

**topic.png**tic-TAC-TOE

With intelligence

And file handling.

**Presented to you by:-**

**Nivedit jain**

**Submitted to:-**

**TaranJeet Singh Sir**

**:-**

**This is to certify that Master Nivedit Jain (of class 12th A of Nirmal Ashram Deepmala Pagarani Public School, Syampur Rishikesh )**

**have made the project on his own with his group. His project titles:-**

**Tic-Tac-Toe With Intelligence And File Handling.**

**Signatures:-**

**Mrs.** [**Lalitha Krishnaswamy**](https://www.facebook.com/lalitha.krishnaswamy.1?hc_ref=ARTCrID9ED_8L08uiTiymtTfLEKKkGHwdZx06oQLh2aNkaIVN5oporGNQ52pTGZKPZE)

**(Principal)**

**Mr.TaranJeet Singh**

**(Computer Teacher)**

Since, childhood we all are very fascinated by games specially computer games and class time passes like Tic-Tac-Toe we decided to combine both, inspite of all other Tic-Tac-Toe programmes available on this level we had tried to make computer think whereas most of available on net provide you with random moves, we have also connected the files so that you can view previos score and make improvements.

Tic –Tac-Toe is very well known with the hope that all rules are known but still defining them again:-

* Each player plays his symbol on the board.
* If he/she could make his/her symbol completely in a row,coloumn or diagonal he/she wins the game.
* If no one is able to do so then game is considered to be a draw.

I am quite thankful to all my group members Tejasva Jugran, Dev Dang, and Aditya Satija for their support.

Saksham Jain for the inspiration of the project .

TaranJeet Sir for his support on every step.

**Source Code**

//files shown in drive may contain symbols but they work just save files and cpp file in your c++ bin and plzz do not forget to comment your score.

//project file thanks to all the contributors mentioned.

//thanks to my teacher TaranJeetSingh Sir and group members DevDang , TejasvaJugran , and SakshamJain

//plzz tell new ideas and improvements in comments or email at NiveditJain

//idea and debugged SAKSHAM JAIN

#include <iostream.h>

#include<dos.h>

#include<conio.h>

#include<ctype.h>

#include<stdlib.h>

#include<fstream.h>

#include<stdio.h>

#include<string.h>

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

    int checkwin();                 //nivedit

    int checkwin2();                  //tejasva

    void board(char\*,int);                       //nivedit

    void board2();                 //nivedit

    char check\_comp\_X();     //nivedit

    int convert(char);         //nivedit

    char check\_comp\_O();    //nivedit

    int intelligent();       //nivedit

    int intelligent2();                 //dev

    char finale();           //dev

    int player1(char\*);         //nivedit

    int player2();         //nivedit

    int dev();                 //dev

    void demo();            //tejasva

    void career();           //tejasva

    int gamestart(char\*);        //tejasva

    int mainT();                      //tejasva

    void player0(int);                          //tejasva

    void restore();                        //nivedit

    int entry();                   //nivedit

    void drawer();                     //nivedit

    void replay(char\*);                       //nivedit

    struct data

    {

    char a[100];

    int win;

    int lose;

    int draw;

    };

    void restore()

    {

int i;

for(i=1;i<=9;i++)

{

    square[i]=(char)(i+48);

}

    }

    void drawer()

    {

      char \*ch;

      cout<<endl<<"to end press E(e) or else press any key to exit";

      gets(ch);

      if(strlen(ch)==1)

      {

      if(ch[0]=='e'|| ch[0]=='E')

      {

      return;

      }

      else

      {

      mainT();

      }

      }

      else

      {

      mainT();

      }

    }

    void main()

    {

    mainT();

    }

    int mainT()

    {char a,\*b;

int static l=0;

data temp;

label:

clrscr();

if(l==0)

{

demo();

cout<<"\n\t\t Heartful Thanks to all"<<endl;

cout<<"press any key to continue....";

getch();

l++;

}

clrscr();

restore();

cout<<endl;

temp.win=0;

temp.lose=0;

temp.draw=0;

cout<<" \t\t\t #TIC TAC TOE WITH ~@tnda~\n";

cout<<"\t\t\t LETS SEE WHO WINS !!!! \n";

cout<<"PREVIOUS RECORDS(p)-->(P)""\t\t\t\t"<<"(N)<--NEW GAME(n)"<<"\n";

cout<<"\t\t\t EXIT(E) \n";

cin>>a;

if(a=='P'||a=='p')

{ career();

}

else

{if(a=='N'||a=='n')

