

ABSTRACT:

A website with incorporated electronic games is called E-Game Arena. The customer has a large selection of options to choose the game they wish to play. Tic-tac-toe and the dino game are the two games included in this mini-project, and the user can select whichever one they want to play.

Trex, a cute-looking dino that can run and hop, is the creature in the dino game. In the game's dessert section, the player must rescue the dinosaur from the cactus obstacle in order to get points.

Two players mark the spaces on a three-by-three grid with an X or an O in the game of tic-tac-toe, noughts and crosses, or Xs and Os. The winner is the player who successfully arranges three of their marks in a row that is either horizontal, vertical, or diagonal.

The foundational functions of this mini-program, such as win (), platform (), color (), and image (), are included. It also contains fundamental C principles. The application lets the user do the actions necessary to play the game and runs on the command prompt.

The objective of this mini-project is to provide the user with an interactive system with which a player can play. It develops problem-solving skills and acts as a stress relief tool.

TABLE OF CONTENTS:

1. INTRODUCTION

2. DESIGN AND IMPLEMENTATION

2.1 CODE

3. TESTING

3.1 CODE ERRORS AND DEBUGGING

4. RESULT AND ANALYSIS

4.1 CODE OUTPUT

5. CONCLUSION AND FUTURE ENHANCEMENT

6. REFERENCES

1. INTRODUCTION

Dennis M. Ritchie created the C programming language in 1972 at Bell Telephone Laboratories to create the UNIX operating system. It is a general-purpose, procedural, imperative programming language. The most often used computer language is C. Along with the equally popular and widely used Java programming language, which is utilised by the majority of contemporary software programmers, it consistently ranks at the top of the popularity scale. For students and working professionals who want to excel as software engineers, especially those who work in the software development field, proficiency in the C programming language is a MUST. Key advantages of learning C Programming:

- Easy to learn
- Structured language
- It produces efficient programs
- It can handle low-level activities
- It can be compiled on a variety of computer platforms

The e-game arena is a C-based application that includes two well-known games: tic-tac-toe and dino game. It is one of the programs that are user-friendly and are easy to implement for beginners and incorporates a variety of C language capabilities. It uses standard libraries like `stdin`, `conio`, and `windows` and includes `std` library functions, loops, user-defined functions, and switch cases.

Esports is a booming global industry where skilled video gamers play competitively. In the same way that traditional sports have competitions in baseball, basketball, and football, esports encompasses competitions across a variety of video games. This project was developed to be a simple and strong example of the use of C programming in gaming.

2. DESIGN AND IMPLEMENTATION

2.1 CODE :

The code includes 3 main parts:

1. The main function
2. The dino game
- 3 .the tic-tac-toe game

The code is as follows:

```
#include <windows.h>
```

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
void cacti1(int p);
```

```
void cacti2(int p);
```

```
void dino_run(int x);
```

```
void dead_dino(int x);
```

```
void loss(int x,int y);
```

```
void sun();
```

```
void trex(int x,int y);
```

```
void cloud(int x,int y);
```

```
void bird1(int x,int y);
```

```
void bird2(int x,int y);
```

```
void red();
```

```
void yellow();
```

```
void green();
```

void blue();

void purple();

void cyan();

void reset();

int dino();

int tictactoe();

int win();

void platform();

void tictac(int x,int y);

COORD c = {0, 0};

void setxy (int x, int y)

{

c.X = x; c.Y = y; // Set X and Y coordinates

SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), c);

}

char block[10] = { 'o', '1', '2', '3', '4', '5', '6', '7', '8', '9' };

int tictactoe()

{

char c;

int i,j;

system("cls");

tictac(35,18);

char ch;

```
ch=getchar();
if(ch=='\n')
{
j=1;
}
while (j)
{
    int player = 1, i, choice;

    char mark;
    do
    {
        platform();
        player = (player % 2) ? 1 : 2;

        green();
        printf(" PLAYER %d -> The number you choose is : ", player);
        reset();
        scanf("%d", &choice);

        mark = (player == 1) ? 'X' : 'O';

        if (choice == 1 && block[1] == '1')
            block[1] = mark;
```

else if (choice == 2 && block[2] == '2')

block[2] = mark;

else if (choice == 3 && block[3] == '3')

block[3] = mark;

else if (choice == 4 && block[4] == '4')

block[4] = mark;

else if (choice == 5 && block[5] == '5')

block[5] = mark;

else if (choice == 6 && block[6] == '6')

block[6] = mark;

else if (choice == 7 && block[7] == '7')

block[7] = mark;

else if (choice == 8 && block[8] == '8')

block[8] = mark;

else if (choice == 9 && block[9] == '9')

block[9] = mark;


