

```

1  /*  Program: A3P4 - Tic Tac Toe
2      Author: Tom Stutler
3      Last Date Modified: 3/19/15
4
5      The intent of this program is to use a class tictactoe that will emulate a game
of Tic Tac Toe between two human users,
6      or one human versus the computer.
7  */
8
9  #include <iostream>
10 // #include <cstdlib>
11 // #include <ctime>
12
13 using namespace std;
14
15 class tictactoe
16 {
17 public:
18     tictactoe() : board() {}
19     ///Constructor.
20     void clearboard();
21     ///Resets a used board to all 0's.
22     void showboard();
23     ///Displays the current game board to the user.
24     void getXmove();
25     ///Prompts the user for a row and column they want to put their X on.
26     ///This function validates the move is to a position that is not already used.
27     void getOmove();
28     ///Prompts the user for a row and column they want to put their X on.
29     ///This function validates the move is to a position that is not already used.
30     //void tictactoe::compOmove();
31     ///The computer generates a move for O.
32     ///This function validates the move is to a position that is not already used.
33     int checkwin();
34     ///Checks whether X wins, O wins, if the game is a tie, or still in progress.
35
36 private:
37     int xRow;
38     int xCol;
39     int oRow;
40     int oCol;
41     int board[3][3];
42 };
43
44 int main()
45 {
46     int status;
47     char repeat;
48     bool gameOn;
49     tictactoe game;
50
51     cout << "Time to play Tic Tac Toe!\n\n";
52
53     do {
54         game.clearboard();
55         game.showboard();
56
57         do {
58             game.getXmove();
59             game.showboard();
60             game.getOmove();
61             game.showboard();
62             status = game.checkwin();
63
64             switch (status)
65             {

```

```

66         default:
67             cout << "ERROR IN MAIN() WITH RETURN FROM
tictactoe::checkwin()!!!\n";
68             gameOn = false;
69             break;
70         case 0:
71             cout << "The game is a draw...\n";
72             gameOn = false;
73             break;
74         case -1:
75             cout << "O has won the game!!!\n";
76             gameOn = false;
77             break;
78         case 1:
79             cout << "X has won the game!!!\n";
80             gameOn = false;
81             break;
82         case 2:
83             gameOn = true;
84             break;
85     }
86     } while (gameOn == true);
87
88     //Prompt the user if they want to play another game.
89     cout << "Would you like to play again? (y or n) ";
90     cin >> repeat;
91     } while ((repeat == 'y') || (repeat == 'Y'));
92 }
93
94 void tictactoe::showboard()
95 {
96     for (int r=0; r<3; r++) {
97         for (int c=0; c<3; c++) {
98             if (board[r][c] == -1) {
99                 cout << "\tO";
100             } else if (board[r][c] == 1) {
101                 cout << "\tX";
102             } else {
103                 cout << "\t-";
104             }
105         }
106         cout << endl;
107     }
108 }
109 void tictactoe::clearboard()
110 {
111     for (int r=0; r<3; r++){
112         for (int c=0; c<3; c++){
113             board[r][c] = 0;
114         }
115     }
116 }
117 void tictactoe::getXmove()
118 {
119     bool repeat;
120     do {
121         cout << "Enter X play position (row# column#): ";
122         cin >> xRow >> xCol;
123
124
125         if ((xRow>0) && (xRow<4) && (xCol>0) && (xCol<4)){
126             if (board[xRow-1][xCol-1] == 0) {
127                 board[xRow-1][xCol-1] = 1;
128                 repeat = false;
129             }else{
130                 cout << "The space is not available. Please pick another move.\n";

```

```

131         repeat = true;
132     }
133     } else {
134         cout << "Please enter a valid input.\n";
135         repeat = true;
136     }
137     } while (repeat == true);
138 }
139
140 void tictactoe::getOmove()
141 {
142     bool repeat;
143     do {
144         cout << "Enter O play position (row# column#): ";
145         cin >> oRow >> oCol;
146
147         if ((oRow>0) && (oRow<4) && (oCol>0) && (oCol<4)){
148             if (board[oRow-1][oCol-1] == 0) {
149                 board[oRow-1][oCol-1] = 1;
150                 repeat = false;
151             }else{
152                 cout << "The space is not available. Please pick another move.\n";
153                 repeat = true;
154             }
155         } else {
156             cout << "Please enter a valid input.\n";
157             repeat = true;
158         }
159     } while (repeat == true);
160 }
161
162 int tictactoe::checkwin()
163 {
164     int line;
165
166     //Check if the game is still in progress.
167     for(int r=0; r<3; r++){
168         for(int c=0; c<3; c++){
169             if (board[r][c] == 0){
170                 return 2;
171             }
172         }
173     }
174     //Check each row for a win.
175     for(int r=0; r<3; r++){
176         line = 0;
177         for (int c=0; c<3; c++){
178             line += board[r][c];
179             if (line == 3) {
180                 return 1;
181             } else if (line == -3) {
182                 return -1;
183             } else {
184                 return 0;
185             }
186         }
187     }
188 }

```