**PROJECT REPORT: TIC TAC TOE**

**1.OVERVIEW:**

The project name is TIC TAC TOE.This game is very popular and fairly simple by itself.It is actually a two player game.In this game,there is a board of n x n squares.In this game 4 x 4 grid is used.The goal of the TIC TAC TOE game is to be one of the palyers to get the same three symbols horizontally,vertically or diagonally on a 4 x 4 grid.The game can be played in 4 x 4 squares.The game is played by two palyers(Player 1 & Player 2).In this project C language is used which is a general prpose programming language and can efficiently work on enterprise applications,games,graphics,applications requiring calculations etc.OpenGL(Open Graphics Library) is also used in this project which is a cross-platform application programming interface(API) for rendering 2D and 3D vector graphics.

**2.HOW THE PROJECT IS DIFFERENT FROM OTHER SIMILAR PROJECT:**

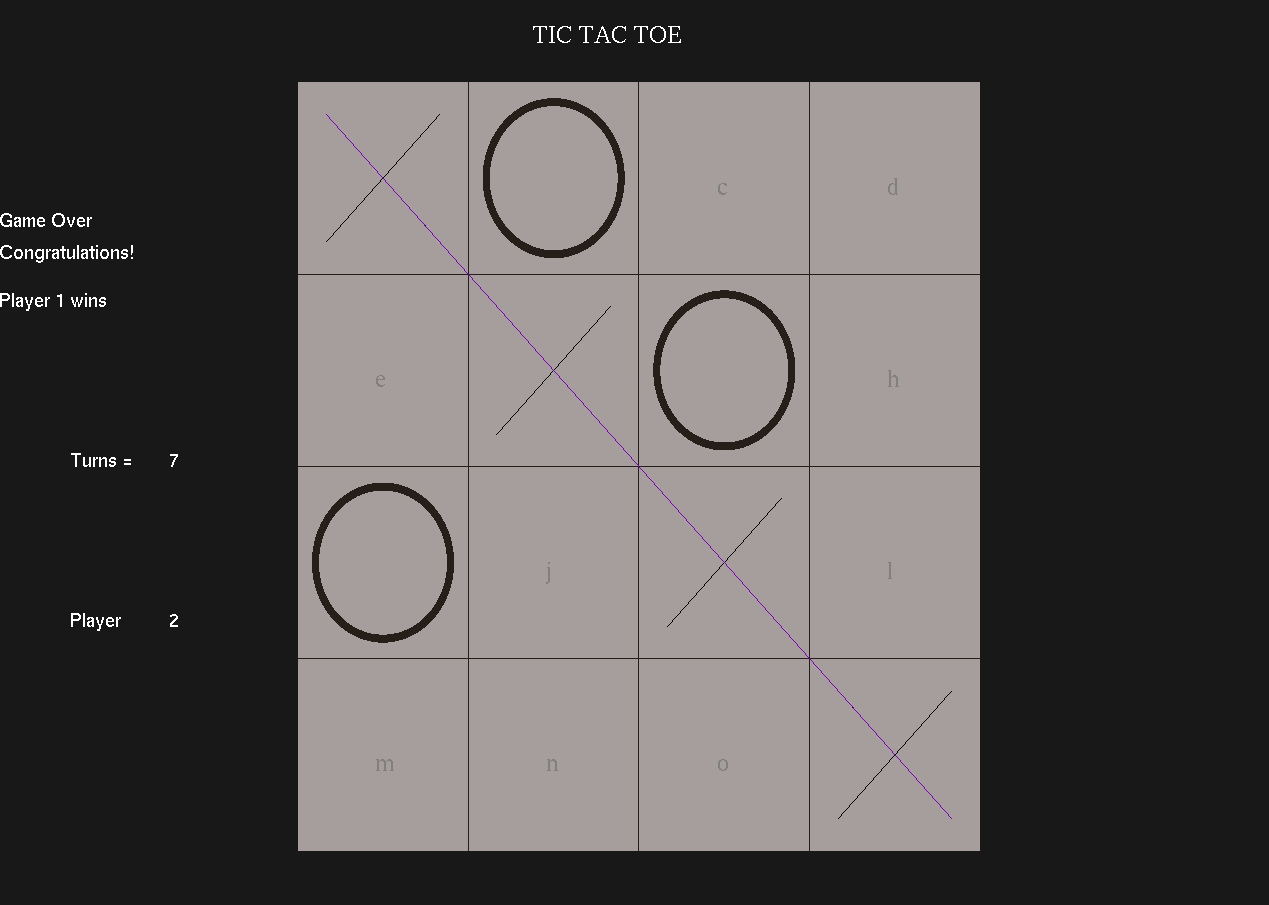
The project is different from other similar projects because of the features added to make the project : **(i).**Adding Graphics using OpenGL.