```
    else if (block[3] == block[6] && block[6] == block[9])
        return 1;

    else if (block[1] == block[5] && block[5] == block[9])
        return 1;

    else if (block[3] == block[5] && block[5] == block[7])
        return 1;

    else if (block[1] != '1' && block[2] != '2' && block[3] != '3' &&
        block[4] != '4' && block[5] != '5' && block[6] != '6' && block[7]
        != '7' && block[8] != '8' && block[9] != '9')

        return 0;
    else
        return - 1;
}

//board
void platform()
{
    system("cls");
    red();
    printf("\n\n\t\t\t\t\tTIC TAC TOE\n\n\n");
    cyan();
```

```
printf("\t\t\tRULES:ENTER 1 NO. AT A TIME AND WIN THE GAME  
USING YOUR WITS : ) !!!\n\n\n\n");
```

```
reset();
```

```
purple();
```

```
printf("\t\t\t\t\t Player 1 (X) - Player 2 (O)\n\n\n\n");
```

```
reset();
```

```
red();
```

```
printf("                /   /   \n");
```

```
yellow();
```

```
printf("                %c | %c | %c \n", block[1],  
block[2], block[3]);
```

```
red();
```

```
printf("                ____/____/____\n");
```

```
printf("                /   /   \n");
```

```
yellow();
```

```
printf("                %c | %c | %c \n", block[4],  
block[5], block[6]);
```

```
red();
```

```
printf("                ____/____/____\n");
```

```
printf("                /   /   \n");
```

```
        yellow();

        printf("                %c | %c | %c\n", block[7],
        block[8], block[9]);

        red();

        printf("                /   /   \n\n");

        reset();
    }

void tictac(int x,int y)
{
    blue();
    setxy(x,y-1);

    printf("MMMM88MMMM MM ,adPPYba,  MMMM88MMMM ,adPba,
,adPPYba,  MMMM88MMMM ,abdPPYba,  MMM88MMM");

    setxy(x,y);

    printf(" 88  88 a8P          88  a8  8a a8P          88  a8P
P8a 88");

    yellow();

    setxy(x,y+1);

    printf(" 88  88 8PP          88  88  88 8PP          88
8PP  PP8 88");

    setxy(x,y+2);
```

```
        printf(" 88 88 8PP      88 88MM8MM88 8PP      88
8PP  PP8 88MM88");
        setxy(x,y+3);
        printf(" 88 88 8PP      88 88 88 8PP      88
8PP  PP8 88");
        blue();
        setxy(x,y+4);
        printf(" 88 88 \"8b,      88 88 88 \"8b,      88
\"8b, ,b8\" 88");
        setxy(x,y+5);
        printf(" 88 MM \"Ybbd8\"      88 MM MM \"Ybbd8\"
88 \"8Ybbd8\"^ MMM88MMM");
        purple();
        setxy(x,y+10);
        printf("                WIN THE GAME WITH YOUR WIT\n");
        green();
        setxy(x,y+12);
        printf("                press enter to start the game");
        reset();

}

int main()
{
```

```
int choice,j;
char ch;
system("cls");
yellow();
setxy(70,15);
printf("Pick the game you wanna play ");
setxy(80,16);
printf("1.Trex");
setxy(80,17);
printf("2.TICTACTOE");
setxy(80,18);
printf("3.Exit");
setxy(80,20);
printf("HAVE FUN!!    ");
reset();
scanf("%d",&choice);
ch=getchar( );
switch(choice)
{
    case 1:dino();
    break;
    case 2:tictactoe();
    break;
    case 3:return 0;
}
```

