University of Dhaka
Department of Computer Science and Engineering
CSE-1211

Lab Project: Traffic Racing.

Submitted by:
1. Md Shahin Alom Shuvo, Roll - 57
2.Nahid Ibne Anwar ,Roll -93

Submitted to:
1.Mr.Md. Shiplu Hawlader
Lecturer, Department of CSE, DU
2. Mr. Hasnain Heickal
Lecturer, Department of CSE, DU

Date of Submission: November 24, 2016

Table of Contenet:

Introduction:

The project is intended to make the students familiar with some application of basic C programming using Borland Graphics Interface (BGI) . Implementation of the students’ theoretical knowledge of C language in some real life scenario was the objective of this project.

Project introduction:

The project is a car based 2D game, in which the player has to control one or more car and escape collisions. The game is named “Traffic Racing”. I chose this name because the meaning goes with the theme of the game and also, the word sounds good.

Game outline:

This is a simple, elegant, car based game, where the player has to control one car   and has to survive as long as he can.

The player has to use the direction keys to avoid collision. The task becomes a little hard in five lane level, where the player has to control and save cars.

The velocity of the cars, coming from opposite direction, increases with time.

Every passing car rewards 1points in both modes.

Game objective:

The Game was designed with the aim to keep the player busy always. Before starting on the project I had thought about the conception of “great games” and “boring games” and finally decided that a game that requires the player to be alert all the time is less likely to be considered as “boring”.

So, the main objective of the game was to provide its user an entertaining experience, and also to make the game as addictive and eye-pleasing as possible.

High quality interactive menus, submenus as well as clean and flicker-free game play environments were created with great care and efforts to achieve that certain goal.

Challenges for the gamer:

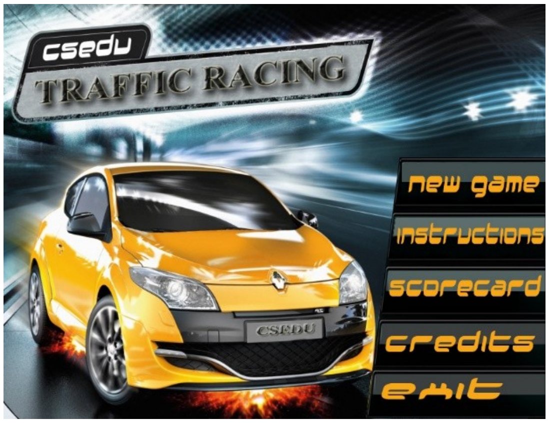
It has been proved in researches that average human brain faces difficulty while concentrating on two different things simultaneously.

In this game, the gamer will have to master the skill of fast thinking and multitasking capabilities to achieve a high score. The scenarios are ever changing and different scenarios require different control input from the gamer. In addition to that, in five lane level, the gamer actually have to concentrate on too many cars which are in opposite parts of the window and also are in opposite direction. Along with the increasing velocity factor, this feature makes this level incredibly hard.

Main features:

(i) Almost zero-flicker game play: In my opinion, this is one of the most eye catching features of game. While working on the project using BGI, many of us, including me, faced the problem of flickering which is very acute in this graphics interface. However, I came up with a completely different technique involving no clearing or filling function, which made this game very smooth and almost completely flicker free. This technique also became popular among my friends.

(ii) Interactive menus: This game has several interactive menus which are very easy to use, and also greatly eye catching. This feature adds to the charm of this game by creating great first impressions. The menus and the submenus were created with high quality images and appropriate keyboard and mouse functions.



(iii) A professional look in the interface: With the help of my limited image processing skill and coding skill, I tried to create a good looking graphics interface for my game.

(iv) Two different modes: The game has two different modes three lane mode and five lane mode which are slightly different from each other in appearance, taste and game play. The logic behind these two levels has their similarities, but at the same time, each of them offers a different taste and strategy than the other. This provides the user with a variation which I think will attract more people to play this game.

(v) Different levels of game play: The Three Lane Mode comes with three different levels; each is different from the other in appearances and in the level of difficulty. This feature adds to the variation of the game.

(vi) Sound effects: This is another feature that makes this game different. The BGI library doesn’t support sound effects on its own, which makes the games a little dull. But with the help of a little online research and the “winmm.lib” library, I managed to include sound effects in my game. This feature helps this game to be more interesting for the player.

Additional features:

(i) Main menu: The main menu is like a dashboard that gives the user access to any part of the program.

(ii) Instruction : The game has an “easy to access” instruction menu that can be accessed through the main menu.

(iii) Score Card: The game also has a score board option that can be accessed through the main menu. The game stores the top nine scores from each level and shows the highest three of them in the Leader board.

(iv) Exit option: The game can be terminated at anytime using the EXIT option, which is located at the main menu.

Source Code properties:

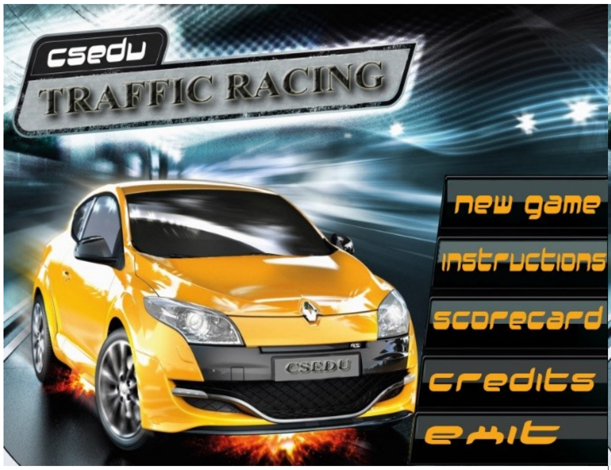
(i) The source was written by following an efficient modular programming system, dividing the whole code in different separate parts and making new user defined header files. This method was encouraged by our teachers and it made our tasks a lot easier while combining all the features and joining the whole project together.

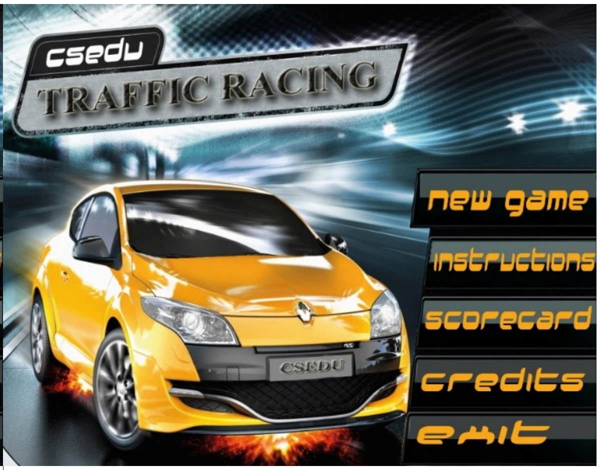
(ii) I tried to use as many different types of programming tools as possible, so that it covers the maximum amount of what we have been taught in C language. The coding part of this project contains-

a. Basic input output b. File input output c. Conditional logic d. For loop and While loop e. Single and multidimensional arrays f. String handling g. Mouse operations h. Keyboard operations i. Pointers j. Structures k. Sorting l. Reading image and music files m. Graphics operations.
And so on.

