```cpp
/*******************************************************************************
 * PROGRAMMER : Ali Eshghi
 * STUDENT ID : 1112261
 * CLASS      : CS1B
 * SECTION    : MW 7:30pm
 * Assign #2  : tic-tac-toe game (multi-dimensional arrays)
 * DUE DATE   : 19 September 2019
 ******************************************************************************/

#include "MyHeader.h"

/*******************************************************************************
 *  CheckWin
 *   This function checks to see if either player has won. Once it is
 *   possible for a win condition to exist, this should run after each a
 *   player makes a play.
 *
 * RETURNSthe character value of the player that won or a value that
 *   indicates a tie.
 ******************************************************************************/

char CheckWin(const char boardAr[][3]) //PROCESS - the function checks boardAr
{

    /*************
     * VARIABLES *
     ************/

    bool    xWinWays; //PROCESS - How token X is considered won
    bool    oWinWays; //PROCESS - How token O is considered won

    int     i; // PROCESS - used in loop
    int     j; // PROCESS - used in loop

    charwonPlayer; // PROCESS - returns the won token

    /***************************************************************************
    * PROCESS - There are 9 ways that the token X can win the game
    ***************************************************************************/
    xWinWays = (((boardAr[0][0] == 'X')  &&
                 (boardAr[1][0] == 'X')  &&
                 (boardAr[2][0] == 'X')) ||

                ((boardAr[0][1] == 'X')  &&
                 (boardAr[1][1] == 'X')  &&
                 (boardAr[2][1] == 'X')) ||

                ((boardAr[0][2] == 'X')  &&
                 (boardAr[1][2] == 'X')  &&
                 (boardAr[2][2] == 'X')) ||

                ((boardAr[0][0] == 'X')  &&
                 (boardAr[0][1] == 'X')  &&
                 (boardAr[0][2] == 'X')) ||

                ((boardAr[1][0] == 'X')  &&
                 (boardAr[1][1] == 'X')  &&
                 (boardAr[1][2] == 'X')) ||

                ((boardAr[2][0] == 'X')  &&
                 (boardAr[2][1] == 'X')  &&
                 (boardAr[2][2] == 'X')) ||

                ((boardAr[0][0] == 'X')  &&
                 (boardAr[1][1] == 'X')  &&
                 (boardAr[2][2] == 'X')) ||

                ((boardAr[0][2] == 'X')  &&
                 (boardAr[1][1] == 'X')  &&
                 (boardAr[2][0] =='X')));


    /***************************************************************************
    * PROCESS - There are 9 ways that the token O can win the game
    ***************************************************************************/
    oWinWays = (((boardAr[0][0] == 'O')  &&
                 (boardAr[1][0] == 'O')  &&
                 (boardAr[2][0] == 'O')) ||

                ((boardAr[0][1] == 'O')  &&
                 (boardAr[1][1] == 'O')  &&
                 (boardAr[2][1] == 'O')) ||

                ((boardAr[0][2] == 'O')  &&
                 (boardAr[1][2] == 'O')  &&
                 (boardAr[2][2] == 'O')) ||

                ((boardAr[0][0] == 'O')  &&
                 (boardAr[0][1] == 'O')  &&
                 (boardAr[0][2] == 'O')) ||

                ((boardAr[1][0] == 'O')  &&
                 (boardAr[1][1] == 'O')  &&
                 (boardAr[1][2] == 'O')) ||

                ((boardAr[2][0] == 'O')  &&
                 (boardAr[2][1] == 'O')  &&
                 (boardAr[2][2] == 'O')) ||

                ((boardAr[0][0] == 'O')  &&
                 (boardAr[1][1] == 'O')  &&
                 (boardAr[2][2] == 'O')) ||

                ((boardAr[0][2] == 'O')  &&
                 (boardAr[1][1] == 'O')  &&
                 (boardAr[2][0] == 'O')));
```

```
    /****************************************************************************
    * PROCESS - Initializes the wonPlayer based on which token has won the game
    ****************************************************************************/
    if(xWinWays)
    {
        wonPlayer = 'X';
    }

    else if(oWinWays)
    {
        wonPlayer = 'O';
    }
    else
    {
        for(i = 0; i < ROW_SIZE; i++)
        {
            for(j = 0; j < COL_SIZE; j++)
            {
                if((boardAr[i][j] == ' ') && (!xWinWays && !oWinWays))
                {
                    wonPlayer = 'K';// For KEEP PLAY
                }
            }
        }
    }



    return wonPlayer;

}
```