/\*2023FA-ENGR—2304-81001-P00 TSEGEZAB ATAKLTI\*/

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define BOARD\_SIZE 3

void printBoard(char board[BOARD\_SIZE][BOARD\_SIZE]) {

for (int i = 0; i < BOARD\_SIZE; i++) {

for (int j = 0; j < BOARD\_SIZE; j++) {

printf(" %c ", board[i][j]);

if (j < BOARD\_SIZE - 1) {

printf("|");

}

}

printf("\n");

if (i < BOARD\_SIZE - 1) {

printf("-----------\n");

}

}

printf("\n");

}

int isBoardFull(char board[BOARD\_SIZE][BOARD\_SIZE]) {

for (int i = 0; i < BOARD\_SIZE; i++) {

for (int j = 0; j < BOARD\_SIZE; j++) {

if (board[i][j] == ' ') {

return 0; // Board is not full

}

}

}

return 1; // Board is full

}

int checkWinner(char board[BOARD\_SIZE][BOARD\_SIZE], char player) {

for (int i = 0; i < BOARD\_SIZE; i++) {

// Check rows and columns

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

(board[0][i] == player && board[1][i] == player && board[2][i] == player)) {

return 1; // Player wins

}

}

// Check diagonals

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

(board[0][2] == player && board[1][1] == player && board[2][0] == player)) {

return 1; // Player wins

}

return 0; // No winner

}

int makeMove(char board[BOARD\_SIZE][BOARD\_SIZE], int row, int col, char symbol) {

if (row < 0 || row >= BOARD\_SIZE || col < 0 || col >= BOARD\_SIZE || board[row][col] != ' ') {

return 0; // Invalid move

}

board[row][col] = symbol;

return 1; // Move successful

}

void getComputerMove(char board[BOARD\_SIZE][BOARD\_SIZE], char symbol) {

int row, col;

do {

row = rand() % BOARD\_SIZE;

col = rand() % BOARD\_SIZE;

} while (!makeMove(board, row, col, symbol));

}

void playGame() {

char board[BOARD\_SIZE][BOARD\_SIZE] = {{' ', ' ', ' '}, {' ', ' ', ' '}, {' ', ' ', ' '}};

int currentPlayer = 1; // 1 for player, 2 for computer

while (1) {

printBoard(board);

if (currentPlayer == 1) {

getComputerMove(board, 'X');

} else {

getComputerMove(board, 'O');

}

if (checkWinner(board, currentPlayer == 1 ? 'X' : 'O')) {

printBoard(board);

printf("%s wins!\n", currentPlayer == 1 ? "Player" : "Computer");

break;

} else if (isBoardFull(board)) {

printBoard(board);

printf("The game is a tie!\n");

break;

}

currentPlayer = (currentPlayer % 2) + 1;

}

}

int main() {

srand(time(NULL));

playGame();

return 0;

}