```
        return 0;
    }

    int dino()
    {
        char ch;
        int i,j;
        int p=150,q=0,x;
        int score=0;
        int k=0;
        system("cls");
        trex(65,18);
        ch=getchar( );
        if(ch=="\n")
        {
            j=1;
        }
        while(j)
        {

            for (;p>=5,q>=5,!kbhit();p=p-5,q=q-5)
            {
                system("cls");
                for(int i=7;i<160;i++)
                {
```

```
        setxy(i,35);
        printf("-");
    }
    setxy(140,5);
    yellow();
    printf("SCORE : %d",score);
    reset();
    sun();
    cloud(110,12);
    cloud(15,4);
    bird1(40,14);
    bird1(130,6);
    dino_run(0);
    cacti1(p);
    if(p==75 && k==0)
    {
        q=150;
        k++;
    }
    if(q>=5 && q<=150)
        cacti2(q);
    if(p==25 || q==25)
    {
        score=0;
        dead_dino(0);
```



```
        loss(65,18);
setxy(0,39);
        return 0;
    }
    if(p==5)
    {
        p=150;
        score++;
    }
    if(q==5)
    {
        q=150;
        score++;
    }
    for(int c=1;c<=50000000;c++)
    {
    }
}
ch=getch();
if(ch==' ')
{
    for (x=0;p>=5,q>=5,x<=9;p=p-5,q=q-5,x+=3)
    {
        system("cls");
        for(int i=7;i<160;i++)
```

```
        {  
            setxy(i,35);  
            printf("-");  
        }  
        setxy(140,5);  
        yellow();  
        printf("SCORE : %d",score);  
        reset();  
        sun();  
        cloud(110,12);  
        cloud(15,4);  
        bird2(40,14);  
        bird2(130,6);  
        dino_run(x);  
        cacti1(p);  
        if(q>=5 && q<=150)  
            cacti2(q);  
        if(((p>=10 && p<=25) || (q>=10 && q<=25)) &&  
(x>=0 && x<3))  
        {  
            score=0;  
            dead_dino(0);  
            loss(65,18);  
            setxy(0,39);  
            return 0;  
        }
```

```
    }  
    if(p==5)  
    {  
        p=150;  
        score++;  
    }  
    if(q==5)  
    {  
        q=150;  
        score++;  
    }  
    for(int c=1;c<=50000000;c++)  
    {  
    }  
}  
for (x=9;q>=5,p>=5,x>=0;p=p-5,x-=3,q=q-5)  
{  
    system("cls");  
    for(int i=7;i<160;i++)  
    {  
        setxy(i,35);  
        printf("-");  
    }  
    setxy(140,5);  
    yellow();
```

```
printf("SCORE : %d",score);
reset();
sun();
cloud(110,12);
cloud(15,4);
bird2(40,14);
bird2(130,6);
dino_run(x);
cacti1(p);
if(q>=5 && q<=150)
cacti2(q);
if(((p>=10 && p<=25) || (q>=10 && q<=25)) &&
(x>=0 && x<3))
{
    score=0;
    dead_dino(0);
    loss(65,18);
    setxy(0,39);
    return 0;
}
if(p==5)
{
    p=150;
    score++;
}
```

```
        if(q==5)
        {
            q=150;
            score++;
        }
        for(int c=1;c<=50000000;c++)
        {
        }
    }

    }
    else if (ch=='x')
        return(0);
}

}

//colours
void red ()
{
    printf("\033[1;31m");
}
void yellow()
{
    printf("\033[1;33m");
}
```

```
void green()
{
    printf("\033[0;32m");
}
void reset ()
{
    printf("\033[0m");
}
void blue()
{
    printf("\033[0;34m");
}
void purple()
{
    printf("\033[0;35m");
}
void cyan()
{
    printf("\033[0;36m");
}
```

```
void dino_run(int x)
{
    yellow();
```

```
    setxy(10,30-x);
    printf("      o-o \n");
    yellow();
    setxy(10,31-x);
    printf("    /(\\_/)\\n");
    setxy(10,32-x);
    printf(",_.--` /'-` \\n");
    setxy(10,33-x);
    printf("'-. _ )/^\\n");
    setxy(10,34-x);
    printf("    ^\\n");
    reset();
}
```