**(ii).**Adding Music.

**(iii).**File Work(Used while combining the 4 games ,since it was a project of 4 members each to make an individual game and to combine the game so that user can choose which game to play form main menu).

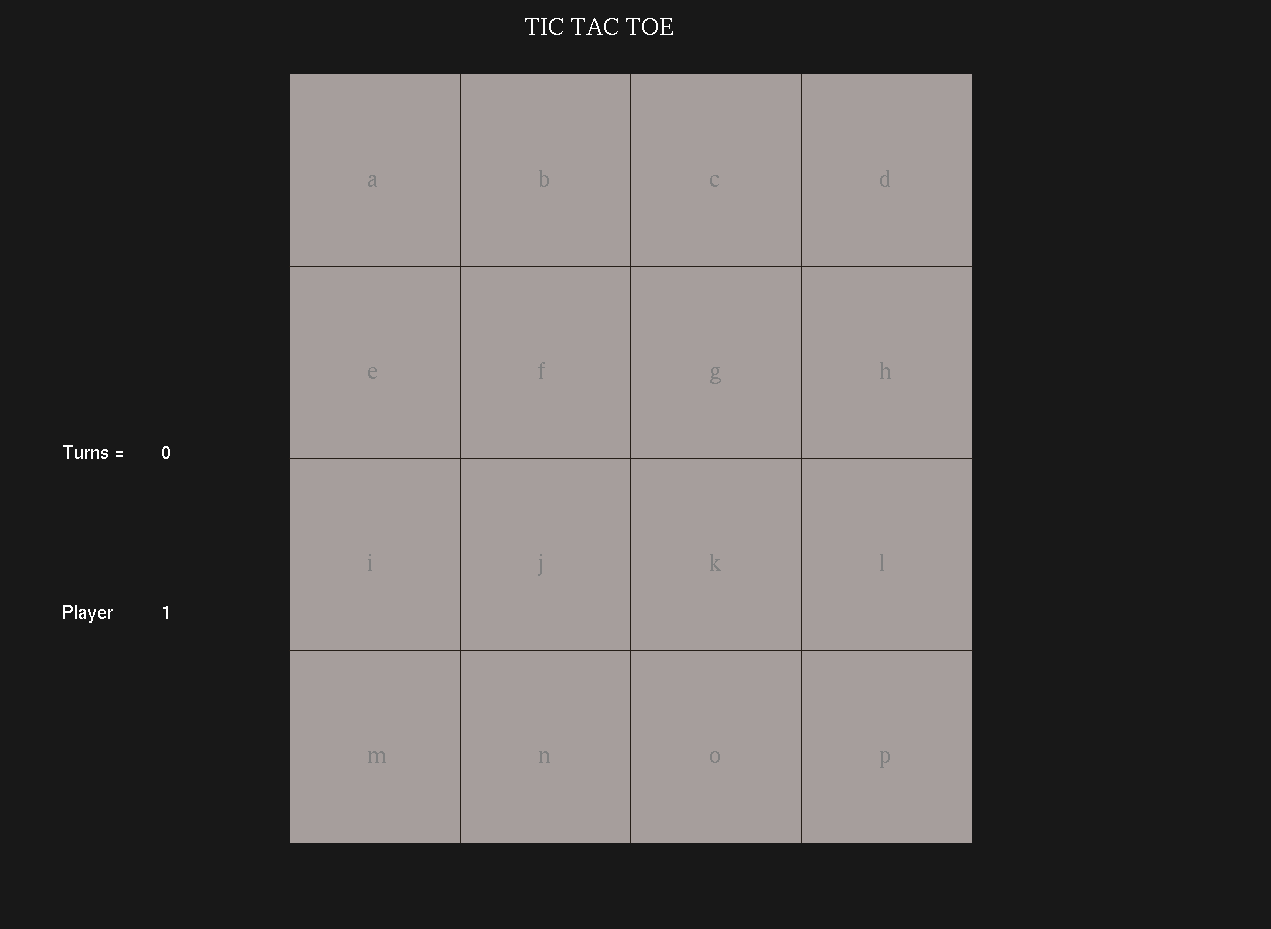
**3.FEATURES OF THE PROJECT:**

In this project,the game is played by 2 players,player 1 uses X and player 2 uses O.It has a 4X4 grid which means there are 16 small squares in total inside the grid.So the number of turns is set to zero initially,each time the player gives input the number of turns increases and when it becomes 16 the game is over.Each players intention in this game is to match the same symbols horizontally,vertically,diagonally. Whoever matches the symbol horizontally or vertically or diagonally wins the game.A background music is also added which starts when the game starts and plays until the game is over or until any of the palyer wins the game.



**4.INTERACTION OF COMPONENTS WITH EACH OTHER:**

The components interact with each other through different functions used in the project.A 2D array or matrix is used to access the 4 x4 grid and the grid is made using the OpenGL functions.The functions used to draw the grid are DrawGameArea( ),Drawbox( ),Drawlines( ).pitnuminarray( ) function is used to change the turn of the player.Then void display( ) function is used to to display the output on the console.In this function (a to q) 16 alphabets are initialized in the matrix(as shown below) for playing the game using keyboard.At last to finish the game bool gameover is used in this function.



**5.CODE FOR IMPORTANT COMPONENTS:**

The code for important components are given below:

**(i).Changing player and no. of turns each time after the input(X or O):**

Here state is matrix of 4 by 4.When player 1 gives the input (X) then it shifts to player 2.

void putnuminarray(int x){

