#include "stdio.h"

int main() {

int grid[3][3], row, column, winner, currentPlayer = 1, moveCount = 0;

for (row = 0; row < 3; row++) {

for (column = 0; column < 3; column++) {

grid[row][column] = 0;

}

}

printf("This game is for two players. Play it as Tic Tac Toe. \n");

printf("Trust me, you can't win your opponent.\n\n");

printf("Warning: Attempting to select a taken box will immediately make you lose!\n\n");

while ((winner != 1 && winner != 2) && moveCount < 9) {

for (row = 0; row < 3; row++) {

for (column = 0; column < 3; column++) {

printf(" %d ", grid[row][column]);

}

printf("\n");

}

printf("It's player %d's turn!\n", currentPlayer);

printf("What location you want to choose?\n");

printf("Row: ");scanf("%d", &row);

printf("Column: ");scanf("%d", &column);

if (grid[row - 1][column - 1] != 0) {

moveCount = 0;

switch (currentPlayer) {

case 1: {

printf("Player 1 lost the game by rule breaking! Player 2 wins!\n");

return 0;

break;

}

case 2: {

printf("Player 2 lost the game by rule breaking! Player 1 wins!\n");

return 0;

break;

}

}

}

switch (currentPlayer) {

case 1: {

grid[row - 1][column - 1] = 4;

currentPlayer = 2;

break;

}

case 2: {

grid[row - 1][column - 1] = 7;

currentPlayer = 1;

break;

}

}

for (row = 0; row < 3 && winner == 0; row++) {

if (((grid[row][0] == grid[row][1]) && (grid[row][1] == grid[row][2])) &&

grid[row][0] != 0 && grid[row][2] != 0 && grid[row][3] != 0) {

switch (grid[row][0]) {

case 4: {

winner = 1;

break;

}

case 7: {

winner = 2;

break;

}

}

}

}

for (column = 0; column < 3 && winner == 0; column++) {

if (((grid[0][column] == grid[1][column]) && (grid[1][column] == grid[2][column])) &&

grid[0][column] != 0 && grid[1][column] != 0 && grid[2][column] != 0) {

switch (grid[0][column]) {

case 4: {

winner = 1;

break;

}

case 7: {

winner = 2;

break;

}

}

}

}

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

((grid[0][2] == grid[1][1]) && (grid[1][1] == grid[2][0])) &&

grid[1][1] != 0 && winner == 0) {

switch (grid[1][1]) {

case 4: {

winner = 1;

break;

}

case 7: {

winner = 2;

break;

}

}

}

moveCount++;

if (moveCount == 9 && winner == 0) {

printf("Draw!\n");

}

else if (winner > 0){

printf("The winner is player %d!\n", winner);

}

}

for (row = 0; row < 3; row++) {

for (column = 0; column < 3; column++) {

printf(" %d ", grid[row][column]);

}

printf("\n");

}

return 0;

}