

1 Roll out your own grid-based logic puzzle	6
2 User documentation	7
2.1 PGF Layers	7
2.2 Environments	8
2.2.1 Puzzle environments	8
2.2.1.1 Options	8
2.2.2 Supporting environments	9
2.3 Commands	9
2.3.1 In the grid	9
2.3.2 Presentation	11
2.3.3 Puzzle specific commands	12
2.3.3.1 2D-Sudoku	12
2.3.3.2 Battleship	12
2.3.3.3 Bokkusu	13
2.3.3.4 Bridges	14
2.3.3.5 Chaos Sudoku	14
2.3.3.6 Four Winds	14
2.3.3.7 Hakyuu	14
2.3.3.8 Hitori	15
2.3.3.9 Kakuro	15
2.3.3.10 Kendoku	15
2.3.3.11 Killer Sudoku	15
2.3.3.12 Laser Beam	16
2.3.3.13 Magic Labyrinth	16
2.3.3.14 Magnets	17
2.3.3.15 Masyu	17
2.3.3.16 Minesweeper	18
2.3.3.17 Nonogram	18
2.3.3.18 Number Link	18
2.3.3.19 Resuko	19
2.3.3.19.1 Track tiles	19
2.3.3.20 Schatzsuche	20
2.3.3.21 Skyline	20
2.3.3.22 Slitherlink	20
2.3.3.23 Star Battle	20
2.3.3.24 Stars and Arrows	21
2.3.3.24.1 Arrows	21
2.3.3.25 Sudoku	21
2.3.3.26 Sun and Moon	21
2.3.3.26.1 Howl at the Moon	21
2.3.3.27 Tents and Trees	22
2.3.3.28 Tunnel	22
3 Examples	22
3.1 2D-Sudoku	22
3.1.1 Example	23
3.1.2 Options	23

3.2 Battleship	24
3.2.1 Example	24
3.2.2 Options	25
3.3 Bokkusu	26
3.3.1 Example	26
3.3.2 Options	26
3.4 Bridges	27
3.4.1 Example	27
3.4.2 Options	28
3.5 Chaos Sudoku	29
3.5.1 Example	29
3.5.2 Options	30
3.6 Four Winds	31
3.6.1 Example	31
3.6.2 Options	32
3.7 Hakyuu	32
3.7.1 Example	32
3.7.2 Options	34
3.8 Hitori	34
3.8.1 Example	34
3.8.2 Options	35
3.9 Kakuro	36
3.9.1 Example	36
3.9.2 Options	37
3.10 Kendoku	37
3.10.1 Example	37
3.10.2 Options	39
3.11 Killer Sudoku	40
3.11.1 Example	40
3.11.2 Options	41
3.12 Laser Beam	42
3.12.1 Example	42
3.12.2 Options	43
3.13 Magic Labyrinth	43
3.13.1 Example	44
3.13.2 Options	44
3.14 Magnets	45
3.14.1 Example	45
3.14.2 Options	46
3.15 Masyu	46
3.15.1 Example	47
3.15.2 Options	47
3.16 Minesweeper	48
3.16.1 Example	48
3.16.2 Options	49
3.17 Nonogram	49
3.17.1 Example	49

3.17.2 Options	50
3.18 Number Link	51
3.18.1 Example	51
3.18.2 Options	52
3.19 Resuko	53
3.19.1 Example	53
3.19.2 Options	54
3.20 Schatzsuche	55
3.20.1 Example	55
3.20.2 Options	55
3.21 Skyline	56
3.21.1 Example	56
3.21.1.1 Variants	57
3.21.1.1.1 Skyline Sudoku	57
3.21.1.1.2 Skyline Sudoku (N*N)	58
3.21.2 Options	60
3.22 Slitherlink	60
3.22.1 Example	60
3.22.2 Options	61
3.23 Star Battle	62
3.23.1 Example	62
3.23.2 Options	63
3.24 Stars and Arrows	63
3.24.1 Example	64
3.24.2 Options	64
3.25 Sudoku	65
3.25.1 Example	65
3.25.2 Options	66
3.25.3 Supporting bash scripts	67
3.25.3.1 createlpsudoku	67
3.25.3.2 lpsmag	67
3.26 Sun and Moon	68
3.26.1 Example	68
3.26.2 Options	68
3.27 Tents and Trees	69
3.27.1 Example	69
3.27.2 Options	70
3.28 Tunnel	71
3.28.1 Example	71
3.28.2 Options	71
4 Implemenation	72
4.1 logicpuzzle.sty	72
4.1.1 Package initialization	72
4.1.2 Defining options	74
4.1.3 Defining colors	75
4.1.4 Drawing grids	76

4.1.5	Drawing the puzzle background	76
4.1.6	In the grid	77
4.1.7	Around the grid	89
4.1.8	Presentation	91
4.1.9	Misc macros	92
4.2	lpenv.sty	97
5	References	157
6	Change History	160
7	Index	162

1 Roll out your own grid-based logic puzzle

As an example we take a look at the former `bokkusu.sty` package. First, we ignore the LPPL license stuff.

```
\ProvidesPackage{bokkusu}[2013/03/25 bokkusu.sty v1.2 - Josef Kleber (C) 2013]%
\RequirePackage{logicpuzzle}%
```

