**Name – Sakshee Agrawal TY-CS-D Batch-1 Roll no -07**

***---------------------------------------------------------------------------***

**Assignment-1 - A:**Tic-tac-toe game with Non-AI approach**.**

**Code-**

#include <iostream>

#include <stdlib.h>

using namespace std;

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

char choice;

int row,column;

char turn = 'X';

bool draw = false;

string ply1,ply2;

void display\_board(){

cout<<"\n\t------------------------------------";

cout<<"\n\t\t"<<ply1<<" [X]\t "<<ply2<<" [O]\n";

cout<<"\t------------------------------------\n\n";

cout<<"\t\t | | \n";

cout<<"\t\t "<<board[0][0]<<" | "<<board[0][1]<<" | "<<board[0][2]<<" \n";

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

cout<<"\t\t | | \n";

cout<<"\t\t "<<board[1][0]<<" | "<<board[1][1]<<" | "<<board[1][2]<<" \n";

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

cout<<"\t\t | | \n";

cout<<"\t\t "<<board[2][0]<<" | "<<board[2][1]<<" | "<<board[2][2]<<" \n";

cout<<"\t\t | | \n";

}

void player\_turn(){

if(turn == 'X'){

cout<<"\n\n\t"<<ply1<<"'s [X] turn : ";

}

else if(turn == 'O'){

cout<<"\n\n\t"<<ply2<<"'s [O] turn : ";

}

cin>> choice;

switch(choice){

case '1': row=0; column=0; break;

case '2': row=0; column=1; break;

case '3': row=0; column=2; break;

case '4': row=1; column=0; break;

case '5': row=1; column=1; break;

case '6': row=1; column=2; break;

case '7': row=2; column=0; break;

case '8': row=2; column=1; break;

case '9': row=2; column=2; break;

default:

cout << "Invalid Move. Please choose a number between 1 and 9.\n";

player\_turn();

return;

}

if(turn == 'X' && board[row][column] != 'X' && board[row][column] != 'O'){

board[row][column] = 'X';

turn = 'O';

}else if(turn == 'O' && board[row][column] != 'X' && board[row][column] != 'O'){

board[row][column] = 'O';

turn = 'X';

}else {

cout<<"That cell has been already filled!Please choose another!!\n\n";

player\_turn();

return;

}

display\_board();

}

bool game(){

//checking the win for row and column

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

if(board[i][0] == board[i][1] && board[i][0] == board[i][2] || board[0][i] == board[1][i] && board[0][i] == board[2][i])

return false;

//checking the win for both diagonal

if(board[0][0] == board[1][1] && board[0][0] == board[2][2] || board[0][2] == board[1][1] && board[0][2] == board[2][0])

return false;

//Checking if board has been filled or not

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

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

if(board[i][j] != 'X' && board[i][j] != 'O')

return true;

draw = true; //if any player has not won and board has been filled then the game is draw

return false;

}

void play(){

cout<<"\n\t\t\tEnter names - \n";

cout<<"\t\t\t\tPlayer 1 - ";

cin>>ply1;

cout<<"\n\t\t\t\tPlayer 2 - ";

cin>>ply2;

cout<<"\n\tEnter your move by choosing a number between 1-9 representing the board position.\n";

display\_board();

while(game()){

player\_turn();

}

if(turn == 'O' && draw == false){

cout<<"\n\nCongratulations! "<<ply1<<" has won the game";

}

else if(turn == 'X' && draw == false){

cout<<"\n\nCongratulations! "<<ply2<<" has won the game";

}

else

cout<<"\n\nGAME DRAW!!Nobody has won!!\n\n";

cout<<"\n\nThank you for playing the game!Hope you enjoyed it! :)";

}

int main(){

char restart;

do {

cout<<"\t\t\t\t W E L C O M E T O ";

cout<<"\n\t\t\t T I C -- T A C -- T O E -- G A M E\t\t\t";

cout<<"\n\t\t\t\t FOR 2 PLAYERS\n\t\t\t";

play();

cout<<"\n\n\nDo you want to restart the game? [Y/N]: ";

cin>>restart;

if (restart == 'Y' || restart == 'y') {

// Reset the board

char num = '1';

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

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

board[i][j] = num++;

}

}

turn = 'X';

draw = false;

system("cls");

}

} while (restart == 'Y' || restart == 'y');

return 0;

}