if(x == 1) if(state[0][0] == 0) {state[0][0] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 2) if(state[0][1] == 0) {state[0][1] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 3) if(state[0][2] == 0) {state[0][2] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 4) if(state[0][3] == 0) {state[0][3] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 5) if(state[1][0] == 0) {state[1][0] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 6) if(state[1][1] == 0) {state[1][1] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 7) if(state[1][2] == 0) {state[1][2] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 8) if(state[1][3] == 0) {state[1][3] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 9) if(state[2][0] == 0) {state[2][0] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 10) if(state[2][1] == 0) {state[2][1] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 11) if(state[2][2] == 0) {state[2][2] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 12) if(state[2][3] == 0) {state[2][3] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 13) if(state[3][0] == 0) {state[3][0] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 14) if(state[3][1] == 0) {state[3][1] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 15) if(state[3][2] == 0) {state[3][2] = player; (player == 1) ? player = 2 : player = 1; turns++;}

if(x == 16) if(state[3][3] == 0) {state[3][3] = player; (player == 1) ? player = 2 : player = 1; turns++;}}

**(ii).CODE FOR FIINDING HORIZONTAL MATCH:**

for(i = 0; i<4; i++){

if(state[i][0] == state[i][1] && state[i][2] == state[i][1] && state[i][0] == state[i][3] && state[i][2] == state[i][3] && state[i][0] != 0) {

glColor3f(0.5, 0, 0.75);

Drawlines(-220, 180-(120\*i) ,220, 180-(120\*i));

gameover = true;

break;

}

}

void Drawlines(int x1, int y1, int x2, int y2){

glBegin(GL\_LINE\_LOOP);

glVertex2f(x1, y1);

glVertex2f(x2, y2);

glEnd();

}

**(iii).CODE FOR FINDING VERTICAL MATCH:**

for(j = 0; j<4;j++){

if(state[0][j] == state[1][j] && state[2][j] == state[1][j] && state[0][j] == state[3][j] && state[2][j] == state[3][j] && state[0][j] != 0) {

gameover = true;

glColor3f(0.5, 0, 0.75);