    {   int l,check=0;

cout<<"Enter your name: \n";

gets(temp.a);

data t2;

ifstream f2;

f2.open("player.TXT",ios::binary);

if(f2)

{

while(!f2.eof())

{

f2.read((char\*)&t2,sizeof(t2));

if(!strcmp(t2.a,temp.a))

{

temp.win=t2.win;

temp.lose=t2.lose;

temp.draw=t2.draw;

check++;

break;

}

}

f2.close();

}

else

{

cout<<"some file problem";

}

l=gamestart(temp.a);

ofstream f1;

if(l!=3)

{

f1.open("player.TXT",ios::binary|ios::app|ios::out);

switch(l)

{

 case 0:

 temp.draw++;

 f1.write((char\*)&temp,sizeof(temp));

 cout<<endl<<"game draw";

 break;

 case 1:

 temp.win++;

 f1.write((char\*)&temp,sizeof(temp));

 cout<<endl<<"==>"<<temp.a<<" wins";

 break;

 case 2:

 temp.lose++;

 f1.write((char\*)&temp,sizeof(temp));

 cout<<endl<<"==> computer wins";

 break;

}

f1.close();

 if(check!=0)

 {

 data tic;

 int nivi=0;

ofstream ftemp;

ftemp.open("temp.TXT",ios::binary);

ifstream f3;

f3.open("player.TXT",ios::binary);

while(f3.read((char\*)&tic,sizeof(data)))

{

 if(nivi==0 && strcmp(tic.a,temp.a)==0)

 {

 nivi++;

 }

 else

 {

 ftemp.write((char\*)&tic,sizeof(tic));

 }

}

     ftemp.close();

     f3.close();

     ifstream ftemp1;

     ofstream f4;

     ftemp1.open("temp.TXT",ios::binary);

     f4.open("player.TXT",ios::binary);

     data p;

     while(ftemp1.read((char\*)&p,sizeof(data)))

{

 f4.write((char\*)&p,sizeof(tic));

}

     ftemp1.close();

     f4.close();

     }

     cout<<endl<<"to replay your game press R(r) else press any key";

     char \*ch;

   gets(ch);

  if(strlen(ch)==1)

  {

  if(ch[0]=='r'|| ch[0]=='R')

   {

   replay(temp.a);

  }

  }  }

     drawer();

     }

    else

    {

if(a=='e'||a=='E')

{

    cout<<"\*\*\*\*\*\*\*\*\* BYE BYE BYE \*\*\*\*\*\*\*\*\*";

}

else

{

    cout<<"Invalid entry";

    goto label;

    }}}

getch();

return 0;

    }

void career()

{       int t=0;

data k;

int n=0,count=0;

ifstream f2;

f2.open("player.TXT",ios::binary,ios::in);

while(f2.read((char\*)&k,sizeof(data)))

{

n++;

}

f2.close();

data \*sorted;

sorted =new data[n];

ifstream f1;

char ch;

f1.open("player.TXT",ios::binary,ios::in);

if(!f1)

{

cout<<"Enable to  Open";

}

clrscr();

while(f1.read((char\*)&k,sizeof(data)))

{

sorted[count]=k;

count++;

}

f1.close();

int i,j;

data tempo;

for(i=1;i<n;i++)

{

 for(j=0;j<n-i;j++)

 {

 if(sorted[j].win<sorted[j+1].win)

 {

 tempo=sorted[j];

 sorted[j]=sorted[j+1];

 sorted[j+1]=tempo;

 }

 }

}

for(i=0;i<n;i++)

{

k=sorted[i];

cout<<"PLayer Name: "<<k.a<<endl<<"Total  Winnings:"<<k.win<<endl<<"Total Game Loss:"<<k.lose<<endl<<"Total Game Draw:"<<k.draw<<endl;

cout<<endl<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

t++;

if(t%4==0)

{

getch();

}

}

drawer();

getch();

    }

    int gamestart(char\* name)

    {

int t;

cout<<"\n Person"<<"\t Starting your game \n";

cout<<"AGAINST COMPUTER-->press 1 \t\t\t  AGAINST 2player-->press 2 \n";

int i;

cin>>i;

if(i==1)

{

    int k;

    cout<<"You are Player1 \n";

    cout<<"Press any key to continue";

    getch();

    k=player1(name);

    return(k);

}

else

{

    cout<<"Human vs Human started press any key to continue";

    cout<<endl<<"human vs human records are not noted in progress meter";

    getch();

    player2();

    return (3);

}

    }

    void demo()

    {

randomize();

int i=random(9)+1;

player0(i);