We wrote a package `bokkusu.sty` with version number `v1.2` and date `2013/03/25` and added a copyright remark. We need to load the code base package `logicpuzzle.sty`.

```
\newcommand*\LP@BK@init@prefix{\LP@BK}%
\newcommand*\LP@BK@init@package{bokkusu}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{rows}{5}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{columns}{5}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{scale}{1}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{counterstyle}{none}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{color}{black}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{bgcolor}{}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{width}{6.7cm}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{cvmoffset}{-38pt}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{title}{}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{titleindent}{0.75cm}%
\LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{titlewidth}{5.85cm}%
\LP@define@choicekey@fontsize{\LP@BK@init@prefix}{\LP@BK@init@package}{Large}%
\ExecuteOptionsX{rows,columns,width,fontsize,scale,color,bgcolor,cvmoffset,
counterstyle,title,titleindent,titlewidth}%
\ProcessOptionsX\relax%
```

We save the package prefix and name in a macro for easy change. Then we define the options for package `bokkusu.sty` and the environment `bokkusu`, which are executed afterwards to create the macros for the option values.

```
\let\valueH\LP@bottomrow%
\let\valueV\LP@leftcolumn%
\let\sumH\LP@toprow%
\let\sumV\LP@rightcolumn%
```

We need numbers around the grid. Therefore, we define some aliases for the existing generic commands.

```
\newcommand*\bokkususetup[1]%
{%
  \setkeys{bokkusu.sty}{#1}%
}%
```

We define `\bokkususetup` for resetting the global package options.

Finally, we define the bokkusu environment.

```
\newenvironment{bokkusu}[1][1]{%
  {%
    \setkeys{bokkusu}{#1}%
    \LP@set@package{bokkusu}%
    \LP@set@env@prefix{LP@BK}%
    \setcounter{LP@rows}{\LP@BK@rows}%
    \setcounter{LP@columns}{\LP@BK@columns}%
    \stepcounter{LP@rows}%
    \stepcounter{LP@columns}%
  }
```

We locally set the environment options and the prefix and name of the current puzzle environment. We need to reset the counters for rows and columns, as they might have been altered.

```
\begin{minipage}[t]{\LP@BK@width}%
  \ifthenelse{\equal{\LP@BK@title}{}}{%
    {\par\enspace\par}% empty
  }{\enspace\par\noindent\hspace{\LP@BK@titleindent}\parbox{\LP@BK@titlewidth}
    {\strut\LP@titleformat\LP@BK@title}\vspace{3mm}\par}%
  \begin{tikzpicture}[scale=\LP@BK@scale]%
    \LP@drawbackground{1}{1}{\LP@BK@columns}{\LP@BK@rows}{\LP@BK@bgcolor}%
    \LP@drawgrid{1}{1}{\LP@BK@columns}{\LP@BK@rows}{1cm}%
  }\end{minipage}
```

We start a minipage with width $\{\langle width \rangle\}$. If the user defined a title, we typeset the title and add a vertical space. Then, we draw the puzzle with the help of tikz.sty. We start drawing the background and the grid.

```
{%
  \end{tikzpicture}%
  \LP@drawcounter{\LP@BK@counterstyle}%
  \stepcounter{LP@puzzlecounter}%
  \end{minipage}%
}%
```

Finally, we just end the picture for the puzzle. We draw and step the counter. As last action, we need to close the minipage environment. That's it. Easy, isn't it? You just need to copy this skelton and change or add some code for your specific puzzle.

2 User documentation

2.1 PGF Layers

The logicpuzzle.sty package defines the PGF layers: LPdump, LPbgcolor, LPbackgroundtwo, LPbackground, LPforeground and LPforegroundtwo

Without specifying a special layer, the standard main layer is used. The LPbackground and LPforeground layers can be accessed with the puzzlebackground

and `puzzleforeground` environments. The `LPbgcolor` is and should only be used for the background color of the grid.

All layers can also be accessed with the generic PGF method:

```
\begin{pgfonlayer}{layer}
...
\end{pgfonlayer}
```

Order: `LPdump` → `LPbgcolor` → `LPbackgroundtwo` → `LPbackground` → `main` → `LPforeground` → `LPforegroundtwo`

So, if you are in the need to place something behind `LPbackground` or in front of `LPforeground`, you can use the `LPbackgroundtwo` and `LPforegroundtwo` layers. You can hide elements like help nodes behind the background color on the `LPdump` layer.

2.2 Environments

2.2.1 Puzzle environments

`logicpuzzle` The `logicpuzzle` environment is the generic environment for typesetting logic puzzles. With the optional argument of the environment, you can reset the options with local scope. Here, a blank grid is created. Furthermore, there are the other puzzle environments. They have their own set of options, that is also different option values and defaults! These can be changed with the `\puzzlesetup` commands with global scope or in the optional argument of the environment with local scope.

2.2.1.1 Options

`rows` [5] defines the number of rows in the grid.