Drawlines(-180+(120\*j), -220 ,-180+(120\*j), 220);

break;

}

}

void Drawlines(int x1, int y1, int x2, int y2){

glBegin(GL\_LINE\_LOOP);

glVertex2f(x1, y1);

glVertex2f(x2, y2);

glEnd();

}

**(iv).CODE FOR FINDING DIAGONAL MATCH:**

if(state[0][0] == state[1][1] && state[2][2] == state[1][1] && state[2][2] == state[3][3] && state[0][0] == state[3][3] &&state[0][0] != 0) {

glColor3f(0.5, 0, 0.75);

Drawlines(220, -220 ,-220, 220);

gameover = true;

k = 1;

}

else if(state[0][3] == state[1][2] && state[2][1] == state[1][2] && state[2][1] == state[3][0] && state[0][3] == state[3][0] &&state[0][3] != 0) {

glColor3f(0.5, 0, 0.75);

Drawlines(-220, -220 ,220, 220);

gameover = true;

k = 2;

}

**(v).CODE FOR DRAWING X AND O:**

for(int p = 0; p <4; p++){

for(int q = 0; q < 4; q++){

if(state[p][q] == 1) PrintinBox(p,q);

else if(state[p][q] == 2){

glColor3f(0.15, 0.12, 0.10);

drawCircle(50, 50, 0, -180+(120\*q), 180-(120\*p));

glColor3f(0.65, 0.62, 0.610);

drawCircle(45, 45, 0, -180+(120\*q), 180-(120\*p));

}

}

}

void PrintinBox(int i, int j)

{

glColor3f(0.65, 0.62, 0.610);

DrawBox(-220+(j\*120),-140+(j\*120),220-(i\*120),140-(i\*120));

glColor3f(0.0, 0.0, 0.0);

Drawlines(-220+(j\*120),220-(i\*120), -140+(j\*120), 140-(i\*120));

Drawlines(-220+(j\*120),140-(i\*120), -140+(j\*120), 220-(i\*120));

}

void drawCircle(float rx, float ry, int half, int xp, int yp){

glPushMatrix();

glPointSize(3);

glBegin(GL\_POLYGON);

int i;

int len = 360;

if(half == 1) len = 180;

for(i=0; i<len; i++){

float theta = (i\*3.1416)/180.0;

glVertex2f(rx\*cos(theta) + xp, ry\*sin(theta) + yp);

}

glEnd();

glPopMatrix();}

**(vi).CODE FOR GAMEOVER:**

if(turns==16 && !gameover){

glColor3f(1,1,1);

drawString(GLUT\_BITMAP\_HELVETICA\_18,"ITS A DRAW!",-420,150);

}

glColor3f(1,1,1);

if(gameover==true){

drawString(GLUT\_BITMAP\_HELVETICA\_18,"Game Over",-450,150);

if(player == 1){

drawString(GLUT\_BITMAP\_HELVETICA\_18,"Congratulations!",-450,130);

drawString(GLUT\_BITMAP\_HELVETICA\_18,"Player 2 wins",-450,100);

}

if(player ==2){

drawString(GLUT\_BITMAP\_HELVETICA\_18,"Congratulations!",-450,130);

drawString(GLUT\_BITMAP\_HELVETICA\_18,"Player 1 wins",-450,100);

}

}

**6.DIFFICULTY FACED WHILE IMPLEMENTING THE CODE:**

There are a few number of difficulties that arised while implanting the codes. The difficulties faced are:

**(i).**Firstly I faced difficulty while drawing the circle i.e symbol (O) for player 2.Then I fixed it seeing some youtube videos and by the help of google.

**(ii).**Secondly, difficulty arised while setting the font size. In OpenGL only two types of font size can be used which is a limitation of GLUT.

**(iii).**Thirdly, while adding the music different music can’t be added all at a time.

**7.RESOURCES TAKEN HELP FROM:**

**(i).** <https://stackoverflow.com/>

**(ii).** [www.youtube.com](http://www.youtube.com)