system and OPENGL is initiated
- glutInitDisplayMode() : used when double buffering is required and depth information is required
- glutCreateWindow() : this opens the OPENGL window and displays the title at top of the window
- glutInitWindowSize() : specifies the size of the window
- glutInitWindowPosition() : specifies the position of the window in screen co-ordinates
- glPushMatrix() : set current matrix on the stack
- glPopMatrix() :pop the old matrix without the transformations.

- glutKeyboardFunc() : handles normal ascii symbols
- glutSpecialFunc() : handles special keyboard keys
- glLightfv() : gives light in an area
- glutIdleFunc() : this handles the processing of the background
- glutDisplayFunc() : this handles redrawing of the window
- glutMainLoop() : this starts the main loop, it never returns
- glViewport() : used to set up the viewport
- glVertex3fv() : used to set up the points or vertices in three dimensions
- glColor3fv() : used to render color to faces
- glFlush() : used to flush the pipeline
- glutPostRedisplay() : used to trigger an automatic redrawal of the object
- glMatrixMode() : used to set up the required mode of the matrix
- glLoadIdentity() : used to load or initialize to the identity matrix
- glTranslatef() : used to translate or move the rotation centre from one point to another in three dimensions
- glRotatef() : used to rotate an object through a specified rotation angle.

## 5.used key

The controls in the project are three.

1-Press (SPACE BAR) to enter.

2-MOUSE

- Press (RIGHT BUTTON) for menu.

3-KEYBOARD

- X-Y-Z keys for corresponding potation.
- A-S-Q car custom size selection.
- U-F for camera view selection.
- (LEFT ARROW "<- ") and (RIGHT ARROW "->") to move car.
- ESCAPE to exit.

## 6.Screen code

```
finaly_project (Global Scope) InitGL(GLfloat Width, GLfloat Height)
1  #define STB_IMAGE_IMPLEMENTATION
2  #include<Windows.h>
3  #include <stdio.h>
4  #include <stdlib.h>
5  #include <iostream>
6  #include <GL/glut.h>
7  #include <gl/stb_image.h>
8  #include <math.h>
9  #include<string.h>
10
11  #define ESCAPE 27
12
13  signed int texture;
14  t w, h, nrChannels;
15  signed char* data = NULL;
16
17  float width = GetSystemMetrics(SM_CXSCREEN);
18  float height = GetSystemMetrics(SM_CYSCREEN);
19
20  float i, theta;
21  int nml = 0, day = 1;
22
23  char proj_name[] = "Car Simulator";
24
25  float xt = 0.0, yt = 0.0, zt = 0.0, xw = 0.0; /* x,y,z translation */ /* xw = نسبة الزيادة للأمام */
26
27  float xs = 1.0, ys = 1.0, zs = 1.0; /* xs = scale on x Axis */
28
29  float xangle = 0.0, yangle = 0.0, zangle = 0.0, angle = 0.0; /* axis angles */
30
31  float r = 0, g = 0, b = 1, //car colorz
32  r = 1, gc = 1, bc = 1; //cube colors
33  int light = 1;
34  t count = 1, flg = 1;
35  t view = 0, c = 0;
36  t flag1 = 0, aflag = 1; //to switch car driving mode
37  t flag2 = 0, wheelflag = 0; //to switch fog effect
38  UquadricObj* t;
39
40  static void SpecialKeyFunc(int Key, int x, int y);
41
42  Simple transformation routine */
43  void Transform(GLfloat Width, GLfloat Height)
44
45      glViewport(0, 0, Width, Height); /* Set the viewport */
46      glMatrixMode(GL_PROJECTION); /* Select the projection matrix */
47      glLoadIdentity(); /* Reset The Projection Matrix */
48      gluPerspective(45.0, Width / Height, 0.1, 100.0); /* Calculate The Aspect Ratio Of The Window */
49      glMatrixMode(GL_MODELVIEW); /* Switch back to the modelview matrix */
50
51
52  A general OpenGL initialization function.Sets all of the initial parameters. */
53  void InitGL(GLfloat Width, GLfloat Height)
54
55
56      glClearColor(1.0, 1.0, 1.0, 1.0);
57      glLineWidth(2.0); /* Add line width, ditto */
58      Transform(Width, Height); /* Perform the transformation */
59
```