```
void dead_dino(int x)
{
    red();
    setxy(10,30-x);
    printf("      x-x ");
    reset();
    yellow();
    setxy(10,31-x);
    printf("    /(\\_ ");
    setxy(10,32-x);
    printf(" ,____.--` /'- ");
}
```

```
    setxy(10,33-x);  
    printf("  '-. _ )/");  
    setxy(10,34-x);  
    printf("    //");  
    reset();  
}
```

```
void cacti1(int p)  
{  
    green();  
    setxy(p,29);  
    printf("  _ \n");  
    setxy(p,30);  
    printf(" _ ( ) \n");  
    setxy(p,31);  
    printf("( \\ / _ \n");  
    setxy(p,32);  
    printf(" \\, . / ) \n");  
    setxy(p,33);  
    printf(" / ^ \n");  
    setxy(p,34);  
    printf(" / \n");  
    reset();  
}
```



```
void cacti2(int p)
{
    green();
    setxy(p,29);
    printf(" _\n");
    setxy(p,30);
    printf(" ( ) _\n");
    setxy(p,31);
    printf(" _/ /\n");
    setxy(p,32);
    printf("( \|/\n");
    setxy(p,33);
    printf(" \|/\n");
    setxy(p,34);
    printf(" /\n");
    reset();
}
```

```
void loss(int x,int y)
{
    red();
    setxy(x,y);
    printf("88\n");
    setxy(x,y+1);
    printf("88\n");
}
```

```
    setxy(x,y+2);
    printf("88\n");
    setxy(x,y+3);
    printf("88 ,adPPYba, ,adPPYba, ,adPPYba,\n");
    setxy(x,y+4);
    printf("88 a8\"   \"8a I8[   \"\" I8[   \"\"\\n");
    setxy(x,y+5);
    printf("88 8b    d8 \"Y8ba, \"Y8ba, \n");
    setxy(x,y+6);
    printf("88 \"8a, ,a8\" aa  J8I aa  J8I\\n");
    setxy(x,y+7);
    printf("88 \"YbbdP\"' \"YbbdP\"' \"YbbdP\"'\n");
    reset();
}
```

```
void sun()
{
    yellow();
    setxy(70,4);
    printf("   ; : ;\n");
    setxy(70,5);
    printf(" .  \\_!,_/_ ,\n");
    setxy(70,6);
    printf("  `,'  `.\n");
    setxy(70,7);
```

```
printf("  /      \\n");
setxy(70,8);
printf("~ -- :      : -- ~\n");
setxy(70,9);
printf("  \\      ^n");
setxy(70,10);
printf("  ,^._ _.^.\n");
setxy(70,11);
printf("  ' / '!^ \\  `n");
setxy(70,12);
printf("    ; : ;\n");
reset();
}
```

```
void trex(int x,int y)
{
    green();
    setxy(x,y-1);
    printf(" ,d\n");
    setxy(x,y);
    printf(" 88\n");
    setxy(x,y+1);
    printf("MM88MMM 8b,dPPYba, ,adPPYba, 8b,    ,d8\n");
    setxy(x,y+2);
    printf(" 88  88P\'  \'Y8 a8P ____88 `Y8, ,8P\' \n");
}
```

```
    setxy(x,y+3);
    printf(" 88 88      8PP\"\\\"\\\"\\\"\\\"\\\" )888( \\n");
    setxy(x,y+4);
    printf(" 88, 88      \\\"8b, ,aa ,d8\\\" \\\"8b, \\n");
    setxy(x,y+5);
    printf(" \\\"Y888 88      \\\"Ybbd8\\\"\" 8P\\'  `Y8\\n");
    blue();
    setxy(x,y+7);
    printf("      You can jump with a space key!");
    setxy(x,y+8);
    yellow();
    printf("      Exit with 'x' key!");
    setxy(x,y+9);
    purple();
    printf("      Tap enter to start the Game.");
    reset();
}
```

```
void cloud(int x,int y)
{
    blue();
    setxy(x,y);
    printf("  _\\n");
    setxy(x,y+1);
    printf(" ,-' '-,\\n");
}
```

```
    setxy(x,y+2);  
    printf(" (_____)\\n");  
    reset();  
  