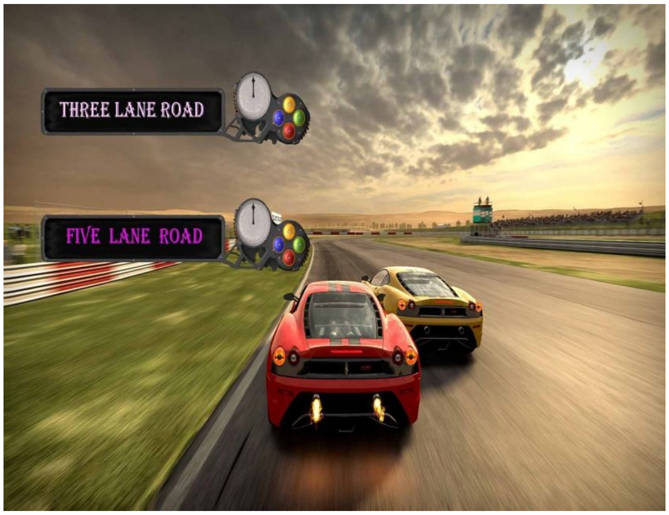
Graphical interface of the game:

The main menu:

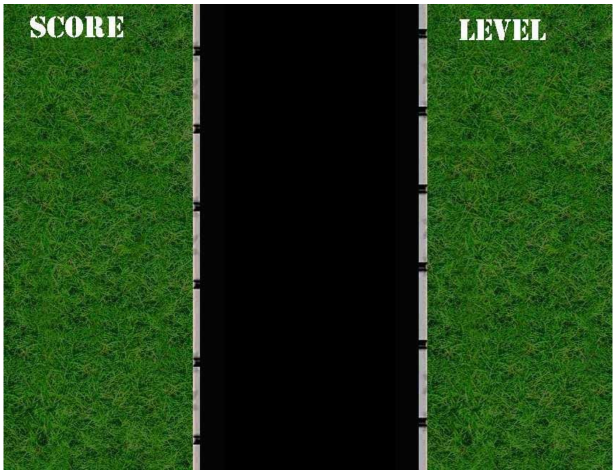


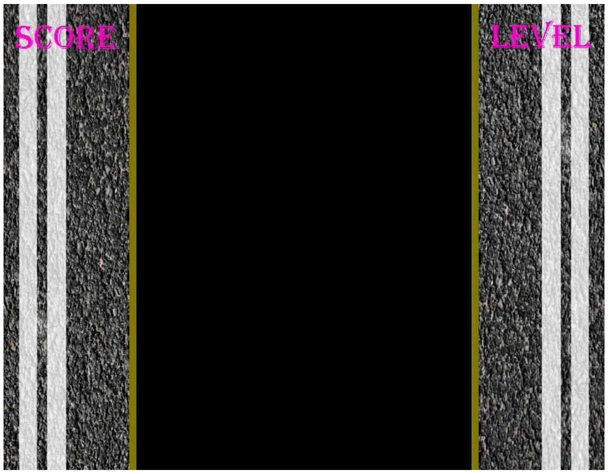


Level Selection:

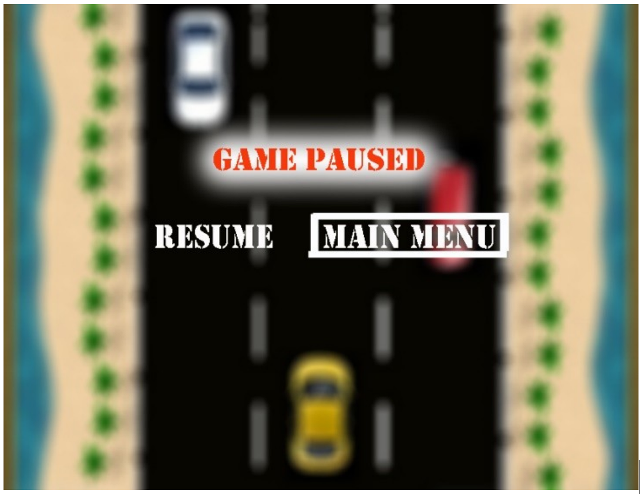


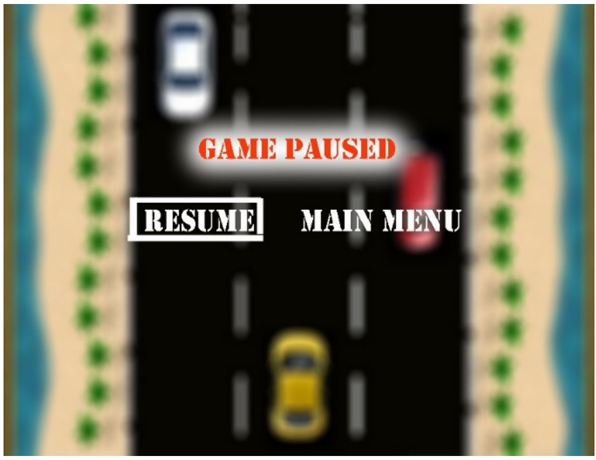
Background:

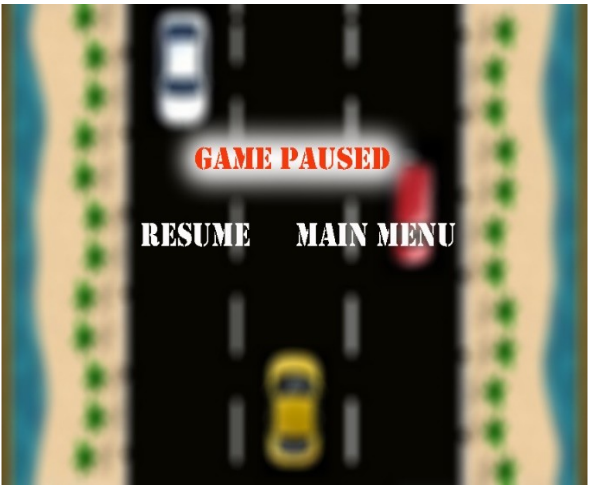




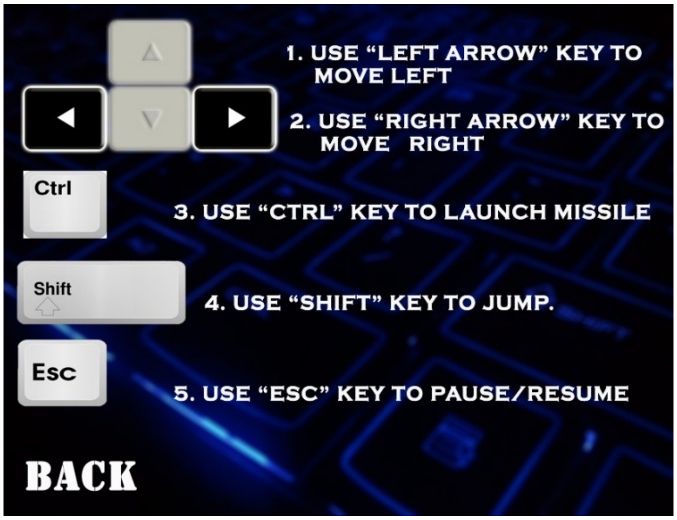
Pause Menu:





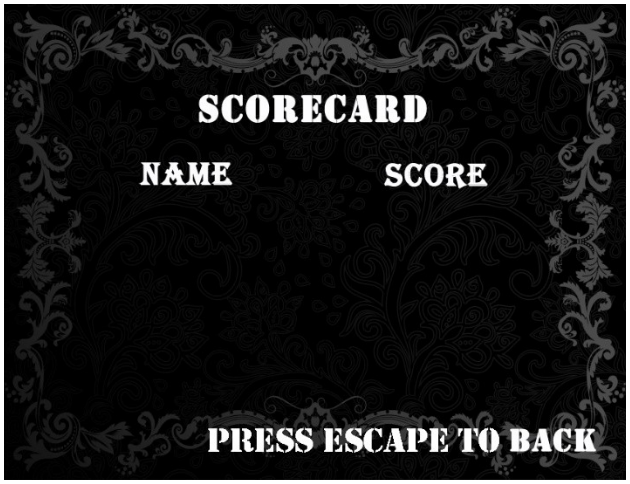


Instruction:



1. USE “LEFT ARROW” KEY TO MOVE LEFT
2. USE “RIGHT ARROW” KEY TO MOVE RIGHT
3. USE “CTRL” KEY TO LAUNCH MISSILE
4. USE “SHIFT” KEY TO JUMP.
5. USE “ESC” KEY TO PAUSE/RESUME

Scorecard:

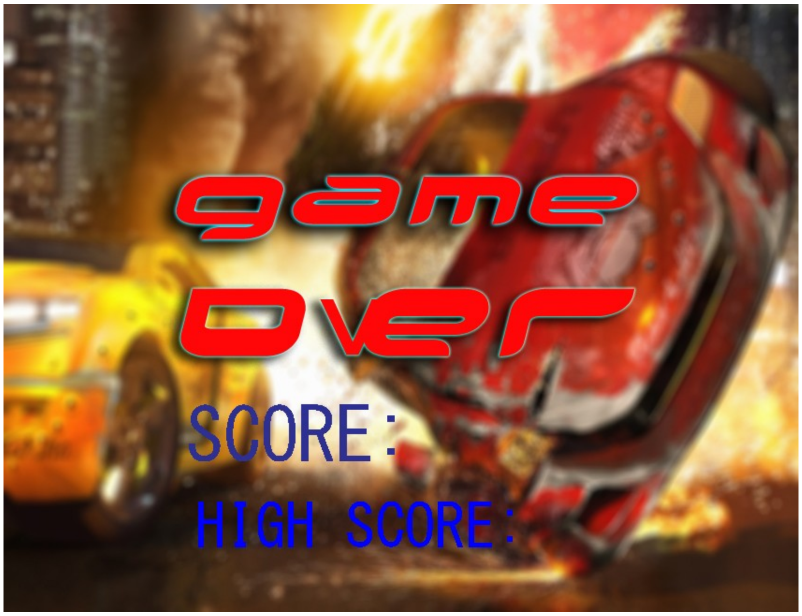

SCORECARD

NAME

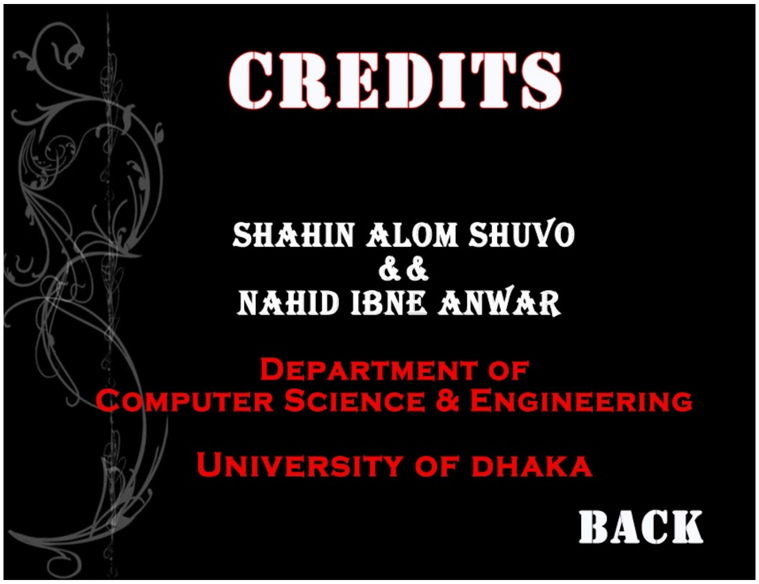
SCORE

PRESS ESCAPE TO BACK

Game Over:



Credits:



CREDITS

SHAHIN ALOM SHUVO
&&
NAHID IBNE ANWAR

DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING

UNIVERSITY OF DHAKA

Game Structure:

The game structure is so amazing. After opening the program, it takes the user to the main menu. From the main menu, the user can take one of four possible options. These options are the “New Game” submenu, the “Instructions” screen, the “Score board” screen, ”Credits” and the “Exit” option.

In the new game submenu, the user can choose between the two modes of the game. Once a choice has been made, the program will take the user to his desired destination.