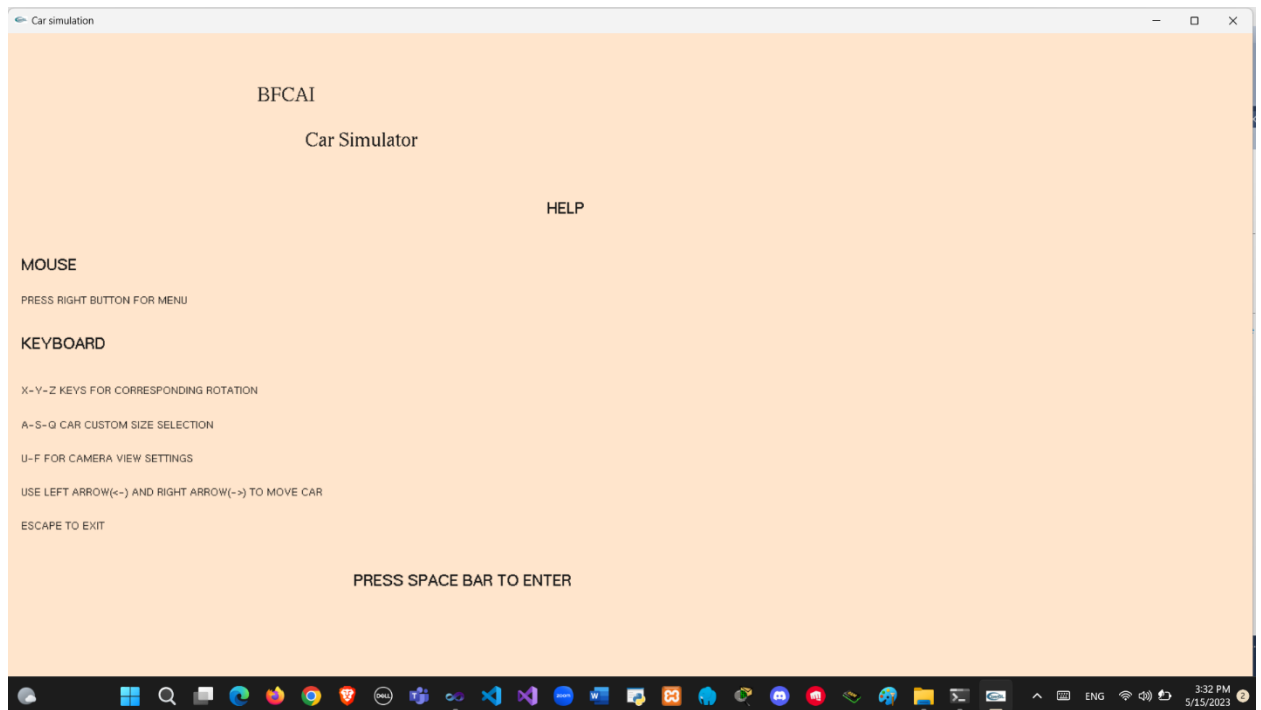
```

84 void init()
85 {
86     glClearColor(0, 0, 0, 0);
87     glEnable(GL_TEXTURE_2D);
88     glPointSize(5.0);
89     glMatrixMode(GL_PROJECTION);
90     glLoadIdentity();
91     glOrtho(0.0, 900.0, 0.0, 600.0, 50.0, -50.0);
92     glutPostRedisplay(); // request redisplay
93 }
94
95 void display_string(int x, int y, char* string, int font)
96 {
97     glColor3f(0.8, 0.52, 1.0);
98     glRasterPos2f(x, y);
99     int len = (int)strlen(string);
