

The Ships Game

This is a documentation of the Ships game, which is based on the package of ‘Simon Tatham's Portable Puzzle Collection’. Find the hidden ships.

This manual is copyright 2025 Oleg Zaitsev. All rights reserved. You may distribute this documentation under the MIT licence. See appendix A for the licence text in full. This game was developed using the package of ‘Simon Tatham's Portable Puzzle Collection’ (www.chiark.greenend.org.uk/~sgtatham/puzzles). This software is *not* part of the ‘Collection’.

Contents

Chapter 1: How to compile the code	3
Chapter 2: How to play	4
2.1 Ships controls	4
2.2 Ships parameters	5
Appendix A: Licence	6
Index	7

Chapter 1: How to compile the code

The code is written in C. Because it is based on the package of ‘Simon Tatham's Portable Puzzle Collection’, it must be compiled as if it were a part of the package. Specifically, in Linux do the following:

1. Copy the file `ships.c` to the source directory of the ‘Collection’.
2. Add the game specification to the `CMakeLists.txt` file:

```
puzzle(ships
  DISPLAYNAME "Ships"
  DESCRIPTION "Ship-finding puzzle"
  OBJECTIVE "Find the hidden ships."
)
```

3. Execute the commands:

```
cmake -S <source_directory>
cmake --build <build_directory>
```

To run the game, execute the file `ships` in the `<build_directory>`.

Chapter 2: How to play

The aim of the game is to determine the positions of ships in a grid, given the sum totals of ship cells per row or column. A ship is a sequence of grid cells in the horizontal or vertical direction. The ships cannot touch each other horizontally, vertically or diagonally.

The sum totals of rows and columns are specified on the left of and above the grid, respectively. The lengths of the ships to find are listed below the grid, in the line beginning with the word ‘ships’. There are as many ships as the lengths specified.

The end cells of a ship are marked by *triangles* that point along the ship axis in the directions away from the inner cells. The inner cells are marked by *squares*. In the special case of a one-cell ship, the cell is marked by a *rhombus*. To indicate that a cell is occupied by a ship, it can be temporarily filled with a *white background*; to solve the puzzle, the cell must be eventually marked with a triangle, a square or a rhombus. A cell not occupied by a ship can be marked by a *dot*, but this is not required to solve the puzzle (except in the case where dots are used to indicate the limits of a ship, see section 2.1).

2.1 Ships controls

An unfilled cell can be marked occupied by *left-clicking* it. Left-clicking a filled cell removes the mark. The *right* click labels an unfilled cell with a dot indicating that the cell is not occupied, or removes an existing dot. Multiple cells in a row or a column can be marked with dots or cleared from dots by a right *drag*.

Alternatively, the cell mark can be switched between ‘occupied’, ‘not occupied’ and ‘unfilled’ by consecutively pressing *Enter*. The cursor can be moved around the grid by using the *arrow* keys.

The game takes care of labeling the occupied cells with a specific symbol (triangle, square, rhombus) automatically. The player, however, must tell the game where the ship ends by placing a dot in the cell next the end cell of the ship along the ship axis (unless the ship ends at the border). A one-cell ship is marked by placing dots next to its all four sides.

Some cells are marked at the time of generating the puzzle. They can be distinguished by a thicker border. The marks of these cells cannot be changed, with the exception of white filled cells without a symbol, where the symbols are to be determined during the game.

The sum totals for rows and columns can be left-clicked to mark them done (grey them out) or unmark them again. Completed ships are greyed out automatically (which does not necessarily mean, however, that their positions are correct).

(All the actions described in section 2.1 of the documentation of ‘Simon Tatham's Portable Puzzle Collection’ are also available.)

2.2 Ships parameters

These parameters are available from the ‘Custom...’ option on the ‘Type’ menu.

Height, Width

Size of grid in squares.

Difficulty

Controls the difficulty of the generated puzzle. At the Basic and Intermediate levels the one-cell ships are avoided (they are more difficult to find). At the Advanced and Unreasonable levels some of the sum totals for rows and columns may be hidden. The Unreasonable level may require backtracking.

Appendix A: Licence

This software is copyright 2025 Oleg Zaitsev.

This software was developed using the package of ‘Simon Tatham's Portable Puzzle Collection’ (www.chiark.greenend.org.uk/~sgtatham/puzzles). This software is *not* part of the ‘Collection’.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the ‘Software’), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ‘AS IS’, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Index

build	3
compile	3
controls, for Ships	4
copyright	6
licence	6
licence, MIT	6
Linux	3
MIT licence	6
parameters, for Ships	5
Ships	4
Simon Tatham's Portable Puzzle Collection	1, 3, 4, 6