#include <stdio.h>  
  
**void**main()  
{  
  **int**i = 0;                                   /\* Loop counter                         \*/  
  **int**player = 0;                              /\* Player number - 1 or 2               \*/  
  **int**go = 0;                                  /\* Square selection number for turn     \*/  
  **int**row = 0;                                 /\* Row index for a square               \*/    
  **int**column = 0;                              /\* Column index for a square            \*/  
  **int**line = 0;                                /\* Row or column index in checking loop \*/  
  **int**winner = 0;                              /\* The winning player                   \*/  
  **char**board[3][3] = {                         /\* The board                            \*/  
                       {'1','2','3'},          /\* Initial values are reference numbers \*/  
                       {'4','5','6'},          /\* used to select a vacant square for   \*/  
                       {'7','8','9'}           /\* a turn.                              \*/  
                     };  
  
   /\* The main game loop. The game continues for up to 9 turns \*/  
   /\* As long as there is no winner                            \*/  
   **for**( i = 0; i<9 && winner==0; i++)  
   {  
      /\* Display the board \*/  
    printf("\n\n");  
      printf(" %c | %c | %c\n", board[0][0], board[0][1], board[0][2]);  
      printf("---+---+---\n");  
      printf(" %c | %c | %c\n", board[1][0], board[1][1], board[1][2]);  
      printf("---+---+---\n");  
      printf(" %c | %c | %c\n", board[2][0], board[2][1], board[2][2]);  
        
      player = i%2 + 1;                           /\* Select player \*/  
   
      /\* Get valid player square selection \*/  
      **do**  
      {  
         printf("\nPlayer %d, please enter the number of the square "  
       "where you want to place your %c: ", player,(player==1)?'X':'O');  
         scanf("%d", &go);  
  
         row = --go/3;                                 /\* Get row index of square      \*/  
         column = go%3;                                /\* Get column index of square   \*/  
      }**while**(go<0 || go>9 || board[row][column]>'9');  
  
      board[row][column] = (player == 1) ? 'X' : 'O';        /\* Insert player symbol   \*/  
  
      /\* Check for a winning line - diagonals first \*/       
      **if**((board[0][0] == board[1][1] && board[0][0] == board[2][2]) ||  
         (board[0][2] == board[1][1] && board[0][2] == board[2][0]))  
        winner = player;  
      **else**  
      /\* Check rows and columns for a winning line \*/  
        **for**(line = 0; line <= 2; line ++)  
          **if**((board[line][0] == board[line][1] && board[line][0] == board[line][2])||  
             (board[0][line] == board[1][line] && board[0][line] == board[2][line]))  
            winner = player;  
        
  
   }  
   /\* Game is over so display the final board \*/  
   printf("\n\n");  
   printf(" %c | %c | %c\n", board[0][0], board[0][1], board[0][2]);  
   printf("---+---+---\n");  
   printf(" %c | %c | %c\n", board[1][0], board[1][1], board[1][2]);  
   printf("---+---+---\n");  
   printf(" %c | %c | %c\n", board[2][0], board[2][1], board[2][2]);  
  
   /\* Display result message \*/  
   **if**(winner == 0)  
      printf("\nHow boring, it is a draw\n");  
   **else**  
      printf("\nCongratulations, player %d, YOU ARE THE WINNER!\n", winner);  
}  
-------------------------

#include<iostream.h>

#include<conio.h>

#include<stdio.h>

struct rail

{

int seatNo;

int isEmpty;

}seat[67];

void reserve(int n);

int arrRowState[15];

void main()

{

for(int i = 0 ; i <67 ; i++)

{

seat[i].seatNo=(i+1);

seat[i].isEmpty=1;

}

for(i=0 ; i<13 ; i++)

arrRowState[i]=5;

arrRowState[13]=2;

arrRowState[14]=67;

char res='y';

do

{

int n;

clrscr();

cout<<"Enter d no of seats u want to reserve : ";

cin>>n;

reserve(n);

cout<<"

Do u want to reserve more seats?";

res=getchar();

}while(res!='n');

}

void reserve(int n)

{

if(n>arrRowState[14])

{

cout<<"Too large group to accomodate";

getch();

return;

}

int flag=0;

int seatbook;

for(int i = 0 ; flag==0&&i<=13 ; i++)

{

if(arrRowState[i] >= n)

{

flag=1;

// cout<<"Following Seats Alloted";

seatbook=(((i)\*5)+(6-arrRowState[i]));

for(int j = 0 ; j < n ; j++)

{

cout<<"

"<<seatbook+j<<"

";

seat[(seatbook+j)].isEmpty=0;

}

arrRowState[i]=arrRowState[i]-n;

arrRowState[14]=arrRowState[14]-n;

}

}

if(flag==0)

{

while(n!=0)

{

int max,rowNo=0;

max=arrRowState[0];

for( int j = 0 ; j<14 ; j++)

{

if(arrRowState[j] > max)

{

max=arrRowState[j];

rowNo=j;

}

}

if(n>max)

{

n=n-max;

seatbook=(((rowNo)\*5)+(6-arrRowState[rowNo]));

arrRowState[rowNo]=arrRowState[rowNo]-max;

for( int j = 0 ; j<max ; j++)

{

cout<<"

"<<(seatbook+j)<<"

";

seat[(seatbook+j)].isEmpty=0;

}

}

else

{

reserve(n);

n=0;

}

}

}

getch();

}

<http://www.sourcecodesworld.com/source/show.asp?ScriptID=919>

<http://www.sourcecodesworld.com/source/show.asp?ScriptID=947>