100     for (int i = 0; i < len; i++) {
101         if (font == 1)
102             glutBitmapCharacter(GLUT_BITMAP_TIMES_ROMAN_24, string[i]);
103         if (font == 2)
104             glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, string[i]);
105         if (font == 3)
106             glutBitmapCharacter(GLUT_BITMAP_HELVETICA_12, string[i]);
107         if (font == 4)
108             glutBitmapCharacter(GLUT_BITMAP_HELVETICA_10, string[i]);
109     }
110 }
111 }

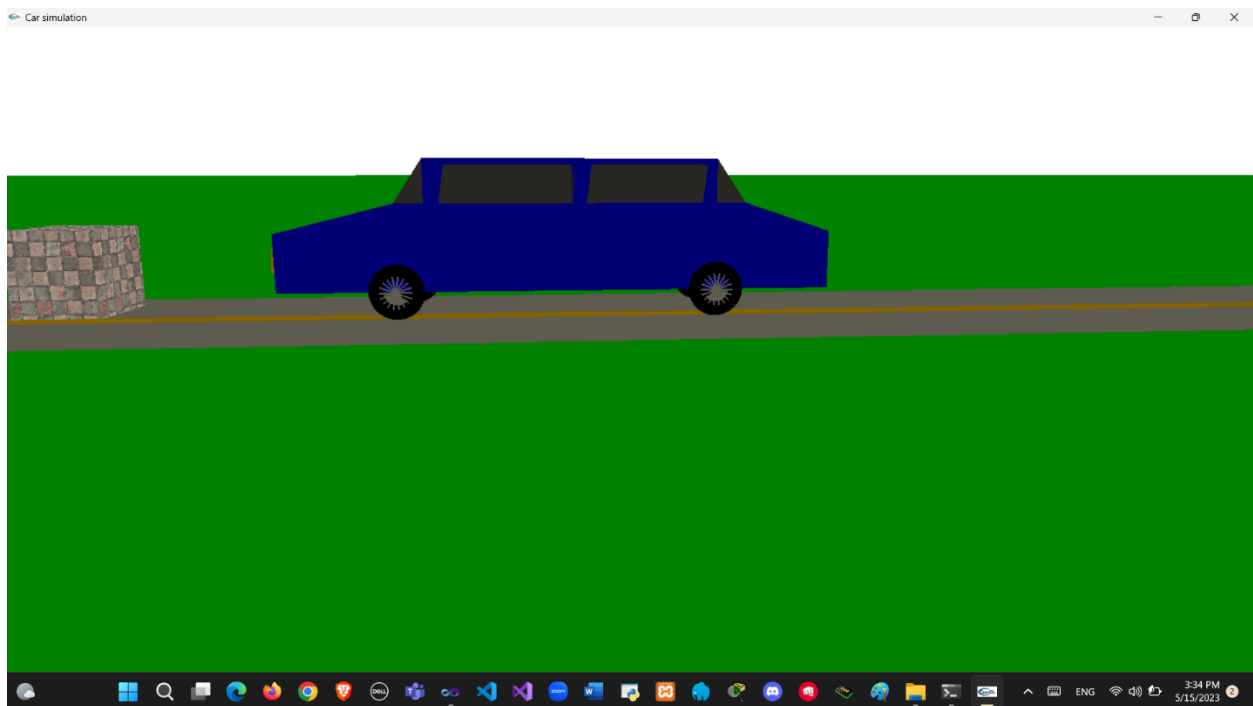
```