}
```

```
void bird1(int x,int y)  
{  
    red();  
    setxy(x,y);  
    printf(" ^\\");  
    yellow();  
    setxy(x,y+1);  
    printf("<0(^)K->");  
    red();  
    setxy(x,y+2);  
    printf(" \\V");  
    reset();  
}
```

```
void bird2(int x,int y)  
{  
    red();  
    setxy(x,y);  
    printf(" /");  
}
```

```
    yellow();  
    setxy(x,y+1);  
    printf("<0(^)K->");  
    red();  
    setxy(x,y+2);  
    printf("  ");  
    reset();  
}
```

3. TESTING

3.1CODE ERRORS AND DEBUGGING

1. Misalignment of letters:



>>Debugged



2 Color Leakage:

```
TIC TAC TOE

RULES:ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS : ) !!!

Player 1 (X) - Player 2 (O)

  1 | 2 | 3
  --+--+--
  4 | 5 | 6
  --+--+--
  7 | 8 | 9

PLAYER 1 -> The number you choose is : Invalid move ^C
C:\Users\DELL\Desktop\E-Game Arena\1>
```

>>Debugged

```
TIC TAC TOE

RULES:ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS : ) !!!

Player 1 (X) - Player 2 (O)


  1 | 2 | 3
  --+--+--
  4 | 5 | 6
  --+--+--
  7 | 8 | 9

PLAYER 1 -> The number you choose is : Invalid move ^C
C:\Users\DELL\Desktop\E-Game Arena\1>
```


4. RESULT AND ANALYSIS

4.1CODE OUTPUT

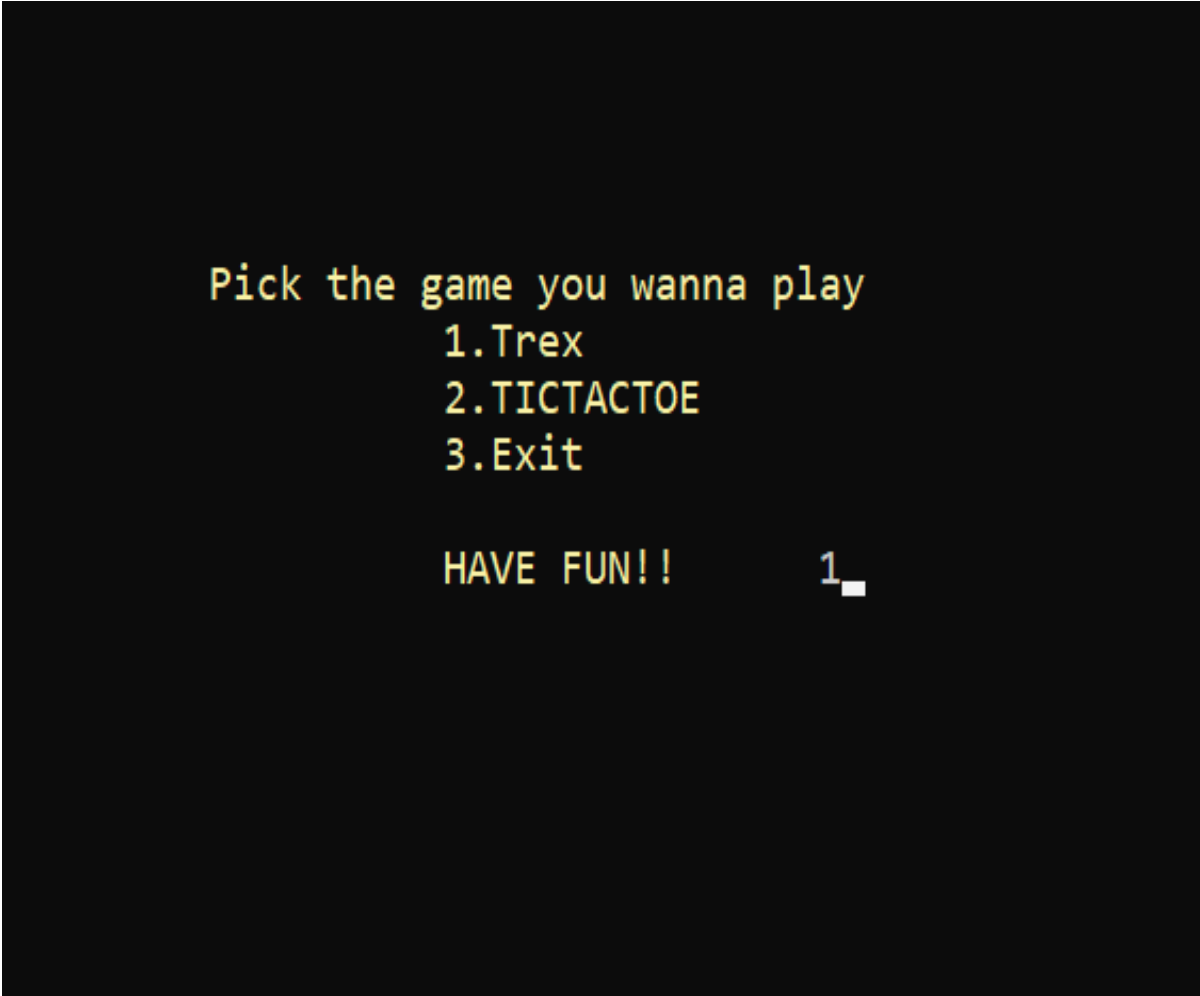
USERS CHOICE ENTRY:

A screenshot of a terminal window with a black background and yellow text. The text displays a menu for game selection. It starts with the prompt 'Pick the game you wanna play', followed by a numbered list: '1.Trex', '2.TICTACTOE', and '3.Exit'. Below the list, it says 'HAVE FUN!!' followed by a small horizontal line.

