```
// TacTacToe solution checker by Chip Henderson for SMU CS7343

#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#define NUM_THREADS 3

char gameBoard[9];
int solutionArray[7];
// Note: solution array consists of row, row, row, column, column, column,
diag, diag
char winner;

void intro()
{
    printf("Welcome to Tac-Tac-toe\n");
    printf("Enter a game board such as XOOOXOOOX\n");

    gets(gameBoard);
    printf( "\nYou entered: %s", gameBoard);

}

void *rowCheck()
{
    int i = 0;
    int j;

    for (j = 0; j < 10; j+=3)
    {
        if (gameBoard[j] == gameBoard[j + 1] && gameBoard[j + 1] ==
gameBoard[j + 2]) {
            solutionArray[i] = gameBoard[j];
        }
            else {
                solutionArray[i] = 0;
            }
        i++;
    }
}

void *columnCheck()
{
    int i = 3;
    int j;

    for (j = 0; j < 3; j++)
    {
        if (gameBoard[j] == gameBoard[j + 3] && gameBoard[j + 3] ==
gameBoard[j+6]) {
            solutionArray[i] = gameBoard[j];
        }
            else {
                solutionArray[i] = 0;
            }
        i++;
    }
}

void *diagCheck()
```

```
{

    if (gameBoard[0] == gameBoard[4] && gameBoard[4] == gameBoard[8]) {
        solutionArray[6] = gameBoard[4];
    }
        else if (gameBoard[2] == gameBoard[4] && gameBoard[4] == gameBoard[6]) {
            solutionArray[6] = gameBoard[4];
        }
        else {
            solutionArray[6] = 0;
        }
}

int main()
{
   intro();

   pthread_t tid[NUM_THREADS];
   int tIndex = 0;
   pthread_attr_t attr;

   pthread_attr_init(&attr);

   pthread_create(&tid[0], &attr, rowCheck, NULL);
   pthread_create(&tid[1], &attr, columnCheck, NULL);
   pthread_create(&tid[2], &attr, diagCheck, NULL);

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

        pthread_join(tid[j],NULL);

    }

    int solutionSum = 0;
    int i;
    for (i = 0; i < 7; i++) {
        solutionSum += solutionArray[i];
    }

    if (solutionSum == 0) {
        printf("\nThere is no winner!\n");
    }
    else if (solutionSum % 88 == 0) {
        printf("\nWinner is X!\n");
    }
    else printf("\nWinner is O!\n");

    pthread_exit(NULL);

    return 0;
}
```