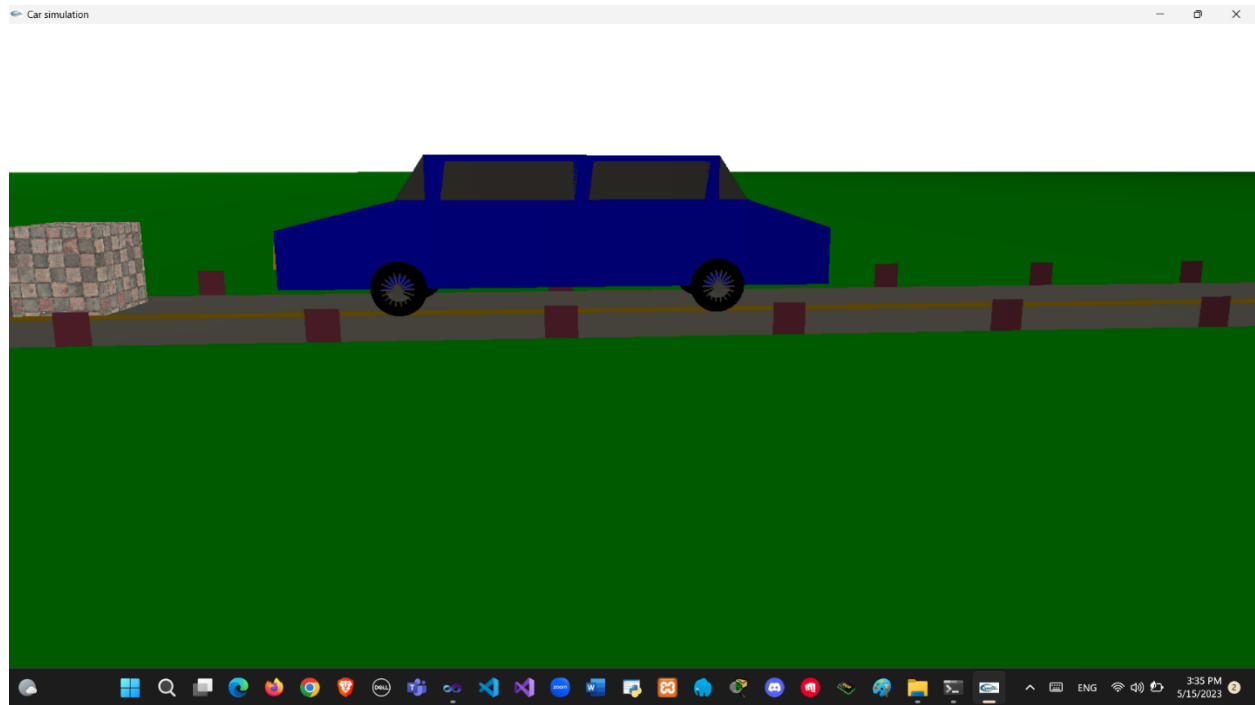
## 6. Screen Gui

### 1- Start screen

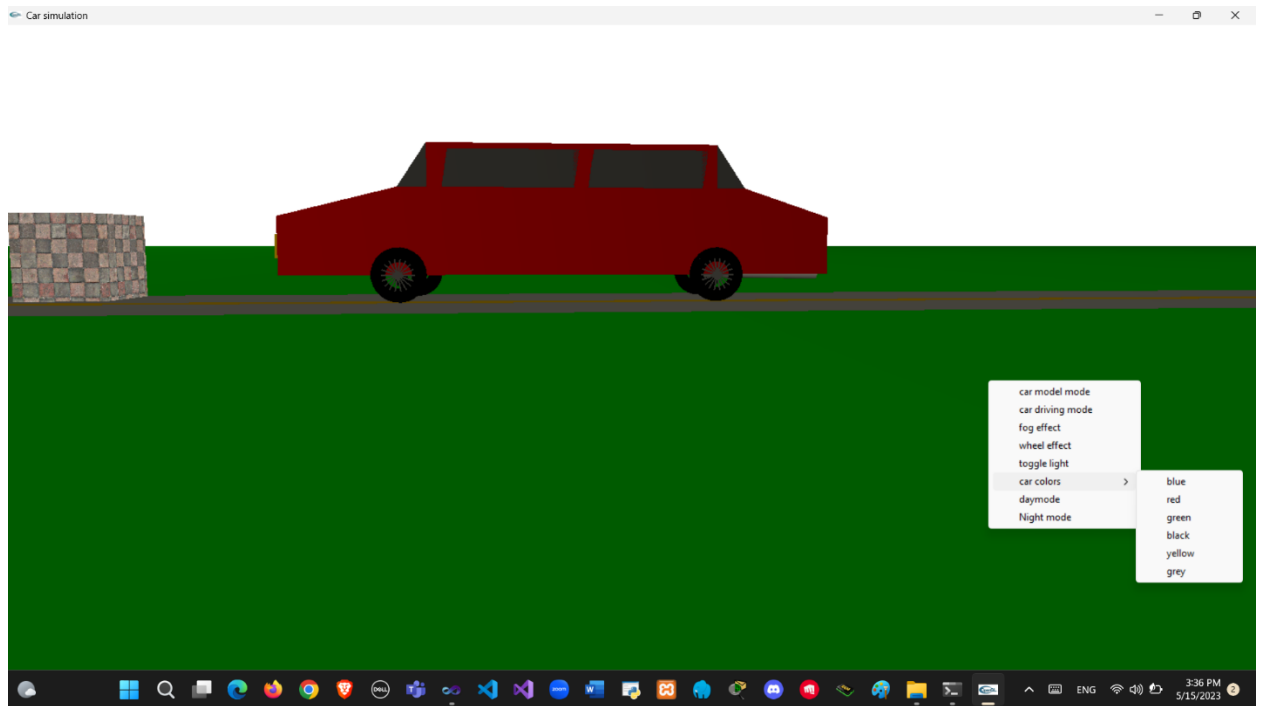


### 2-play

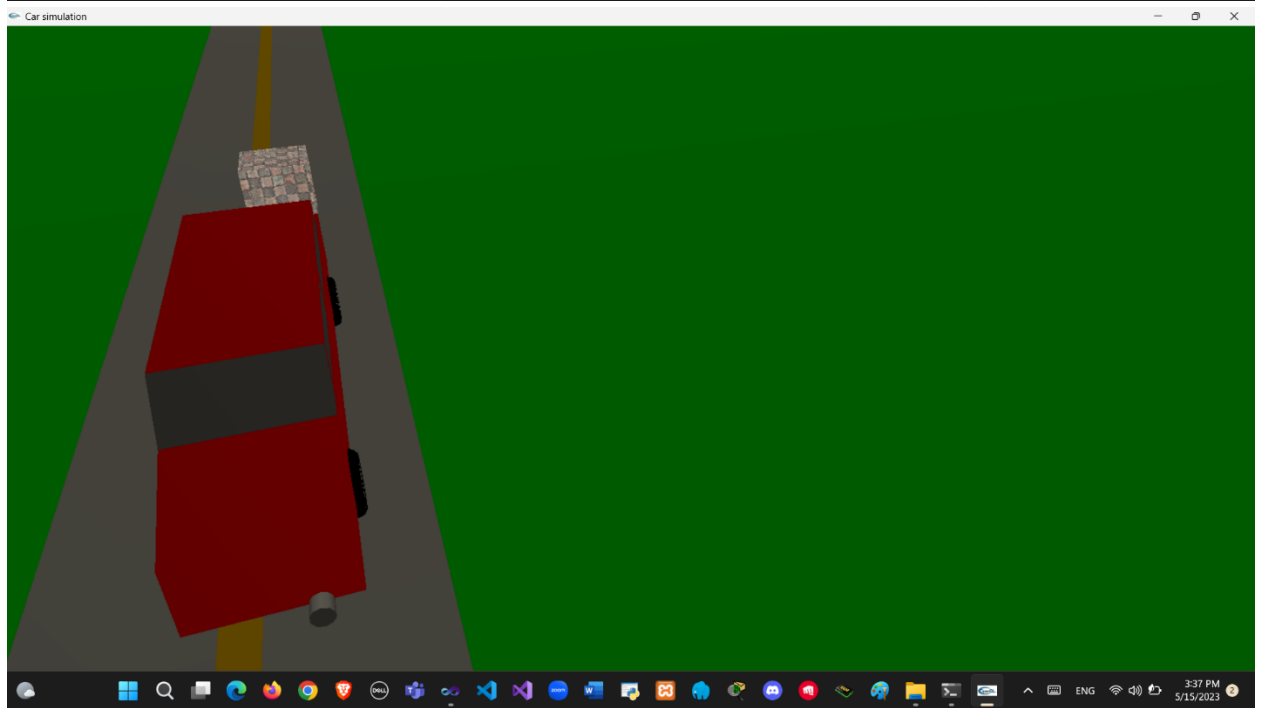