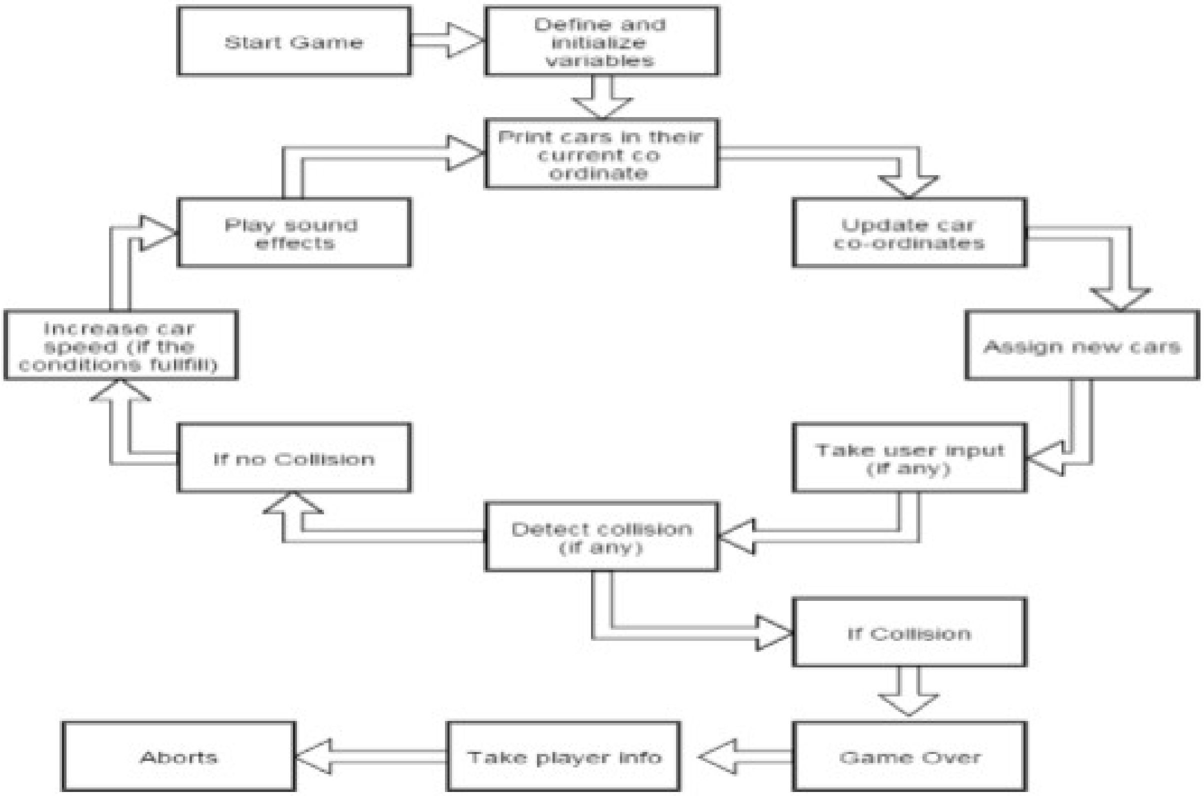
The instruction screen shows the basic instructions to play this game.

The leader board feature stores nine topmost scores from each level. However, the feature only prints the top three results on the screen.

During game play, the user can use the pause function at anytime and can either resume or abort the game.

All menus, submenus and options are completely reversible, that is the user can roam freely from any sector of the interface to the other.

How the level work:



Challenges in coding:

Avoiding impossible case:
As the game is an infinite game (at least until the player kills himself), sometimes an impossible case might rise where there is no escape and the game must end. This possible scenario was avoided by calculating the L.C.M of the car co-ordinates and updating the game accordingly.

Handling bugs:

The interactive menu was initially done by calling one function from another. This caused the functions to become recursive and sometime fall into deep holes of recursions. This was handled by re designing the whole menu operations using only break functions and looping and minimal function calling.
Just before the submission deadline, the game started crashing if played for a long time. This was happening for either not enough memory de-allocation or because of a possible infinite loop. I had to rebuild all the levels logic structure more efficiently overnight to overcome this unwanted problem.
However, I am happy to report that after all the hard works, the game appears to be bug free.

Old dated library:

The BGI graphics library was introduced in 1987 and the last C++ environment which supports BGI was published in 1997 . This is an old graphics library and it has limited capability. For example, BGI does not support .WAV files on its own; I had to use the winmm.lib too.

The functions used in the source code:

User defined functions:

```
void start();
void menu();
int three_lane();
void instruction();
void scorecard();
void credits();
void gameover();
void exit();
int pause1();
int pause2();
void levelselection();
void run_car();
void run_car_2();
int updatecar();
int updatecar_2();
int five_lane();
```

Functions from “graphics.h”:

01. void cleardevice (void); Used for clearing the screen.

02. void clearmouseclick(int kind); To clear any unprocessed mouse operation of a special kind.

03. void closegraph (void); Used while de-allocating memory.

04. void bar (int left, int top, int right, int bottom); Used for clearing a specific part of the screen.

05. void delay (int millisec); Used in various scenarios. This functions stops the execution of the program for a certain amount of time period.

06. void initwindow (int width, int height); Used for initializing graphical operations.

07. bool ismouseclick(int kind); To detect any unprocessed mouse operation of a special kind.

08. int kbhit (void); To detect any unprocessed keyboard input in the buffer.

09. int mousex (void); For getting the x co-ordinate of the cursor at a certain time.

10. int mousey (void); For getting the y co-ordinate of the cursor at a certain time.

11. void outtextxy (int x, int y, char *textstring); Used for printing scores and other strings.

12. void setcolor (int color); To set the current color.

13. void setfillstyle (int pattern, int color); To set the current filling style.

14. void settextstyle (int font, int direction, int charsize); To set the current text stile.

15. PlaySound("file/sound/click.wav", NULL, SND_ASYNC);   [from winmm.lib] To play a .WAV sound.

16. void readimagefile(const char* title=NULL, int left=0, int right=0, int right=INT_MAX, int bottom=INT_MAX); To read an image on the screen.

The list of header files:
Start.h

Overview and conclusion:
The project was a very good way to encourage young students to develop their skills and make something useful with their knowledge. This project is a perfect example of all the exceptional things that can be done with some basic C programming.
This project will also encourage young minds who want to be a game developer in the future. The project promotes originality and creativity and the skill of working in a team.
The project also summarizes the basic tools of C language as they had to be implemented while making the game.