`columns` [5] specifies the number of columns in the grid

`width` [5.1cm] sets the width of the minipage, in which the grid is typeset.

`scale` [1] scales the size of the grid in the minipage.

`fontsize` [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: `tiny`, `scriptsize`, `footnotesize`, `small`, `normalsize`, `large`, `Large`, `LARGE`, `huge`, `Huge`

`title` [] sets the title of a puzzle.

`titleindent` [0cm] defines the indent of the title.

`titlewidth` [5.1cm] specifies the width of the box the title is set in.

color [] specifies the color for coloring the cells.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

2.2.2 Supporting environments

puzzlebackground The puzzlebackground environment allows you to place elements behind the main layer on the LPbackground layer. This is for example usefull for the `\fillarea` command.

puzzleforeground The puzzleforeground environment allows you to place elements in front of the main layer on the LPforeground layer. This is for example usefull for the `\framearea` command.

2.3 Commands

2.3.1 In the grid

\setcell `\setcell{<column>}{<row>}{<element>}`
sets an `{<element>}` into cell `{<column>}{<row>}` as central node. It is aware of the current values of the surrounding environment options `rows`, `columns`, `scale` and `fontsize`. Furthermore, a check if `{<element>}` is within the grid is applied.

\setcells `\setcells{<csv list>}{<element>}`
sets `{<element>}` into several cells by using the column/row format in `{<csv list>}`. It works for numbers, letters and most graphical objects, with the exception of commands like `\KKR`, which is not a graphic itself, but drawing something into the grid.

\setbigcell `\setbigcell[<fontsize>]{<column>}{<row>}{<element>}`
sets `{<element>}` into a big (2×2) cell `{<column>}{<row>}` as central node. The optional argument `[<fontsize>]` is set to 'Huge' by default.

\setrow `\setrow{<row>}{<csv list>}`
sets the contents of a `{<row>}`. These may be numbers or letters.

\setcolorrow `\setcolorrow{<row>}{<csv list>}`
sets the contents of a `{<row>}`. Furthermore, the background of the cell is filled with color `LP@c@romannumber`. With the number 0, you can black out the grid cell.

\setcolumn `\setcolumn{<column>}{<csv list>}`
sets the contents of a `{<column>}`. These may be numbers or letters.

<code>\setcolorcolumn</code>	<code>\setcolorcolumn{<column>}{<csv list>}</code> sets the contents of a <code>{<column>}</code> . Furthermore, the background of the cell is filled with color LP@c@romannumber.
<code>\setrule</code>	<code>\setrule{<column>}{<row>}{<rule>}</code> sets a calculation rule <code>{<rule>}</code> into the top left corner of cell <code>{<column>}{<row>}</code> . The rule is typeset in inline math mode. You might consider using the <code>\times</code> and <code>\div</code> commands.
<code>\fillcell</code>	<code>\fillcell{<column>}{<row>}</code> fills cell <code>{<column>}{<row>}</code> with the color defined with environment option <code>color</code> ¹ . It is aware of the current values of the surrounding environment options <code>rows</code> , <code>columns</code> , <code>scale</code> and <code>color</code> . Furthermore, a check if the cell is within the grid is applied.
<code>\fillrow</code>	<code>\fillrow{<row>}{<csv list>}</code> fills a <code>{<row>}</code> . In <code>{<csv list>}</code> '1' means 'fill' and '0' means 'don't fill'. Internally, <code>\fillrow</code> uses <code>\fillcell</code> .
<code>\fillcolumn</code>	<code>\fillcolumn{<column>}{<csv list>}</code> fills a <code>{<column>}</code> . In <code>{<csv list>}</code> '1' means 'fill' and '0' means 'don't fill'. Internally, <code>\fillcolumn</code> uses <code>\fillcell</code> .
<code>\filldiagonals</code>	<code>\filldiagonals[<color>]</code> fills the diagonals with the color specified with the optional argument <code>[<color>]</code> (default: yellow!20). Furthermore, it checks for a quadratic grid, otherwise an error message is issued.
<code>\framearea</code>	<code>\framearea{<color>}{<TikZ path>}</code> frames the area given by <code>{<TikZ path>}</code> with color <code>{<color>}</code> . The reference for coordinates is the bottom left corner of the cell.
<pre>\framearea{green}{(2,2) -- (2,3) -- (3,3) -- (3,2) -- (2,2)}</pre>	
<p>This command will color the frame of the grid cell (2,2) green. You should consider using this command in the <code>puzzleforeground</code> environment.</p>	
<code>\fillarea</code>	<code>\fillarea{<color>}{<TikZ path>}</code> fills the area given by <code>{<TikZ path>}</code> with color <code>{<color>}</code> . The reference for coordinates is the bottom left corner of the cell. You should consider using this command in the <code>puzzlebackground</code> environment.
<code>\colorarea</code>	<code>\colorarea{<color>}{<TikZ path>}</code> fills the area given by <code>{<TikZ path>}</code> with color <code>{<color>}</code> – just like <code>\framearea</code> without frame.

