Tic Tac Toe Game in C

This document provides the source code for a simple Tic Tac Toe game implemented in the C programming language. The game is played on a 3x3 grid, where two players take turns placing their respective symbols (X and O) on the board. The first player to get three of their symbols in a row, column, or diagonal wins the game. The code initializes the board, accepts player input, checks for valid moves, updates the board, and determines the winner.

```
#include <stdio.h>


int main()
{
    char board[3][3]; // The game board
    int row, col; // Player's chosen row and column
    char player = 'X'; // The player whose turn it is
    int i, j; // Loop counters

    // Initialize the board
    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 3; j++)
        board[i][j] = '-';
    }

    // Print the board
    printf("Welcome to Tic Tac Toe!\n");
```

```
     printf("To play, enter the row (0-2) and column (0-2) where you want to place your
%c.\n", player);
    printf("   0 1 2\n");
    for (i = 0; i < 3; i++)
    {
        printf("%d ", i);
        for (j = 0; j < 3; j++)
        printf("%c ", board[i][j]);
        printf("\n");
    }

    // Game loop
    while (1)
    {
        // Get the player's move
        printf("Player %c's turn.\n", player);
        printf("Enter row: ");
        scanf("%d", &row);
        printf("Enter column: ");
        scanf("%d", &col);

        // Check if the chosen position is valid
        if (row < 0 || row > 2 || col < 0 || col > 2)
        {
            printf("Invalid position. Please try again.\n");
            continue;
```

```
}
if (board[row][col] != '-')
{
    printf("Position already taken. Please try again.\n");
    continue;
}

// Place the player's symbol on the board
board[row][col] = player;

// Print the updated board
printf("   0 1 2\n");
for (i = 0; i < 3; i++)
{
    printf("%d ", i);
    for (j = 0; j < 3; j++)
    printf("%c ", board[i][j]);
    printf("\n");
}

// Check if the game is over
int game_over = 0;
for (i = 0; i < 3; i++)
{
    // Check rows
        if (board[i][0] != '-' && board[i][0] == board[i][1] && board[i][1] ==
```

```
board[i][2])
            {
                game_over = 1;
                break;
            }
            // Check columns
                if (board[0][i] != '-' && board[0][i] == board[1][i] && board[1][i] ==
board[2][i])
            {
                game_over = 1;
                break;
            }
        }
        // Check diagonals
             if (board[0][0] != '-' && board[0][0] == board[1][1] && board[1][1] ==
board[2][2]) {
            game_over = 1;
        }
             if (board[0][2] != '-' && board[0][2] == board[1][1] && board[1][1] ==
board[2][0]) {
            game_over = 1;
        }

        if (game_over) {
            printf("Player %c wins!\n", player);
            break;
```

```
        }


        // Switch players
        player = (player == 'X') ? 'O' : 'X';
    }
    return 0;
}
```