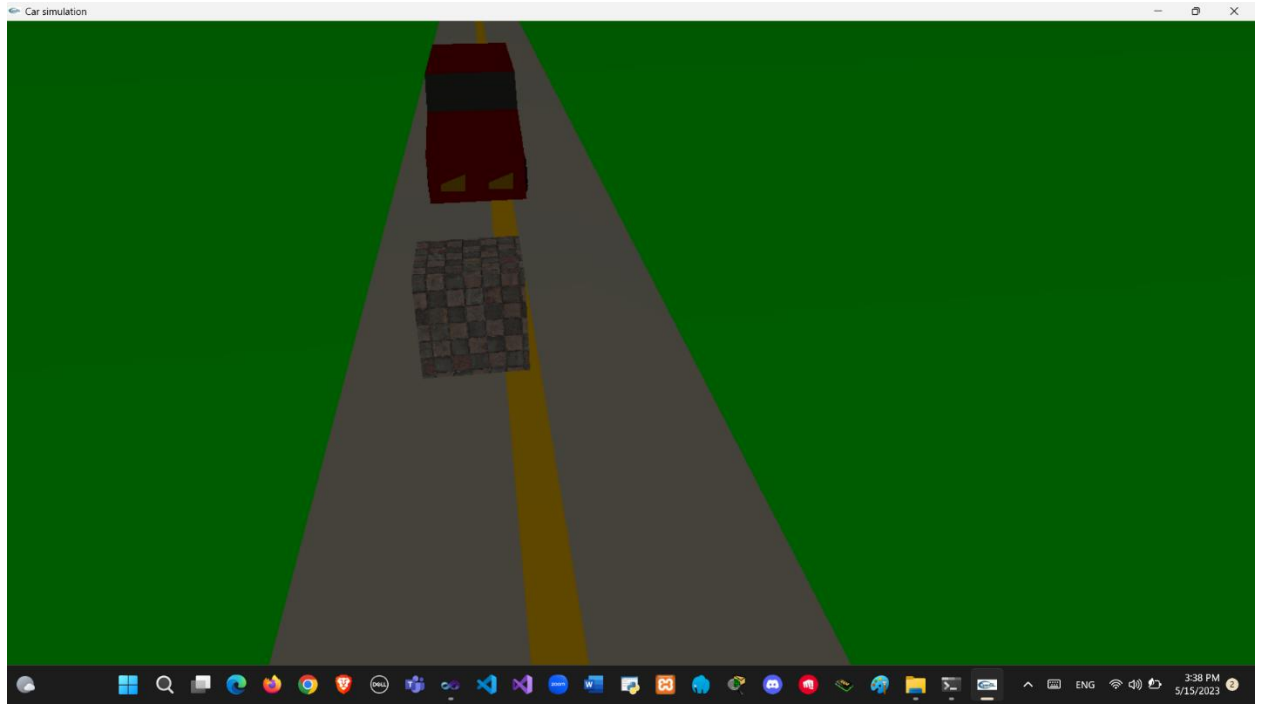




We can change the color of car:







## **7. CONCLUSION**

The project “Car Simulation” clearly demonstrates the simulation of 3d Car Simulation using OpenGL. Finally, we conclude that this program clearly illustrates the 3d Car Simulation using OpenGL and has been completed successfully and is ready to be demonstrated.