```
Pick the game you wanna play
1.Trex
2.TICTACTOE
3.Exit

HAVE FUN!!  _
```

WHEN USER ENTERS 1:



Pick the game you wanna play

1.Trex

2.TICTACTOE

3.Exit

HAVE FUN!! 1_

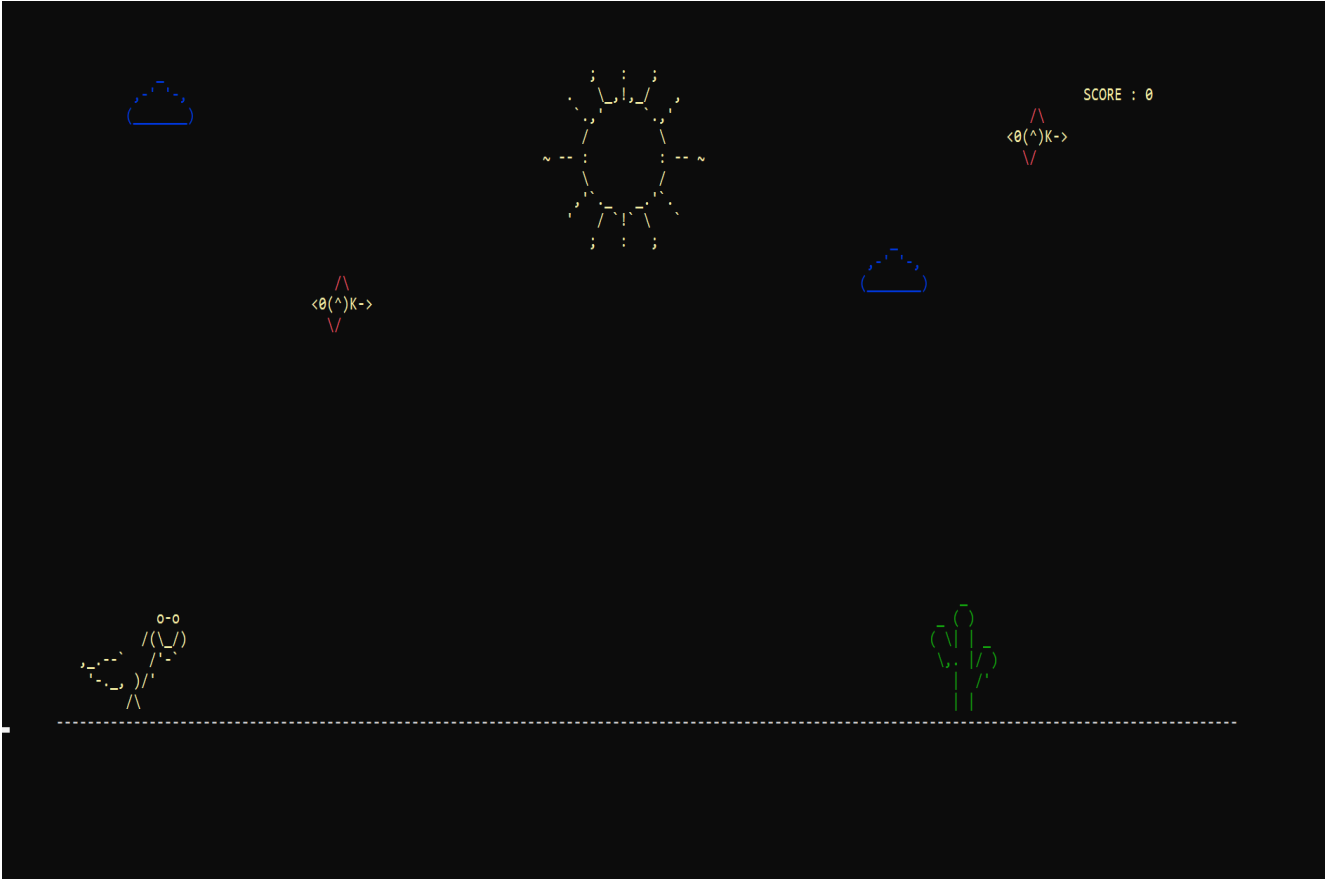
THE USER ENTERS INTO THE 1ST GAME: DINO GAME

