PROJECT REPORT

**Tic-Tac-Toe Game** 

Group Details

Rezowana Zayed Zahin -201014096

Morium Begum - 201014072

Md. Yusuf Hossain - 201014089

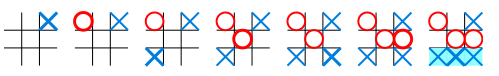
Mizanur Rahman - 201014061

Tazwar Faiaz – 201014052

22.05.2021

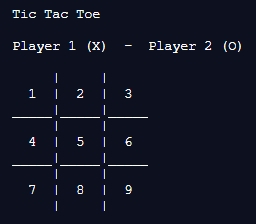
# Project Overview:

Our project name is Tic-Tac-Toe game. This game is very popular and is fairly simple by itself. It’s actually a two players game. In this game, there is a board with 3\*3 squares. In our game, it is 383 squares. The goal of Tic-Tac-Toe is to be one of the players to get three symbols in arrow-horizontally, vertically, or diagonally –on 3\*3 grid. Tic Tac Toe game – is a simple console application without graphics. The game can be start by any one of the two players. The main feature of our project that we can play any level of Tic-Tac-Toe with the robotic arm. We will see the result after we finish the game and the result after playing remain save in a particular file. The most skilled Tic-Tac-Toe competitors put the first "X" in a corner when they get to compete first. This gives the challenger the most the possibilities to make a mistake. If your competitor responds by placing an O anywhere apart from the center, you can guarantee a win. To win this game, if the opponent has completed two steps toward wins, the third step should be blocked by the other player.

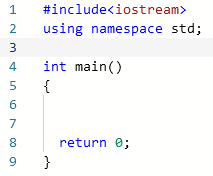


# Object-Oriented Programming Concepts

We have used class for getting user define data type and objects for using as an constructor, inheritance for allowing to create a new class (derived class) from an existing class (base class) and polymorphism which call to a member function will cause a different function to be executed depending on the type of object that invokes the function, virtual function and also file handling to provide mechanism to store the output of a program in a file and to perform various operations on it .



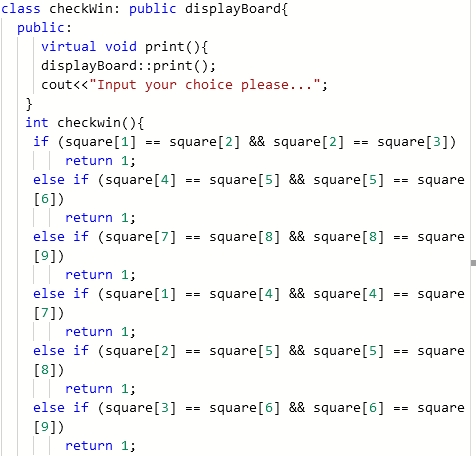
1. At first, we create the basic structure of c++ program. After that we develop the code for getting proper output of tic tac toe game.



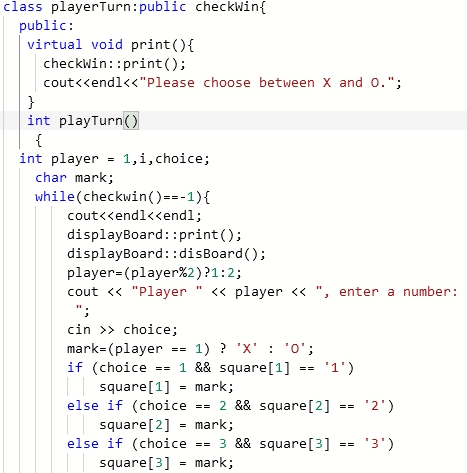
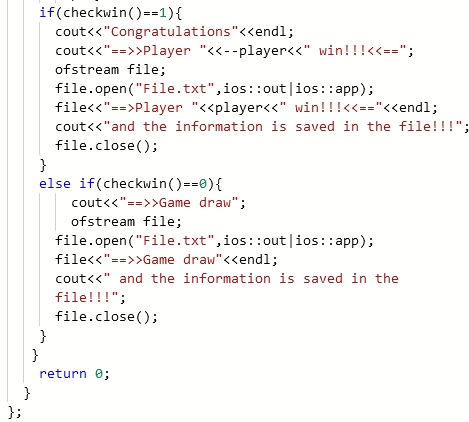
2. From the main function we call three classes – displayBoard(base class), checkWin(derived class), playerTurn(derived class).

3. We have created a class named displayBorad which contains the main board of tic tac toe game (3X3).

4. We also created another class named checkWin which inherits displayBoard and check the input and in this class displayBoard.

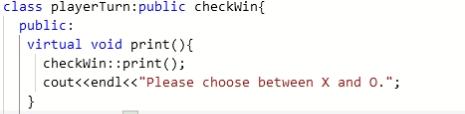
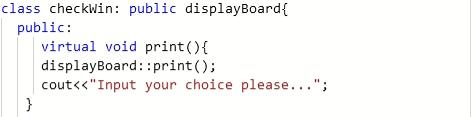


5. After that we used playerTurn class. This function inherits checkWin class and check the input valid or invalid. If it is valid it will take and input in the required position the symbol of the player and show again for another player to input. This process will continue until the full fill of displayBoard and will check the game is draw or any player win in this part we also use file handling part which will store information players win or draw every time.