¹Therefore, you must define an option `color` in the style file you want to use fill commands

`\framepuzzle` `\framepuzzle[$\langle color \rangle$]`
frames the grid (thicker line) with the color specified with the optional argument [$\langle color \rangle$] (default: black).

`\tikzpath` `\tikzpath{ $\langle column \rangle$ }{ $\langle row \rangle$ }{ $\langle csv list \rangle$ }`
does easily construct a TikZ path. You just need to define a starting point { $\langle column \rangle$ }{ $\langle row \rangle$ } (bottom left corner) and a { $\langle csv list \rangle$ } with direction indicators relative to the current position.

7: up left	8: up	9: up right
4: left	5: no change	6: right
1: down left	2: down	3: down right

```
\framearea{green}{\tikzpath{2}{2}{8,6,2,4}}
```

This command will frame grid cell (2,2) green.

`\xtikzpath` `\xtikzpath{ $\langle column \rangle$ }{ $\langle row \rangle$ }{ $\langle csv list \rangle$ }`
is an evolution of the `\tikzpath` command with slightly different input syntax. In the { $\langle csv list \rangle$ } argument, it expects pairs in the form direction/length. Therefore, you can easily define paths from corner to corner.

```
\framearea{green}{\xtikzpath{2}{2}{8/2,6/2,2/2,4/2}}
```

This command will frame an area defined by the grid cells (2,2) and (3,3) green.

2.3.2 Presentation

`\titleformat` `\titleformat{ $\langle format \rangle$ }`
defines the { $\langle format \rangle$ } of the title. By default, the definition is as follows:

```
\titleformat{\centering\Large\color{blue}}
```

`\puzzlecounter` `\puzzlecounter`
provides the puzzle counter in textual form to use it in `\definecounterstyle`.

`\setpuzzlecounter` `\setpuzzlecounter{ $\langle number \rangle$ }`
resets the puzzle counter, for example before the solutions.

`\definecounterstyle` `\definecounterstyle{ $\langle name \rangle$ }{ $\langle definition \rangle$ }`
allows you to define your own styles. For example, the style left is defined as follows:

```
\definecounterstyle{left}{
  \begin{group}\reversemarginpar\marginnote{
    \tikz\node[shape=rectangle,fill=yellow!40,inner sep=7pt,
```

```

draw,rounded corners=3pt,thick]
{\Huge\puzzlecounter};;][\LP@cvcvoffset]\endgroup}
}

```

To typeset the counter into the margin we use the command `\marginnote`. We need to use the command `\reversemarginpar` to set the counter into the left margin. Of course, we must use this command in a group for local scope. Finally we use `\puzzlecounter` in a `\tikz` node with a vertical offset set with the option `cvcvoffset`.

<code>\setgridlinestyle</code>	<code>\setgridlinestyle{<style>}</code> sets the style of lines used in the grid. By default, the style is set to solid, whereas <code>slitherlink</code> uses dashed.
<code>\setnormallinewidth</code>	<code>\setnormallinewidth{<dimension>}</code> sets the width of the standard lines (default: 0.5pt)
<code>\setthicklinewidth</code>	<code>\setthicklinewidth{<dimension>}</code> sets the width of the 'thicker' lines (default: 1.5pt)







2.3.3 Puzzle specific commands

2.3.3.1 2D-Sudoku




<code>\ddsudokucell</code>	<code>\ddsudokucell{<column>}{<row>}{<number>}</code> sets <code>{<number>}</code> into grid cell <code>{<column>}{<row>}</code> .
<code>\ddsudokusetup</code>	<code>\ddsudokusetup{<options>}</code> resets the options with global scope.

2.3.3.2 Battleship

<code>\placeship</code>	<code>\placeship{<direction>}{<column>}{<row>}{<length>}</code> places complete ships in the grid. It expects the specification of the direction as horizontal (H) or vertical (V). Furthermore, it requires the starting coordinates and the length of the ship.
<code>\placesegment</code>	<code>\placesegment{<column>}{<row>}{<ship segment>}</code> is used for the placement of ship segments in the grid. In the mandatory argument <code>{<ship segment>}</code> , you can use the following commands:

<code>\Ship</code>			<code>\ShipC</code>
<code>\ShipL</code>			<code>\ShipR</code>
<code>\ShipB</code>			<code>\ShipT</code>

<code>\ship</code>	<code>\ship</code> The command <code>\ship</code> was replaced by the <code>\placesegment</code> command. The command <code>\ship</code> is deprecated and should not be used longer. It may still be used, but it is not recommended.
--------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<code>\placewater</code>	<code>\placewater{<column>}{<row>}</code> places water markers (•) in the grid.
<code>\placeisland</code>	<code>\placeisland{<column>}{<row>}</code> places islands () in the grid. The island outlines are created randomly:  ,  , ...
<code>\shipH</code>	<code>\shipH{<csv list>}</code> typesets the horizontal numbers above the grid. It expects a comma-separated list as an argument.
<code>\shipV</code>	<code>\shipV{<csv list>}</code> typesets the vertical numbers beside the grid. It also expects a comma separated list.
<code>\shipbox</code>	<code>\shipbox{<csv list>}</code> defines the number and size of the ships, which are typeset under the grid.
<code>\battleshipsetup</code>	<code>\battleshipsetup{<options>}</code> resets the options with global scope.
<code>\classicgame</code>	<code>\classicgame{<csv list>}</code> typesets a game sheet for playing classic Battleship. It expects a comma separated list with the number and sizes of the ships.