```
      ,d
      88
MM88MMM 8b,dPPYba,  ,adPPYba, 8b,      ,d8
88      88P'   "Y8 a8P _____88 `Y8, ,8P'
88      88      8PP" "" "" "" "" ""      )888(
88,     88      "8b,      ,aa ,d8" "8b,
"Y888 88      `Y8bd8"' 8P'      `Y8
```

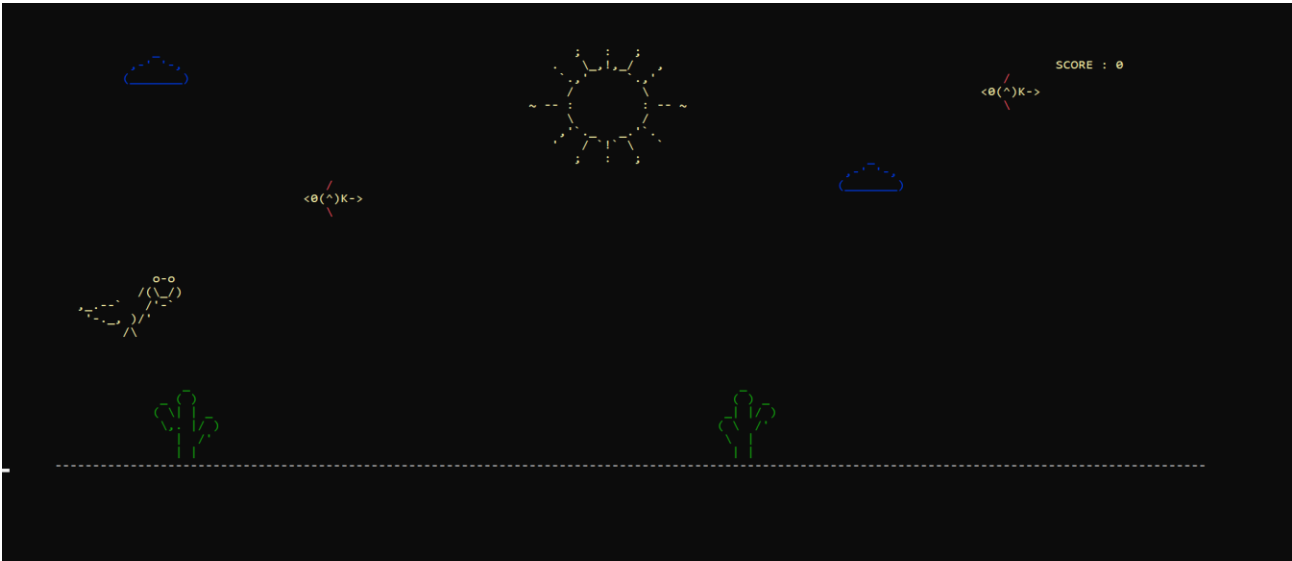
You can jump with a space key!
Exit with 'x' key!
Tap enter to start the Game.

THE DINO GAME:

>When the user enters one it enters the dino-game



>dino jumps when you hit the space bar.



>The game terminates when dino hits the obstacle.



USER ENTERS 2:

Pick the game you wanna play

1.Trex

2.TICTACTOE

3.Exit

HAVE FUN!! 2_

TIC-TAC-TOE GAME:

>It enter the 2nd game