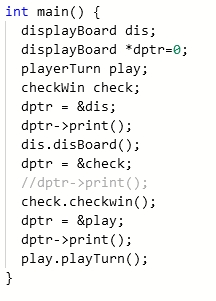


6. If any player can take three input horizontally, vertically, or diagonally, will win the game other wise the game will draw and will show the result.

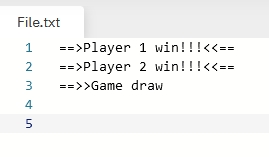
7. In order to use the concept of polymorphism we have to include a virtual function in our code and we did it successful. The virtual function of our code is named print(). The base class means displayBoard have a virtual function named print().

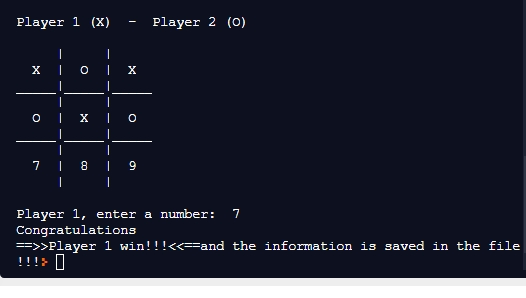
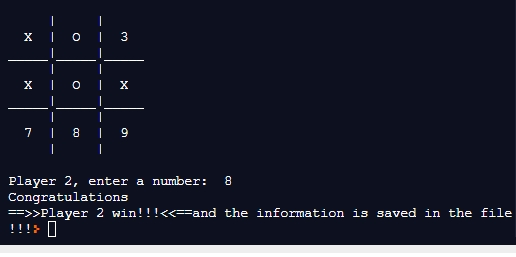


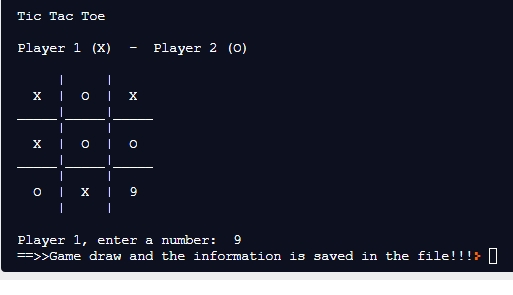
8. Then the derived class checkWin also have that type of function which carries the print function of displayBoard. The same process is happens when the playerTurn class inherits the checkWin class.



9. Finally we add the file handling concept to open a file and write the players result inside the file.







[Here write in details which object-oriented programming concepts are used in your project and how you applied in your code.

You need to write details with example codes.]

# Contribution

Rezowana Zayed Zahin -201014096- Presentation and Inheritance

Morium Begum – 201014072- Polymorphism, virtual function and report

Md. Yusuf Hossain – 201014089- Class and Object

Mizanur Rahman – 201014061- File handling, slide and report

Tazwar Faiaz – 201014052 – slide and function call from main function

# Repl Link: <https://replit.com/join/oiwiyphw-moriumbegum>

# Detailed Source Code:

#include<iostream>

#include<fstream>

using namespace std;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

FUNCTION TO DRAW BOARD OF TIC TAC TOE WITH PLAYERS MARK

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class displayBoard { //base class

public:

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

virtual void print(){

cout<<endl<<"Tic Tac Toe"<<endl<<endl;

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

cout<<endl;

}

void disBoard(){

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;

}

};

class checkWin: public displayBoard{

public:

virtual void print(){

displayBoard::print();

cout<<"Input your choice please...";

}

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;

displayBoard::disBoard();

}

};

class playerTurn:public checkWin{

public:

virtual void print(){

checkWin::print();

cout<<endl<<"Please choose between X and O.";

}

int playTurn()

{

  int player = 1,i,choice;

char mark;

while(checkwin()==-1){

cout<<endl<<endl;

displayBoard::print();

displayBoard::disBoard();

player=(player%2)?1:2;

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

cin >> choice;

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

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 move ";

player--;

}

player++;

if(checkwin()==1){

cout<<"Congratulations"<<endl;

cout<<"==>>Player "<<--player<<" win!!!<<=="<<endl;

ofstream file;

      file.open("File.txt", ios::out | ios::app);

      file << "==>>Player " << player << " win!!! " << endl;

      cout << "and the information is saved";

      file.close();

}

else if(checkwin()==0){

cout<<"==>>Game draw";

ofstream file;

      file.open("File.txt", ios::out | ios::app);

      file << "==>>Game draw " << endl;

      cout << "and the information is saved";

      file.close();

}

}

return 0;

}

};

int main() {

displayBoard dis;

displayBoard \*dptr=0;

playerTurn play;

checkWin check;

dptr = &dis;

dptr->print();

dis.disBoard();

dptr = &check;

//dptr->print();

check.checkwin();

dptr = &play;

dptr->print();

play.playTurn();

}