The Source Code:

```
#include<stdio.h>
#include<graphics.h>
#include<string.h>
#include<windows.h>
#include "start.h"
int x=850,y=650;
char ch;
void start();
void menu();
int normalmode();
int speedmode();

void instruction();
void scorecard();
void credits();
void gameover();
void exit();
int pause1();
int pause2();
void run_car1();
int enemy_car1();
void run_car_2();
int enemy_car_2();
void levelselection();
//Main function starts from here.
main()
{
    //Calling start(); from user defined header file "start.h"
    //start();
    //calling menu function.
     initwindow(850,650,"Traffic Racing");
    menu();
    return 0;
}
//level select
void levelselection()
```

```
{
    readimagefile("img/ngm/levelselect.jpg",0,0,850,650);
    while(1)
    {
        int x,y;
        getmouseclick(WM_LBUTTONDOWN,x,y);
        if(x>=50 && y>= 107 && x<=283 && y<=161)
        {
            PlaySound(TEXT("snd/sc.wav"),NULL,SND_SYNC);
            normalmode();
        }
        else if(x>=50 && y>= 270 && x<=281 && y<=325)
        {
            PlaySound(TEXT("snd/sc.wav"),NULL,SND_SYNC);
            speedmode();

        }

       else if(kbhit())
        {
        ch=getch();
        while(kbhit())
            getch();
        if(ch==27)
        {
            PlaySound(TEXT("snd/sc.wav"),NULL,SND_SYNC);
            menu();

        }
        }
         else levelselection();



    }

}
//gameover function.
void gameover()
```

```
{
    readimagefile("img/gameover/go1.jpg",0,0,850,650);
    while(!ismouseclick(WM_LBUTTONDOWN))
    {
        int x,y;
        x= mousex();
        y= mousey();
        if(x>=250 && y>=214 && x<=576 && y<=257 )
        {
            readimagefile("img/gameover/go2.jpg",0,0,850,650);
        }
        else if(x>=254 && y>=334 && x<=570 && y<=376 )
        {
            readimagefile("img/gameover/go3.jpg",0,0,850,650);
        }
        else readimagefile("img/gameover/go1.jpg",0,0,850,650);
        getmouseclick(WM_LBUTTONDOWN,x,y);
        if(x>=250 && y>=214 && x<=576 && y<=257 )
        {
            mciSendString("play snd/sc.wav",0,0,0);
            levelselection();
        }
        else if(x>=254 && y>=334 && x<=570 && y<=376 )
        {
            mciSendString("play snd/sc.wav",0,0,0);
            menu();
        }

    }
}
//game pause function.
int pause1()
{
    readimagefile("img/ngm/pause1.jpg",0,0,850,650);
    while(!ismouseclick(WM_LBUTTONDOWN))
    {
        int x,y;
        x= mousex();
        y= mousey();
        if(x>=200 && y>=292 && x<=360 && y<=324 )
        {
            readimagefile("img/ngm/resume.jpg",0,0,850,650);
        }
        else if(x>=428 && y>=292 && x<=656 && y<=324 )
        {
            readimagefile("img/ngm/pause2.jpg",0,0,850,650);
        }
        else  readimagefile("img/ngm/pause1.jpg",0,0,850,650);
        getmouseclick(WM_LBUTTONDOWN,x,y);
        if(x>=428 && y>=292 && x<=656 && y<=324 )
        {
            mciSendString("play snd/sc.wav",0,0,0);
            menu();
        }
        else if(x>=200 && y>=292 && x<=360 && y<=324 )
        {
            mciSendString("play snd/sc.wav",0,0,0);
            mciSendString("resume snd/back.wav",0,0,0);
            readimagefile("img/ngm/back4.jpg",0,0,850,650);
            clearmouseclick(WM_LBUTTONDOWN);
            return 1;
        }
        else if(kbhit())
        {
            ch=getch();
            if(ch==27)
            {
                readimagefile("img/ngm/back4.jpg",0,0,850,650);
                clearmouseclick(WM_LBUTTONDOWN);
                mciSendString("resume snd/back.wav",0,0,0);
                return 1;
            }
        }
        else clearmouseclick(WM_LBUTTONDOWN);
    }
}
int pause2()
{
```

```
readimagefile("img/ngm/pause1.jpg",0,0,850,650);
while(!ismouseclick(WM_LBUTTONDOWN))
{
    int x,y;
    x= mousex();
    y= mousey();
    if(x>=200 && y>=292 && x<=360 && y<=324 )
    {
        readimagefile("img/ngm/resume.jpg",0,0,850,650);
    }
    else if(x>=428 && y>=292 && x<=656 && y<=324 )
    {
        readimagefile("img/ngm/pause2.jpg",0,0,850,650);
    }
    else  readimagefile("img/ngm/pause1.jpg",0,0,850,650);
    getmouseclick(WM_LBUTTONDOWN,x,y);
    if(x>=428 && y>=292 && x<=656 && y<=324 )
    {
        mciSendString("play snd/sc.wav",0,0,0);
        menu();
    }
    else if(x>=200 && y>=292 && x<=360 && y<=324 )
    {
        mciSendString("play snd/sc.wav",0,0,0);
        mciSendString("resume snd/back.wav",0,0,0);
        readimagefile("img/ngm/back1.jpg",0,0,850,650);
        clearmouseclick(WM_LBUTTONDOWN);
        return 1;
    }
    else if(kbhit())
    {
        ch=getch();
        if(ch==27)
        {
            readimagefile("img/ngm/back1.jpg",0,0,850,650);
            clearmouseclick(WM_LBUTTONDOWN);
            mciSendString("resume snd/back.wav",0,0,0);
            return 1;
        }
        }
        else clearmouseclick(WM_LBUTTONDOWN);
    }
}
//main menu function.
void menu()
{
    int x,y;
    //checking x,y,coordinates.
    while(1)
    {
        int z=0;
        readimagefile("img/menu/menu.jpg",0,0,850,650);
        while(!ismouseclick(WM_LBUTTONDOWN))
        {
            x= mousex();
            y= mousey();
            if(x>=621 && y>= 244 && x<=850 && y<=307)
            {
                readimagefile("img/menu/newgame.jpg",0,0,850,650);
                if(z==0)
                {
                    mciSendString("stop snd/ch.wav",0,0,0);
                    mciSendString("play snd/ch.wav",0,0,0);
                }
                z=1;
            }
            else if(x>=609 && y>= 329 && x<=850 && y<=393)
            {
                readimagefile("img/menu/instruction.jpg",0,0,850,650);
                if(z==0)
                {
                    mciSendString("stop snd/ch.wav",0,0,0);
                    mciSendString("play snd/ch.wav",0,0,0);
                }
                z=1;
            }
            else if(x>=599 && y>= 413 && x<=850 && y<=480)
            {
```

```
        readimagefile("img/menu/scorecard.jpg",0,0,850,650);
        if(z==0)
        {
            mciSendString("stop snd/ch.wav",0,0,0);
            mciSendString("play snd/ch.wav",0,0,0);
        }
        z=1;
    }
    else if(x>=590 && y>= 497 && x<=850 && y<=563)
    {
        readimagefile("img/menu/credits.jpg",0,0,850,650);
        if(z==0)
        {
            mciSendString("stop snd/ch.wav",0,0,0);
            mciSendString("play snd/ch.wav",0,0,0);
        }
        z=1;
    }
    else if(x>=575 && y>= 578 && x<=850 && y<=642)
    {
        readimagefile("img/menu/exit.jpg",0,0,850,650);
        if(z==0)
        {
            mciSendString("stop snd/ch.wav",0,0,0);
            mciSendString("play snd/ch.wav",0,0,0);
        }
        z=1;
    }
    else
    {
        readimagefile("img/menu/menu.jpg",0,0,850,650);
        z=0;
    }

}
//Selecting option
getmouseclick(WM_LBUTTONDOWN,x,y);
if(x>=621 && y>= 241 && x<=850 && y<=313)
{
```

```
            mciSendString("play snd/sc.wav",0,0,0);
            //normalmode();
            levelselection();
        }
