/\* A simple Tic Tac Toe game.

2020BY MARTIN\*/

#include <stdio.h>

#include <stdlib.h>

**char** matrix[3][3]; /\* the tic tac toe matrix \*/

**char** check(**void**);

**void** init\_matrix(**void**);

**void** get\_p1\_move(**void**);

**void** get\_p2\_move(**void**);

**void** disp\_matrix(**void**);

**int** main(**void**)

{

**char** done;

printf("This is the game of Tic Tac Toe.\n");

printf("You will be playing against the computer.\n");

done = ' ';

init\_matrix();

**do** {

disp\_matrix();

get\_p1\_move();

done = check(); /\* see if winner \*/

**if**(done!= ' ') **break**; /\* winner!\*/

get\_p2\_move();

done = check(); /\* see if winner \*/

} **while**(done== ' ');

**if**(done=='X') printf("You won!\n");

**else** printf("I won!!!!\n");

disp\_matrix(); /\* show final positions \*/

**return** 0;

}

/\* Initialize the matrix. \*/

**void** init\_matrix(**void**)

{

**int** i, j;

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

**for**(j=0; j<3; j++) matrix[i][j] = ' ';

}

/\* Get a player's move. \*/

**void** get\_p1\_move(**void**)

{

**int** x, y;

printf("Enter X,Y coordinates for your move(X): ");

scanf("%d%\*c%d", &x, &y);

x--; y--;

**if**(matrix[x][y]!= ' '){

printf("Invalid move, try again.\n");

get\_p1\_move();

}

**else** matrix[x][y] = 'X';

}

/\* Get a move from the computer. \*/

**void** get\_p2\_move(**void**)

{

**int** x, y;

printf("Enter X,Y coordinates for your move(O): ");

scanf("%d%\*c%d", &x, &y);

x--; y--;

**if**(matrix[x][y]!= ' '){

printf("Invalid move, try again.\n");

get\_p2\_move();

}

**else** matrix[x][y] = 'O';

}

/\* Display the matrix on the screen. \*/

**void** disp\_matrix(**void**)

{

**int** t;

**for**(t=0; t<3; t++) {

printf(" %c | %c | %c ",matrix[t][0],

matrix[t][1], matrix [t][2]);

**if**(t!=2) printf("\n---|---|---\n");

}

printf("\n");

}

/\* See if there is a winner. \*/

**char** check(**void**)

{

**int** i;

/\* check rows \*/

**if**(matrix[0][0]==matrix[0][1] && matrix[0][0]==matrix[0][2]=='x')

(matrix[1][0]==matrix[1][1] && matrix[1][0]==matrix[1][2]=='x')

matrix[2][0]==matrix[2][1] && matrix[2][0]==matrix[2][2]=='x'or

matrix[0][0]==matrix[1][0] && matrix[0][0]==matrix[2][0]=='x'or

matrix[0][1]==matrix[1][1] && matrix[0][1]==matrix[2][1]=='x'or

matrix[0][2]==matrix[1][2] && matrix[0][2]==matrix[2][2]=='x'or

matrix[0][0]==matrix[1][1] && matrix[1][1]==matrix[2][2]=='x'or

matrix[0][2]==matrix[1][1] && matrix[1][1]==matrix[2][0]=='x')

printf("P1 wins");

**else** **if**(matrix[0][0]==matrix[0][1] && matrix[0][0]==matrix[0][2]=='y'

matrix[1][0]==matrix[1][1] && matrix[1][0]==matrix[1][2]=='y'or

matrix[2][0]==matrix[2][1] && matrix[2][0]==matrix[2][2]=='y'or

matrix[0][0]==matrix[1][0] && matrix[0][0]==matrix[2][0]=='y'or

matrix[0][1]==matrix[1][1] && matrix[0][1]==matrix[2][1]=='y'or

matrix[0][2]==matrix[1][2] && matrix[0][2]==matrix[2][2]=='y'or

matrix[0][0]==matrix[1][1] && matrix[1][1]==matrix[2][2]=='y'or

matrix[0][2]==matrix[1][1] && matrix[1][1]==matrix[2][0]=='y'or)

printf("P2 wins");

**return** ' ';

}