2.3.3.3 Bokkusu

<code>\valueH</code>	<code>\valueH{<csv list>}</code> typesets the numbers left to the grid indicating the values of the cells. It expects a comma-separated list as an argument.
<code>\valueV</code>	<code>\valueV{<csv list>}</code> typesets the numbers below the grid specifying the values of the cells. It also expects a comma separated list.
<code>\sumH</code>	<code>\sumH{<csv list>}</code> typesets the numbers right to the grid indicating the sums of the values of the colored cells. It expects a comma-separated list.
<code>\sumV</code>	<code>\sumV{<csv list>}</code> typesets the numbers above the grid specifying the sums of the values of the colored cells. It expects a comma separated list.
<code>\bokkususetup</code>	<code>\bokkususetup{<options>}</code> resets the options with global scope.

2.3.3.4 Bridges

<code>\bridgesrow</code>	<code>\bridgesrow{<row>}{<csv list>}</code> sets the contents of a bridges <code>{<row>}</code> . These are the numbers indicating how many bridges originate from this specific island.
<code>\bridgescolumn</code>	<code>\bridgescolumn{<column>}{<csv list>}</code> sets the contents of a bridges <code>{<column>}</code> .
<code>\bridge</code>	<code>\bridge[<optional arguments>]marginTikZ path</code> draws the bridges between islands. With the optional argument <code>[<double>]</code> you can draw a double bridge. Furthermore, you can set the color of the bridge with the option <code>[<color>]</code> .
<code>\bridgessetup</code>	<code>\bridgessetup{<options>}</code> resets the options with global scope.

2.3.3.5 Chaos Sudoku

<code>\chaossudokucell</code>	<code>\chaossudokucell{<column>}{<row>}{<number>}</code> sets <code>{<number>}</code> into grid cell <code>{<column>}{<row>}</code> .
<code>\chaossudokusetup</code>	<code>\chaossudokusetup{<options>}</code> resets the options with global scope.

2.3.3.6 Four Winds

<code>\fourwindscell</code>	<code>\fourwindscell{<column>}{<row>}{<number>}</code> sets <code>{<number>}</code> into grid cell <code>{<column>}{<row>}</code> . Furthermore, it draws lines specified in <code>{<csv list>}</code> in the direction/length format.
<code>\fourwindssetup</code>	<code>\fourwindssetup{<options>}</code> resets the options with global scope.

2.3.3.7 Hakyuu

<code>\hakyuucell</code>	<code>\hakyuucell{<column>}{<row>}{<number>}</code> sets <code>{<number>}</code> into grid cell <code>{<column>}{<row>}</code> .
<code>\hakyuusetup</code>	<code>\hakyuusetup{<options>}</code> reset the options with global scope.

2.3.3.8 Hitori

`\hitorisetaup` `\hitorisetaup{⟨options⟩}`
resets the options with global scope.

2.3.3.9 Kakuro

`\kakurorow` `\kakurorow{⟨row⟩}{⟨csv list⟩}`
sets the contents of a kakuro {⟨row⟩}. These may be numbers and the commands `\KKR` or `\Black`.

`\kakurocolumn` `\kakurocolumn{⟨column⟩}{⟨csv list⟩}`
sets the contents of a kakuro {⟨column⟩}.

`\KKR` `\KKR{⟨sumV⟩}{⟨sumH⟩}`
sets the contents of a kakuro cell.

`\Black` `\Black`
blacks out a cell.

`\kakurosetup` `\kakurosetup{⟨options⟩}`
resets the options with global scope.

2.3.3.10 Kendoku

`\kendokucell` `\kendokucell{⟨column⟩}{⟨row⟩}{⟨number⟩}`
sets {⟨number⟩} into grid cell {⟨column⟩}{⟨row⟩}.

`\kendokusetaup` `\kendokusetaup{⟨options⟩}`
resets the options with global scope.

2.3.3.11 Killer Sudoku

`\killersudokucell` `\killersudokucell{⟨column⟩}{⟨row⟩}{⟨number⟩}`
sets {⟨number⟩} into grid cell {⟨column⟩}{⟨row⟩}.

`\killersudokusetaup` `\killersudokusetaup{⟨options⟩}`
resets the options with global scope.

2.3.3.12 Laser Beam

`\laserH` `\laserH{<csv list>}`
typesets the numbers above the grid indicating how many cells are traversed by the laser beam. It expects a comma-separated list as an argument.

`\laserV` `\laserV{<csv list>}`
typesets the numbers left to the grid.

`\mirrorH` `\mirrorH{<csv list>}`
typesets the numbers below the grid indicating how many mirrors are placed in the intersections of this column.

`\mirrorV` `\mirrorV{<csv list>}`
typesets the numbers right to the grid.