    }

    void player0(int p)

    {

int t=0;

int player=1,i,choice;

char mark;

do

{

    board2();

    player=(player%2)?1:2;

    if(player==2)

    {

if(t==0)

{

    t++;

    choice=p;

}

else

{

    t++;

    choice =intelligent();

}

    }

    else

    {

choice=intelligent2();

    } mark=(player == 1) ? 'O' : 'X';

    if (choice == 1 && square[1] == '1')

square[1] = mark;

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

square[2] = mark;

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

square[3] = mark;

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

square[4] = mark;

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

square[5] = mark;

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

square[6] = mark;

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

square[7] = mark;

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

square[8] = mark;

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

square[9] = mark;

    else

    {

player--;

    }

    i=checkwin2();

    player++;

}while(i==-1);

board2();

    }

    int player1(char\* name)

    {

int t=0;int l;

int player=1,i,choice;

char mark,\*temp;

ofstream f1;

f1.open("game.TXT",ios::binary);

do

{

    board(name,1);

    player=(player%2)?1:2;

    cout << "Player " << player << ", enter a number:  ";

    if(player==1)

    {

saksham:

gets(temp);

if(strlen(temp)!=1)

{

cout<<"invalid entry";

goto saksham;

}

if(!(isdigit(temp[0])))

{

cout<<endl<<"invalid entry";

goto saksham;

}

choice=convert(temp[0]);

if(choice<1 || choice>9)

{

cout<<endl<<"invalid entry";

goto saksham;

}

    }

    else

    {

if(t==0)

{

    t++;

    choice=dev();

}

else

{

    t++;

    choice =intelligent();

}

    }

    mark=(player == 1) ? 'O' : 'X';

    if (choice == 1 && square[1] == '1')

square[1] = mark;

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

square[2] = mark;

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

square[3] = mark;

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

square[4] = mark;

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

square[5] = mark;

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

square[6] = mark;

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

square[7] = mark;

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

square[8] = mark;

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

square[9] = mark;

    else

    {

cout<<"invalid entry";

goto saksham;

    }

    f1.write((char\*)&choice,sizeof(choice));

    i=checkwin();

    player++;

}while(i==-1);

board(name,1);

if(i==1)

    {

    player--;

    return(player);

    }

else

    {

    return(0);

    }

    }

    int player2()

    {

int player = 1,i,choice;

char mark,\*temp;

do

{

    board("player 1",0);

    player=(player%2)?1:2;

    cout << "Player " << player << ", enter a number:  ";

saksham:

gets(temp);

if(strlen(temp)!=1)

{

cout<<"invalid entry";

goto saksham;

}

if(!(isdigit(temp[0])))

{

cout<<endl<<"invalid entry";

goto saksham;

}

choice=convert(temp[0]);

if(choice<1 || choice>9)

{

cout<<endl<<"invalid entry";

goto saksham;

}

    mark=(player == 1) ? 'O' : 'X';

    if (choice == 1 && square[1] == '1')

square[1] = mark;

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

square[2] = mark;

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

square[3] = mark;

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

square[4] = mark;

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

square[5] = mark;

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

square[6] = mark;

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

square[7] = mark;

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

square[8] = mark;

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

square[9] = mark;

    else

    {

cout<<"invlaid move";

goto saksham;

    }

    i=checkwin();

    player++;

}while(i==-1);

board("player 1",0);

player--;

if(i==1 && player==1)

{

    cout<<"==>\aPlayer "<<player<<" win ";

return 1;

}

else

    {

    if(i==1 && player==2)

    {

     cout<<"==>\aPlayer "<<player<<" win ";

     return 2;

    }

    cout<<"==>\aGame draw";

return 0;

}

    }

    /\*

    1 FOR GAME IS OVER WITH RESULT

    -1 FOR GAME IS IN PROGRESS

    O GAME IS OVER AND NO RESULT

     \*/

    int checkwin()

    {

if (square[1] == square[2] && square[2] == square[3])

    return 1;

else if (square[4] == square[5] && square[5] == square[6])

    return 1;

else if (square[7] == square[8] && square[8] == square[9])

    return 1;

else if (square[1] == square[4] && square[4] == square[7])

    return 1;

else if (square[2] == square[5] && square[5] == square[8])

    return 1;

else if (square[3] == square[6] && square[6] == square[9])

    return 1;

else if (square[1] == square[5] && square[5] == square[9])

    return 1;

else if (square[3] == square[5] && square[5] == square[7])

    return 1;

else if (square[1] != '1' && square[2] != '2' && square[3] != '3'