        else if(x>=609 && y>= 326 && x<=850 && y<=398)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            instruction();
        }

        else if(x>=599 && y>= 410 && x<=850 && y<=485)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            scorecard();
        }
        else if(x>=590 && y>= 494 && x<=850 && y<=568)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            credits();
        }
        else if(x>=575 && y>= 575 && x<=850 && y<=650)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            exit();
        }
        else
            menu();
    }
}
//run_car function for load cars continuously.
void run_car1(int enemycartop[],char carname[][40], int
enemycarleft[])
{
    int i;
    for(i=0; i<3; i++)
        readimagefile(carname[i],enemycarleft[i],enemycartop[i]-
85,enemycarleft[i]+45,enemycartop[i]);
}
//Enemy_car function for read car file & count scores.
```

```
int enemy_car1(int enemycartop[],char carname [][40],int
enemycarleft[], int speed,int score)
{
    int i;
    for(i=0; i<3; i++)
    {
        if(enemycartop[i]<750)
        {
            bar(enemycarleft[i],enemycartop[i]-85,enemycarleft[i]
+45,enemycartop[i]-95);
            enemycartop[i]+=speed;
            setfillstyle(SOLID_FILL,BLACK);
        }
        else
        {
            sprintf(carname[i],"img/cars/%d.jpg",(rand()%20));
            enemycartop[i]=0;
            score++;
            int x=rand()%3;
            if(x==1)
                enemycarleft[i]=303;
            if(x==2)
                enemycarleft[i]=403;
            if(x==0)
                enemycarleft[i]=503;
            char scores[60];
            sprintf(scores,"%d",score);
            settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
            setcolor(WHITE);
            outtextxy(70,80,scores);
        }
    }
    return score;
}
struct sco
    {
        int val;
        char name[20];
    };
```

```
//This is the main game function.
int normalmode()
{
    int w=-
100,x=550,flag=0,f=0,tm,rm,lm,jump=0,y=0,chek=0,mval=2,jva
l=3;
    char c,carname[5][40],scores[7];
    char levels[1],mvals[1],jvals[1];
    int r=448,l=403,t=545,d=630,
speed=5,i,terminate=0,score=0,speedup=1,l3=503,level=0;
    readimagefile("img/ngm/back4.jpg",0,0,850,650);
    setfillstyle(SOLID_FILL,BLACK);
    readimagefile("img/ngm/f.jpg",l,t,r,d);
    readimagefile("img/ngm/r3.jpg",360,w,380,x);
    readimagefile("img/ngm/r3.jpg",472,w,488,x);
    mciSendString("play snd/start.wav",0,0,0);
    delay(3000);

    int enemycartop[5]= {0,-225,-450,-700,0},enemycarleft[6]=
{l3,l,l,r,l3,r};

    for(i=0; i<3; i++)
        sprintf(carname[i],"img/cars/%d.jpg",(rand()%20));
    while(!terminate)
    {
        mciSendString("play snd/back.wav",0,0,0);
        if(jump==0)
            readimagefile("img/ngm/f.jpg",l,t,r,d);
        readimagefile("img/ngm/r3.jpg",360,w,380,x);
        readimagefile("img/ngm/r3.jpg",472,w,488,x);
        w=w+7;
        x=x+7;
        if(x>750)
        {
            w=-100;
            x=550;
        }
        //Calling run_car();
        run_car1(enemycartop,carname,enemycarleft);
```

```
score=enemy_car1(enemycartop,carname,enemycarleft,speed,sc
ore);
        sprintf(levels,"%d",level);
        settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
        setcolor(WHITE);
        outtextxy(650,125,levels);
        sprintf(mvals,"%d",mval);
        settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
        setcolor(WHITE);
        outtextxy(77,424,mvals);
        sprintf(jvals,"%d",jval);
        settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
        setcolor(WHITE);
        outtextxy(70,260,jvals);

        if(y==0)
        {
            if(score%10==0)
            {
                if(speedup==1)
                {
                    level=level+30;
                    speedup=0;
                    mval++;
                    if(mval>5)
                        mval=5;
                        else if(mval<0){
                            mval=0;
                        }
                    speed=speed+2;
                    if(speed>10)
                        speed=10;
                    char horn[20];
                    sprintf(horn,"snd/%d.wav",rand()%7);
                }
            }
             if(score%15==0)
             {
```

```
     if(speedup==1)
     jval++;
     if(jval>5) jval=5;
 }
else if(score%10!=0) speedup=1;
if(GetAsyncKeyState(VK_SHIFT))
{
    if(jval>0){
            y=1;
    jump=1;
      jval--;
    }
}
if(GetAsyncKeyState(VK_CONTROL))
{
    if(mval>0){
    y=1;
    f=1;
    tm=t;
    lm=l;
    rm=r;
    mciSendString("stop snd/missile.wav",0,0,0);
    mciSendString("play snd/missile.wav",0,0,0);
    }
}
if(kbhit())
{
    c=getch();
    if(c==KEY_LEFT)
    {
        if(l>=394 && l<=540)
        {
            bar(l,t,r,d+2);
            l=l-100;
            r=r-100;
            readimagefile("img/ngm/f.jpg",l,t,r,d);
        }
```

```
                }
                else if(c==KEY_RIGHT)
                {
                    if(l>=294 && l<=420)
                    {
                        bar(l,t,r,d+2);
                        l=l+100;
                        r=r+100;
                        readimagefile("img/ngm/f.jpg",l,t,r,d);
                    }
                }
                else if(c==27)
                {
                    mciSendString("pause snd/back.wav",0,0,0);
                    pause1();
                }
            }
        }
        if(f==1)
        {
            bar(lm+14,tm,rm-12,tm+50);
            readimagefile("img/ngm/missile.gif",lm+15,tm-30,rm-
15,tm);
            tm=tm-30;
        }
        for(i=0; i<3; i++) //Printing missile fire.
        {
            if(enemycarleft[i]==l)
            {
                if((tm-20)-enemycartop[i]<=0)
                {
                    mciSendString("play snd/exp.mp3",0,0,0);

readimagefile("img/ngm/exp1.jpg",enemycarleft[i],enemycartop[i]
-100,enemycarleft[i]+45,enemycartop[i]);
                    delay(50);

readimagefile("img/ngm/exp2.jpg",enemycarleft[i],enemycartop[i]
-100,enemycarleft[i]+45,enemycartop[i]);
                    delay(100);
                    bar(enemycarleft[i],enemycartop[i]-100,enemycarleft[i]+45,enemycartop[i]+50);
                    enemycartop[i]=-500;
                    tm=750;
                    f=0;
                    y=0;
                    mval--;
                    break;
                }
            }
        }
        //Jump feature runs here.
        if(jump==1)
        {
            readimagefile("img/ngm/f.jpg",l-7,t-30,r+7,d);
            chek++;
            if(chek>35)
            {
                bar(l-20,t-50,r+20,850);
                enemycartop[i]=0;
                chek=0;
                jump=0;
                y=0;
                bar(390,450,460,650);
            }
        }
        //Detecting collisions..
        if (jump==0)
        {
            for(i=0; i<3; i++)
            {
                if(enemycarleft[i]==l)
                {
                    if((t-enemycartop[i]<=0)||(t-enemycartop[i]<-85))
                    {
                        delay(500);