`\placearrow` `\placearrow{<column>}{<row>}`
is used for the placement of arrows at the grid frame. The reference for coordinates is the bottom left corner of the cell. In the mandatory argument `{<direction>}`, you can use the following indicators: LeftUp, LeftDown, RightUp, RightDown

`\placecross` `\placecross{<column>}{<row>}`
places a cross in the intersections of the grid.

`\placemirror` `\placemirror{<column>}{<row>}{<direction>}`
places mirrors in the intersections of the grid. In the mandatory argument `{<direction>}`, you can use the following indicators: H, V

`\laser` `\laser[<color>]{<TikZ path>}`
draws the laser beam given by `{<TikZ path>}` with color `[<color>]` (default: red). The reference for coordinates is the bottom left corner of the cell.

```
\laser[green]{(1,2)--(2,3)--(1,4)}
```

You should consider using this command in the `puzzlebackground` environment.

`\laserbeamsetup` `\laserbeamsetup{<options>}`
resets the options with global scope.

2.3.3.13 Magic Labyrinth

`\magiclabyrinthcell` `\magiclabyrinthcell{<column>}{<row>}{<number>}`
sets a number into grid cell `{<column>}{<row>}`.

`\mline` `\mline{<TikZ path>}`
draws a line given by `{<TikZ path>}`.

`\magiclabyrinthsetup` `\magiclabyrinthsetup{<options>}`
resets the options with global scope.

2.3.3.14 Magnets

`\plusH` `\plusH{<csv list>}`
typesets the numbers above the grid indicating how many positive poles are in the respective column. It expects a comma-separated list as an argument.


`\minusH` `\minusH{<csv list>}`
typesets the numbers above the grid indicating how many negative poles are in the respective column.

`\plusV` `\plusV{<csv list>}`
typesets the numbers left to the grid indicating how many positive poles are in the respective row.

`\minusV` `\minusV{<csv list>}`
typesets the numbers left to the grid indicating how many negative pole ares in the respective row.

`\magnetsH` `\magnetsH{<csv list>}`
typesets non-magnetic horizontal plates by using the column/row format in `{<csv list>}`.

`\magnetsV` `\magnetsV{<csv list>}`
typesets non-magnetic vertical plates by using the column/row format in `{<csv list>}`.

`\PMH` `\PMH{<csv list>}`
`\MPH` draws horizontal magnetic plates with  arrangement. It expects the column/row format in `{<csv list>}`. You can typeset the three other magnetic arrangements by using the `\MPH`, `\PMV` and `\MPV` commands.

`\magnetssetup` `\magnetssetup{<options>}`
resets the options with global scope.

2.3.3.15 Masyu

`\masyucell` `\masyucell{<column>}{<row>}{<element>}`
sets an element into grid cell `{<column>}{<row>}`.

`\MasyuW` `\MasyuW`
draws an empty (white) circle.

<code>\MasyuB</code>	<code>\MasyuB</code> draws a black circle.
<code>\masyuline</code>	<code>\masyuline{<TikZ path>}</code> draws a line given by <code>{<TikZ path>}</code> .
<code>\masyusetup</code>	<code>\masyusetup{<options>}</code> resets the options with global scope.

2.3.3.16 Minesweeper

<code>\Mine</code>	<code>\Mine</code> draws a mine. It can be used in commands like <code>\setcell</code> or <code>\setrow!</code>
<code>\minesweeperssetup</code>	<code>\minesweeperssetup{<options>}</code> resets the options with global scope.

2.3.3.17 Nonogram

<code>\nonogramrow</code>	<code>\nonogramrow{<row>}{<csv list>}</code> sets the contents of row <code>{<row>}</code> . In <code>{<csv list>}</code> it expects the column/length format.
<code>\nonogramcolumn</code>	<code>\nonogramcolumn{<column>}{<csv list>}</code> sets the contents of column <code>{<column>}</code> . In <code>{<csv list>}</code> it expects the row/length format.
<code>\nonogramV</code>	<code>\nonogramV{<csv list>}</code> sets the contents of the extra cells left to the grid. By definition, the first number is always typeset next to the grid!
<code>\nonogramH</code>	<code>\nonogramH{<csv list>}</code> sets the contents of the extra cells on top of the grid.
<code>\puzzlestrut</code>	<code>\puzzlestrut</code> serves the height adjustment depending on option <code>extracells</code> when you want to typeset puzzle and solution (without extra cells) next to each other.
<code>\nonogramsetup</code>	<code>\nonogramsetup{<options>}</code> resets the options with global scope.

2.3.3.18 Number Link

<code>\numberlinkcell</code>	<code>\numberlinkcell{<column>}{<row>}{<element>}</code> sets a number or letter into grid cell <code>{<column>}{<row>}</code> .
------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------

`\link` `\link{<{<TikZ path>}}>`
draws a line given by `{<TikZ path>}`.



















`\numberlinksetup` `\numberlinksetup{<options>}`
resets the options with global scope.

2.3.3.19 Resuko

`\resukocell` `\resukocell{<column>}{<row>}{<element>}`
sets the `{<element>}` into grid cell `{<column>}{<row>}`.