&& square[4] != '4' && square[5] != '5' && square[6] != '6'

&& square[7] != '7' && square[8] != '8' && square[9] != '9')

    return 0;

else

    return -1;

    }

    void board(char \*name,int i)

    {

clrscr();

cout << "\n\n\tTic Tac Toe\n\n";

if(i==1)

{

cout <<name<<" (O)  -  Computer (X)" << endl << endl;

}

else

{

cout <<"Player 1 (O)  -  Player 2 (X)" << endl << endl;

}

cout << endl;

cout << "     |     |     " << endl;

cout << "  " << square[1] << "  |  " << square[2] << "  |  " << square[3] << endl;

cout << "\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" << endl;

cout << "     |     |     " << endl;

cout << "  " << square[4] << "  |  " << square[5] << "  |  " << square[6] << endl;

cout << "\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" << endl;

cout << "     |     |     " << endl;

cout << "  " << square[7] << "  |  " << square[8] << "  |  " << square[9] << endl;

cout << "     |     |     " << endl << endl;

    }

    char check\_comp\_X()

    {

int t,j,k,i;

t=0;

char nivi[3][3];

for(i=0;i<3;i++)

{

    for(j=0;j<3;j++)

    {

t++;

nivi[i][j]=square[t];

    }

}

t=0;

//i

for(i=0;i<3;i++)

{

    t=0;

    for(k=0;k<3;k++)

    {

if(nivi[i][k]=='X')

{ t++;

}

    }

    if(t==2)

    {

for(k=0;k<3;k++)

{

    if(isdigit(nivi[i][k])!=0)

    {

return(nivi[i][k]);

    }

}

    }

}

//j

for(j=0;j<3;j++)

{

    t=0;

    for(k=0;k<3;k++)

    {

if(nivi[k][j]=='X')

{ t++;

}

    }

    if(t==2)

    {

for(k=0;k<3;k++)

{

    if(isdigit(nivi[k][j])!=0)

    {

return(nivi[k][j]);

    }

}

    }

}

//principal diagnal

t=0;

for(i=0;i<3;i++)

{

    if(nivi[i][i] =='X')

    {

t++;

    }

}

if(t==2)

{

    for(i=0;i<3;i++)

    {

if(isdigit(nivi[i][i])!=0)

{

    return(nivi[i][i]);

}

    }

}

//2ndry diagnal

t=0;

for(i=0;i<3;i++)

{

    if(nivi[i][2-i] =='X')

    {

t++;

    }

}

if(t==2)

{

    for(i=0;i<3;i++)

    {

if(isdigit(nivi[i][2-i])!=0)

{

    return(nivi[i][2-i]);

}

    }

}

return(0);

    }

    int convert(char a)

    {

if(a==NULL)

{

    return((int)0);

}

else

{

    int x=(int)a - 48;

    return(x);

}

    }  //confusion problem with null ask sir

    char check\_comp\_O()

    {

int t,i,j,k;

t=0;

char nivi[3][3];

for(i=0;i<3;i++)

{

    for(j=0;j<3;j++)

    {

t++;

nivi[i][j]=square[t];

    }

}

t=0;

//i

for(i=0;i<3;i++)

{

    t=0;

    for(k=0;k<3;k++)

    {

if(nivi[i][k]=='O')

{ t++;

}

    }

    if(t==2)

    {

for(k=0;k<3;k++)

{

    if(isdigit(nivi[i][k])!=0)

    {

return(nivi[i][k]);

    }

}

    }

}

//j

for(j=0;j<3;j++)

{

    t=0;

    for(k=0;k<3;k++)

    {

if(nivi[k][j]=='O')

{ t++;

}

    }

    if(t==2)

    {

for(k=0;k<3;k++)

{

    if(isdigit(nivi[k][j])!=0)

    {

return(nivi[k][j]);

    }

}

    }

}

//principal diagnal

t=0;

for(i=0;i<3;i++)

{

    if(nivi[i][i] =='O')

    {

t++;

    }

}

if(t==2)

{

    for(i=0;i<3;i++)

    {

if(isdigit(nivi[i][i])!=0)

{

    return(nivi[i][i]);

}

    }

}

//2ndry diagnal

t=0;

for(i=0;i<3;i++)

{

    if(nivi[i][2-i] =='O')

    {

t++;

    }

}

if(t==2)

{

    for(i=0;i<3;i++)

    {

if(isdigit(nivi[i][2-i])!=0)

{

    return(nivi[i][2-i]);

}

    }

}

return(0);