                        terminate=1;
                    }
                }
            }
        }
        // calling gameover();
        if (terminate ==1)
        {
            readimagefile("img/ngm/exp2.jpg",l,t,r,d);
            delay(200);
            mciSendString("stop snd/back.wav",0,0,0);
            mciSendString("play snd/exp.mp3",0,0,0);

        }


    }
    while(kbhit())getch();

    setcolor(WHITE);
    settextstyle(8,HORIZ_DIR,4);
    readimagefile("file/board.jpg",0,0,850,650);
    outtextxy(265,150,"Enter Your Name:");

    int a=0;
    char write[50];
    write[0]='\0';
    while(kbhit()) getch();
    while(1)
    {
        char c=getch();
        if(c==13)
            break;
        else if(c==8)
            a-=2;
        else
            write[a]=c;
        a++;

        write[a]='\0';
        bar(280,200,570,250);
        outtextxy(290,200,write);
```

```
}


int j,k;
struct sco entry[10],temp;
FILE *in,*fp,*wr1;
in = fopen("scoresfile.txt","a");
fprintf(in,"%d %s\n",score,write);
fclose(in);


fp=fopen("scoresfile.txt","r");
for(k=0; k<10; k++)
    fscanf(fp,"%d %s",&entry[k].val,&entry[k].name);
fclose(fp);


for(j=9; j>0; j--)
    if(entry[j].val>entry[j-1].val)
    {
        temp=entry[j];
        entry[j]=entry[j-1];
        entry[j-1]=temp;
    }
wr1=fopen("scoresfile.txt","w");
for(j=0; j<9; j++)
    fprintf(wr1,"%d %s\n",entry[j].val,entry[j].name);
fclose(wr1);

PlaySound(NULL, 0, 0);
char highscores[10];
readimagefile("img/gameover/gameover5.jpg",0,0,850,650);
int s=score,hs=entry[0].val;
sprintf(scores,"%d",s);
settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
setcolor(WHITE);
outtextxy(450,432,scores);
sprintf(highscores,"%d",hs);
settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
```

```
    setcolor(WHITE);
    outtextxy(613,533,highscores);
    if(s>=entry[0].val)
    {
    setcolor(RED);
    settextstyle(8,HORIZ_DIR,4);
    outtextxy(30,593,"Congrats!! You made a new high score!!");

    }

    while(1)
    {
        char ch=getch();
        if(ch==13)
        gameover();
    }
    return 1;
}
//speedmode
void run_car_2(int enemycartop2[],char carname[][40], int
enemycarleft2[])
{
    int i;
    for(i=0; i<5; i++)
        readimagefile(carname[i],enemycarleft2[i],enemycartop2[i]-
85,enemycarleft2[i]+45,enemycartop2[i]);
}
int enemy_car_2(int enemycartop2[],char carname [][40],int
enemycarleft2[], int speed,int score)
{
    int i;
    for(i=0; i<5; i++)
    {
        if(enemycartop2[i]<750)
            {
            bar(enemycarleft2[i],enemycartop2[i]-85,enemycarleft2[i]
+45,enemycartop2[i]-125);
            enemycartop2[i]+=speed;
            setfillstyle(SOLID_FILL,BLACK);
            }
        else
            {
            int v=(rand()%20);
            sprintf(carname[i],"img/cars/%d.jpg",v);
            enemycartop2[i]=0;
            score++;
            bar(205,630,655,650);
            int x=rand()%5;
            if(x==1)
                enemycarleft2[i]=205;
            if(x==2)
                enemycarleft2[i]=305;
            if(x==0)
                enemycarleft2[i]=405;
            if(x==3)
                enemycarleft2[i]=505;
            if(x==4)
                enemycarleft2[i]=605;
            char scores[60];
            sprintf(scores,"%d",score);
            settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
            setcolor(WHITE);
            outtextxy(20,80,scores);
        }
    }
    return score;
}
int speedmode()
{
    int w=-100,x=550,level=0;
    char levels[1];
    int r=450,l=405,t=545,d=630,
speed=5,i,terminate=0,score=0,speedup=1,x1=205,x2=305,x3=
405,x4=505,x5=605;

    readimagefile("img/ngm/back6.jpg",0,0,850,650);

    readimagefile("img/ngm/f2.jpg",l,t,r,d);
    setfillstyle(SOLID_FILL,BLACK);
    mciSendString("play snd/start.wav",0,0,0);
    int enemycartop2[7]= {0,-550,-225,-450,-700,-
900,0},enemycarleft2[7]= {x2,x4,x1,x3,x1,x4,x2};
    char c,carname[5][40],scores[7];
    for(i=0; i<5; i++)
        sprintf(carname[i],"img/cars/%d.jpg",(rand()%20));
    while(!terminate)
    {
        readimagefile("img/ngm/r3.jpg",280,w,285,x);
        readimagefile("img/ngm/r3.jpg",378,w,383,x);
        readimagefile("img/ngm/r3.jpg",476,w,481,x);
        readimagefile("img/ngm/r3.jpg",574,w,579,x);
        w=w+15;
        x=x+15;
        if(x>750)
        {
            w=-100;
            x=550;
        }

        run_car_2(enemycartop2,carname,enemycarleft2);

score=enemy_car_2(enemycartop2,carname,enemycarleft2,spee
d,score);
        sprintf(levels,"%d",level);
        settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
        setcolor(WHITE);
        outtextxy(710,85,levels);
        if(score%20==0)
        {
            if(speedup==1)
                {
                    level++;
                speedup=0;
            }
        }
        else if(score%20!=0) speedup=1;
        if(kbhit())
```

```
{

    c=getch();
    if(c==KEY_LEFT)
    {
        if(l>=220 && l<=700)
        {
            bar(l,t,r,d+2);
            l=l-100;
            r=r-100;
            readimagefile("img/ngm/f2.jpg",l,t,r,d);
        }


    }
    else if(c==KEY_RIGHT)
    {
        if(l>=200 && l<=590)
        {
            bar(l,t,r,d+2);
            l=l+100;
            r=r+100;
            readimagefile("img/ngm/f2.jpg",l,t,r,d);
        }
    }

    else if(c==27)
    {
        pause2();
    }
    else if(c==115){
        speed=speed+5;
        if (speed>40)
            speed=40;
    }
     else if(c==100){
        speed=speed-5;
        if(speed<5);
            speed=5;
```

```
             }
        }
        for(i=0; i<5; i++) // accident
        {
            if(enemycarleft2[i]==l)
            {
                if((t-enemycartop2[i]<=0)||(t-enemycartop2[i]<-85))
                {
                    delay(200);
                    terminate=1;
                    mciSendString("stop snd/start.wav",0,0,0);

                }
            }
        }
}
while(kbhit())getch();

    setcolor(WHITE);
    settextstyle(8,HORIZ_DIR,4);
    readimagefile("file/board2.jpg",0,0,850,650);
    outtextxy(265,150,"Enter Your Name:");

    int a=0;
    char write[50];
    write[0]='\0';
    while(kbhit()) getch();
    while(1)
    {
        char c=getch();
        if(c==13)
            break;
        else if(c==8)
            a-=2;
        else
            write[a]=c;
        a++;

        write[a]='\0';
    bar(280,200,570,250);
    outtextxy(290,200,write);
}


int j,k;
struct sco entry2[10],temp;
FILE *in,*fp,*wr1;
in = fopen("scorefile2.txt","a");
fprintf(in,"%d %s\n",score,write);
fclose(in);


fp=fopen("scorefile2.txt","r");
for(k=0; k<10; k++)
    fscanf(fp,"%d %s",&entry2[k].val,&entry2[k].name);
fclose(fp);


for(j=9; j>0; j--)
    if(entry2[j].val>entry2[j-1].val)
    {
        temp=entry2[j];
        entry2[j]=entry2[j-1];
        entry2[j-1]=temp;
    }
wr1=fopen("scorefile2.txt","w");
for(j=0; j<9; j++)
    fprintf(wr1,"%d %s\n",entry2[j].val,entry2[j].name);
fclose(wr1);
char highscores[10];
readimagefile("img/gameover/gameover5.jpg",0,0,850,650);
int s=score,hs=entry2[0].val;
sprintf(scores,"%d",s);
settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
setcolor(WHITE);
outtextxy(450,432,scores);
sprintf(highscores,"%d",hs);
settextstyle(GOTHIC_FONT,HORIZ_DIR,6);
```

```
    setcolor(WHITE);
    outtextxy(613,533,highscores);
    if(s>=entry2[0].val)
    {
    setcolor(WHITE);
    settextstyle(8,HORIZ_DIR,4);
    outtextxy(30,593,"Congrats!! You made a new high score!!");

    }

    while(1)
    {
        char ch=getch();
        if(ch==13 )
            gameover();
    }
    return 1;

}
//score card function
void scorecard()
{
    while(1)
    {
        char c;
        readimagefile("img/menu/scorecard1.jpg",0,0,850,650);
        setcolor(BLUE);
        settextstyle(8,HORIZ_DIR,4);
        outtextxy(340,270,"Normal Mode");
        setcolor(GREEN);
        outtextxy(350,380,"Speed Mode");
        //Normal mode
        setcolor(WHITE);
        settextstyle(8,HORIZ_DIR,3);
        FILE *leader1;
        leader1=fopen("scoresfile.txt","r");
        int i=3, j=300;
```

```
while(i--)
{
    int val;
    char name[30], vals[60], names[40];
    fscanf(leader1,"%d %s",&val,&name);
    sprintf(names,"%s                              ",name,val);
    sprintf(vals,"%d",val);
    outtextxy(200,j,names);
    outtextxy(550,j,vals);
    j+=20;
}

fclose(leader1);
//Speed mode
setcolor(WHITE);
settextstyle(8,HORIZ_DIR,3);
FILE *leader2;
leader2=fopen("scorefile2.txt","r");
int p=3, q=420;


while(p--)
{
    int val;
    char name[30], vals[60], names[40];
    fscanf(leader2,"%d %s",&val,&name);
    sprintf(names,"%s                              ",name,val);
    sprintf(vals,"%d",val);
    outtextxy(200,q,names);
    outtextxy(550,q,vals);
    q+=20;
}

fclose(leader2);
c=getch();


if(c==27)
{
```

```
            PlaySound(TEXT("snd/sc.wav"),NULL,SND_SYNC);
            menu();
        }
    }


}

//Instruction function
void instruction()
{
    initwindow(850,650,"Traffic Racing");
    while(1)
    {
        readimagefile("img/menu/instruction1.jpg",0,0,850,650);
        while(!ismouseclick(WM_LBUTTONDOWN))
        {
            x=mousex();
            y=mousey();
            if(x>=25 && y>=583 && x<=152 && y<=630)
            {

readimagefile("img/menu/instruction2.jpg",0,0,850,650);
            }
            else
readimagefile("img/menu/instruction1.jpg",0,0,850,650);
        }
        getmouseclick(WM_LBUTTONDOWN,x,y);
        int x,y;
        x=mousex();
        y=mousey();
        if(x>=25 && y>=583 && x<=152 && y<=630)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            menu();
        }
        else instruction();
    }
}
```

```
//credits function.
void credits()
{
    initwindow(850,650,"Traffic Racing");
    readimagefile("menu/c/credits0.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits1.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits2.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits3.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits4.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits5.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits6.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits7.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits8.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits9.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits10.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits11.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits12.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits13.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits14.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits15.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits16.jpg",0,0,850,650);
    delay(5);
    readimagefile("img/crd/credits17.jpg",0,0,850,650);
```

```
delay(5);
readimagefile("img/crd/credits18.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits19.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits20.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits21.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits22.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits23.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits24.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits25.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits26.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits27.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits28.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits29.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits30.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits31.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits32.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits33.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits34.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits35.jpg",0,0,850,650);
delay(5);
readimagefile("img/crd/credits36.jpg",0,0,850,650);
while(1)
```

```
    {

        while(!ismouseclick(WM_LBUTTONDOWN))
        {
            x=mousex();
            y=mousey();
            if(x>=632 && y>=552 && x<=804 && y<=617)
            {

                readimagefile("img/crd/creditsback.jpg",0,0,850,650);
            }
            else
            {
                readimagefile("img/crd/credits37.jpg",0,0,850,650);
            }
        }
        getmouseclick(WM_LBUTTONDOWN,x,y);
        int x,y;
        x=mousex();
        y=mousey();
        if(x>=632 && y>=552 && x<=804 && y<=617)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            menu();
        }
        else credits();
    }
}


//Exit function to get player choice.
void exit()
{

    while(1)
    {
        readimagefile("img/menu/exitmenu.jpg",0,0,850,650);
        while(!ismouseclick(WM_LBUTTONDOWN))
        {
            x=mousex();
            y=mousey();
            if(x>=241 && y>=262 && x<=375 && y<=326)
            {
                readimagefile("img/menu/exitmenuy.jpg",0,0,850,650);
            }
            else if(x>=488 && y>=263 && x<=592 && y<=333)
            {
                readimagefile("img/menu/exitmenun.jpg",0,0,850,650);
            }
            else
readimagefile("img/menu/exitmenu.jpg",0,0,850,650);
        }
        getmouseclick(WM_LBUTTONDOWN,x,y);
        int x,y;
        x=mousex();
        y=mousey();
        if(x>=241 && y>=262 && x<=375 && y<=326)
            exit(0);
        if(x>=488 && y>=263 && x<=592 && y<=333)
        {
            mciSendString("play snd/sc.wav",0,0,0);
            menu();
        }
        else exit();
    }
}
//Thats all. Thank you :)
```

Special thanks to:
01. Mr. Md. Shiplu Hawlader Lecturer, CSEDU
02. Hasnain Heickal Jami Lecturer, CSEDU

References:

[1] https://en.wikipedia.org/wiki/Borland_Graphics_Interface [2] http://www.stereosue.com/ [3] http://www.thewallpapers.org [4] https://en.wikipedia.org/wiki/Borland_Graphics_Interface [5] http://www.stackoverflow.com [6] Borland Graphics Interface for windows: https://www.cs.colorado.edu/~main/bgi/doc/ [7] http://www.programmingsimplified.com [8] www.youtube.com [9] http://maps.google.com [10] C The Complete Reference (4th Edition) By Herbert Schildt [11] Programming in ANSI C By E.Balaguruswami

The End