2.3.3.19.1 Track tiles

`\Straight` `\StraightH` `\StraightV` You can use the following commands to draw different track tiles, e.g. with the `\resukocell` command:

<code>\Cross</code>		<code>\Straight</code>		<code>\StraightH</code>		<code>\StraightV</code>
<code>\CrossH</code>		<code>\Cross</code>		<code>\CrossH</code>		<code>\CrossV</code>
<code>\CrossV</code>		<code>\CurveTL</code>		<code>\CurveTR</code>		<code>\CurveBL</code>
<code>\CurveTL</code>		<code>\CurveTR</code>		<code>\CurveBL</code>		<code>\CurveBR</code>
<code>\CurveTR</code>		<code>\CurveBR</code>		<code>\Graveltrap</code>		
<code>\CurveBL</code>						
<code>\CurveBR</code>						
<code>\Graveltrap</code>						

`\pitlane` `\pitlane{<column>}{<row>}{<direction>}`
draws the pit lane in grid cell `{<column>}{<row>}` with `{<direction>}` V or H.

`\parkinglot` `\parkinglot{<column>}{<row>}`
draws the parking lot in grid cell `{<column>}{<row>}`.

`\trackH` `\trackH{<csv list>}`
typesets the track tiles below the grid indicating how many different tiles are in the respective column. It expects a comma-separated list as an argument with the format straights/curves/intersections.

`\trackV` `\trackV{<csv list>}`
typesets the track tiles left to the grid.

`\track` `\track{<TikZ path>}`
draws the race track given by `{<TikZ path>}`. The design of the race track is based on [Frédéric's answer to this question](#) on T_EX.sx. The design with auto-generated bridges will only work, if the path is not constructed with an intersection point. It's recommended to start the path on a standard straight and define the path from corner to corner with `\xtikzpath`.

`\resukosetup` `\resukosetup{<options>}`
resets the options with global scope.

2.3.3.20 Schatzsuche

<code>\Diamond</code>	<code>\Diamond</code> draws a diamond. It can be used in commands like <code>\setcell</code> or <code>\setrow</code> !
<code>\schatzsuchesetup</code>	<code>\schatzsuchesetup{<options>}</code> resets the options with global scope.

2.3.3.21 Skyline

<code>\skylineT</code>	<code>\skylineT{<csv list>}</code> typesets the numbers above the grid indicating how many skycrapers are visible. It expects a comma-separated list as an argument.
<code>\skylineB</code>	<code>\skylineB{<csv list>}</code> typesets the numbers below the grid.
<code>\skylineL</code>	<code>\skylineL{<csv list>}</code> typesets the numbers left to the grid.
<code>\skylineR</code>	<code>\skylineR{<csv list>}</code> typesets the numbers right to the grid.
<code>\skylinecell</code>	<code>\skylinecell{<column>}{<row>}{<height>}</code> sets <code>{<height>}</code> into grid cell <code>{<column>}{<row>}</code> .
<code>\skylinesetup</code>	<code>\skylinesetup{<options>}</code> resets the options with global scope.

2.3.3.22 Slitherlink

<code>\slitherlinkcell</code>	<code>\slitherlinkcell{<column>}{<row>}{<number>}</code> sets <code>{<number>}</code> into grid cell <code>{<column>}{<row>}</code> .
<code>\slitherlinksetup</code>	<code>\slitherlinksetup{<options>}</code> resets the options with global scope.

2.3.3.23 Star Battle

<code>\starbattlecell</code>	<code>\starbattlecell{<column>}{<row>}{<element>}</code> sets <code>{<element>}</code> into grid cell <code>{<column>}{<row>}</code> , e.g. the <code>\Star</code> command.
<code>\starbattlesetup</code>	<code>\starbattlesetup{<options>}</code> resets the options with global scope.

2.3.3.24 Stars and Arrows









`\starsH` `\starsH{<csv list>}`
typesets the numbers above the grid indicating how many stars are in the respective column. It expects a comma-separated list as an argument.

`\starsV` `\starsV{<csv list>}`
typesets the numbers left to the grid.

`\Star` `\Star`
draws a star. It can be used in commands like `\setcell` or `\setrow`!

2.3.3.24.1 Arrows

`\Right` `\RightUp` `\Up`
You can use the following commands to draw different arrows:

<code>\LeftUp</code>	<code>\Right</code>		<code>\RightUp</code>		<code>\Up</code>		<code>\LeftUp</code>	
<code>\Left</code>	<code>\Left</code>		<code>\LeftDown</code>		<code>\Down</code>		<code>\RightDown</code>	
<code>\LeftDown</code>								

`\Down` `\starsandarrowssetup{<options>}`
`\starsandarrowssetup` resets the options with global scope.

2.3.3.25 Sudoku

`\lpsudokucell` `\lpsudokucell{<column>}{<row>}{<number>}`
sets `{<number>}` into grid cell `{<column>}{<row>}`.

`\lpsudokusetup` `\lpsudokusetup{<options>}`
resets the options with global scope.

2.3.3.26 Sun and Moon

`\Star` `\Star`
draws a star. It can be used in commands like `\setcell` or `\setrow`!