    }

    int intelligent()

    {

char t;

int t1;

t=check\_comp\_X();

t1=convert(t);

if(t1!=0)

{

    return((int)t1);

}

t=check\_comp\_O();

t1=convert(t);

if(t1!=0)

{

    return((int)t1);

}

t=finale();

t1=convert((int)t);

return(t1);

    }

    char finale()

    {

int i;

if(isdigit(square[5]))

{

    return(square[5]);

}

if(isdigit(square[2]))

{

    return(square[2]);

}

if(isdigit(square[4]))

{

    return(square[4]);

}

if(isdigit(square[6]))

{

    return(square[6]);

}

if(isdigit(square[8]))

{

    return(square[8]);

}

for(i=1;i<=9;i++)

{

    if(isdigit(square[i])!=0)

    {

return(square[i]);

    }

}

return 0; //may give error

    }

    int dev()

    {

if(square[5]=='O')

{

    randomize();

    int t=1,x;

    while(t==1)

    {

x=random(9)+1;

if(x==1)

{

    return((int)x);

}

if(x==3)

{

    return((int)x);

}

if(x==7)

{

    return((int)x);

}

if(x==9)

{

    return((int)x);

}

    }

}

else

{

    return((int)5);

}

return 0; //may give error

    }

    int intelligent2()

    {

char t;

int t1;

t=check\_comp\_O();

t1=convert(t);

if(t1!=0)

{

    return((int)t1);

}

t=check\_comp\_X();

t1=convert(t);

if(t1!=0)

{

    return((int)t1);

}

t=finale();

t1=convert((int)t);

return(t1);

    }

    int checkwin2()

    {

if (square[1] == square[2] && square[2] == square[3])

    return -1;

else if (square[4] == square[5] && square[5] == square[6])

    return -1;

else if (square[7] == square[8] && square[8] == square[9])

    return -1;

else if (square[1] == square[4] && square[4] == square[7])

    return -1;

else if (square[2] == square[5] && square[5] == square[8])

    return -1;

else if (square[3] == square[6] && square[6] == square[9])

    return -1;

else if (square[1] == square[5] && square[5] == square[9])

    return -1;

else if (square[3] == square[5] && square[5] == square[7])

    return -1;

else if (square[1] != '1' && square[2] != '2' && square[3] != '3'

&& square[4] != '4' && square[5] != '5' && square[6] != '6'

&& square[7] != '7' && square[8] != '8' && square[9] != '9')

    return 0;

else

    return -1;

    }

    void board2()

    {

clrscr();

cout << endl;

cout<<"\t\t\t\t THE TIC TAC TOE"<<endl<<"Class 12th project";

cout<<endl<<endl;

cout<<"\t\t\t\t Happy playing"<<endl;

cout << "\t\t\t           |     |     " << endl;

cout << "\t\t\t        " << square[1] << "  |  " << square[2] << "  |  " << square[3] << endl;

cout << "\t\t\t      \_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" << endl;

cout << "\t\t\t           |     |     " << endl;

cout << "\t\t\t        " << square[4] << "  |  " << square[5] << "  |  " << square[6] << endl;

cout << "\t\t\t      \_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" << endl;

cout << "\t\t\t           |     |     " << endl;

cout << "\t\t\t        " << square[7] << "  |  " << square[8] << "  |  " << square[9] << endl;

cout << "\t\t\t           |     |     " << endl << endl;

    }

    void replay(char\* name)

    {    int k=1;

  restore();

int player = 1,i,choice;

ifstream f1;

f1.open("game.txt",ios::binary);

char mark,\*temp;

do

{

   delay(1000);

    board(name,k);

    player=(player%2)?1:2;

    mark=(player == 1) ? 'O' : 'X';

    f1.read((char\*)&choice,sizeof(choice));

    if (choice == 1 && square[1] == '1')

square[1] = mark;

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

square[2] = mark;

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

square[3] = mark;

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

square[4] = mark;

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

square[5] = mark;

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

square[6] = mark;

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

square[7] = mark;

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

square[8] = mark;

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

square[9] = mark;

    i=checkwin();

    player++;

}while(i==-1);

board(name,k);

player--;

if(i==1 && player==1)

{

    cout<<"==>\a "<<name<<" win ";

return ;

}

else

    {

    if(i==1 && player==2)

    {

     cout<<"==>\a "<<"computer"<<" win ";

     return ;

    }

    cout<<"==>\aGame draw";

return ;

    }}

//Thank you

* **Computer Science with C++ by Sumita Arora.**
* **Ebalaguruswamay**