```

MMMM88MMMM MM ,adPPYba,  MMM88MMMM ,adPba, ,adPPYba,  MMM88MMMM ,abdPPYba,  MMM88MMMM
88 88 a8P 88 a8 8a a8P 88 a8P P8a 88
88 88 8PP 88 88 88 8PP 88 8PP PP8 88
88 88 8PP 88 88MM8MM88 8PP 88 8PP PP8 88MM88
88 88 8PP 88 88 88 8PP 88 8PP PP8 88
88 88 "8b, 88 88 "8b, 88 "8b, ,b8" 88
88 MM `Ybbd8" ' 88 MM MM `Ybbd8" ' 88 `8Ybbd8" ' 88MM88MMMM

WIN THE GAME WITH YOUR WIT
press enter to start the game

```

```

TIC TAC TOE

RULES:ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS : ) !!!

Player 1 (X) - Player 2 (O)

  1 | 2 | 3
  --|---|
  4 | 5 | 6
  --|---|
  7 | 8 | 9

PLAYER 1 -> The number you choose is :

```

PLAYER 1 WINS:

```
TIC TAC TOE

RULES:ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS : ) !!!

Player 1 (X) - Player 2 (O)

  X | O | X
  ---|---|---
  O | X | O
  ---|---|---
  7 | 8 | X

Player 1 win!!!
```

PLAYER 2 WINS:

```
TIC TAC TOE

RULES:ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS : ) !!!

Player 1 (X) - Player 2 (O)

  O | X | X
  ---|---|---
  4 | O | X
  ---|---|---
  7 | 8 | O

Player 2 win!!! _
```


GAME DRAW:

TIC TAC TOE

RULES: ENTER 1 NO. AT A TIME AND WIN THE GAME USING YOUR WITS :) !!!

Player 1 (X) - Player 2 (O)

X	O	X
O	O	X
X	X	O

'GAME DRAW' : -)

USER ENTERS 3: EXIT

>When the user enters 3 it terminates from the program.

```
Pick the game you wanna play
1.Trex
2.TICTACTOE
3.Exit

HAVE FUN!!    3

C:\Users\DELL\Desktop\E-Game Arena\1>
```

5. CONCLUSION AND FUTURE ENHANCEMENT

The project has thus made use of some of the most basic functions and methods available in the C language. The game also showcases the simplicity with which games can be developed in C by effectively reducing the number of lines required to program a game. These games also emphasize the importance of using various modules to bring out the functionality using the code. In conclusion, this game-E-GAME ARAENA tries to convey to the user the capabilities of C's simple and robust usage.

Future enhancements include:

1. TIC-TAC-TOE:

- AI implementation to the game
- Interface can be done more user-friendly
- Game appearance can be done more presentable and attractive

2. DINO GAME:

- It can be loaded with 3 different levels:
 - > Easy
 - > Medium
 - > Hard
- Game appearance can be done more presentable and attractive
- Different varieties of obstacles can be implemented

6. REFERENCES

- ✓ <https://github.com/farbodahm/chrome-dinosaur>
- ✓ <https://cprogrampracticals.blogspot.com/2016/10/c-program-to-print-text-at-given.html?m=1>
- ✓ <http://www.cprogrammingnotes.com/question/tic-tac-toe-game.html>
- ✓ <https://code-projects.org/dino-game-c-programming-source-code/>
- ✓ C programming language (ANSI C) by Kernighan, Brian W Ritchie, Dennis M