`\Cloud` `\Cloud`
draws a dark cloud.

2.3.3.26.1 Howl at the Moon

`\Moon` `\MoonT` `\MoonB`
You can use the following commands to draw different illuminated moons:

<code>\MoonR</code>	<code>\Moon</code>		<code>\MoonT</code>		<code>\MoonB</code>	
<code>\MoonL</code>						
<code>\MoonTR</code>	<code>\MoonR</code>		<code>\MoonL</code>		<code>\MoonTR</code>	
<code>\MoonTL</code>						
<code>\MoonBR</code>	<code>\MoonTL</code>		<code>\MoonBR</code>		<code>\MoonBL</code>	
<code>\MoonBL</code>						

`\sunandmoonssetup` `\sunandmoonssetup{<options>}`
resets the options with global scope.

2.3.3.27 Tents and Trees

<code>\tentH</code>	<code>\tentH{<csv list>}</code> typesets the numbers above the grid indicating how many tents are in the respective column. It expects a comma-separated list as an argument.
<code>\tentV</code>	<code>\tentV{<csv list>}</code> typesets the numbers left to the grid.
<code>\Tree</code>	<code>\Tree</code> draws a tree. It can be used in commands like <code>\setcell</code> or <code>\setrow</code> ! The design of the tree is based on Alain Matthes ' answer to this question on T _E X.sx.
<code>\Tent</code>	<code>\Tent</code> draws a tent.
<code>\tentsandtreessetup</code>	<code>\tentsandtreessetup{<options>}</code> resets the options with global scope.

2.3.3.28 Tunnel

<code>\tunnelH</code>	<code>\tunnelH{<csv list>}</code> typesets the numbers above the grid indicating how many tube segments are in the respective column. It expects a comma-separated list as an argument.
<code>\tunnelV</code>	<code>\tunnelV{<csv list>}</code> typesets the numbers left to the grid.
<code>\portal</code>	<code>\portal{<column>}{<row>}</code> is used for the placement of tunnel portals in the grid.
<code>\tube</code>	<code>\tube{<TikZ path>}</code> draws the tunnel tube given by <code>{<TikZ path>}</code> . The reference for coordinates is the center of the cell. The design of the tube is based on Xoff 's answer to this question on T _E X.sx.
<div style="background-color: #ffffcc; padding: 5px;"><code>\tube{(1.5,2.5)--(3.5,2.5)--(3.5,4.5)}</code></div>	
<code>\tunnelsetup</code>	<code>\tunnelsetup{<options>}</code> resets the options with global scope.

3 Examples

3.1 2D-Sudoku

Fill every row, every column and each of the two diagonals – if indicated – with numbers from 1 to SIZE of the grid.

3.1.1 Example

1				
3				4
	4		2	
			3	

1	3	4	5	2
3	2	5	1	4
5	4	3	2	1
2	5	1	4	3
4	1	2	3	5

```

\begin{center}
\begin{ddsudoku}
\framepuzzle
\filldiagonals[orange!50]
\ddsudokucell{1}{5}{1}
\ddsudokucell{1}{4}{3}
\ddsudokucell{2}{3}{4}
\ddsudokucell{4}{1}{3}
\ddsudokucell{4}{3}{2}
\ddsudokucell{5}{4}{4}
\end{ddsudoku}
\hspace{1.5cm}
\begin{ddsudoku}
\framepuzzle
\filldiagonals[orange!50]
\setrow{5}{1,3,4,5,2}
\setrow{4}{3,2,5,1,4}
\setrow{3}{5,4,3,2,1}
\setrow{2}{2,5,1,4,3}
\setrow{1}{4,1,2,3,5}
\end{ddsudoku}
\end{center}

```

3.1.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [**Large**] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [**0cm**] defines the indent of the title.

titlewidth [**5.1cm**] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

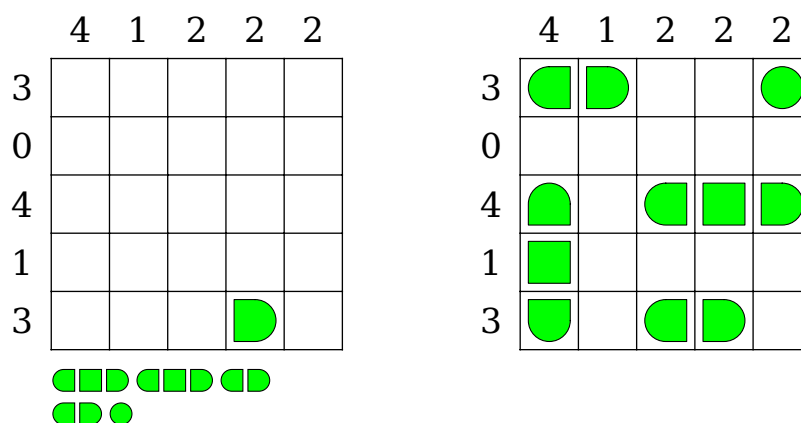
counterstyle [**none**] defines the counter style. Predefined styles: none, left, right

cvoffset [**-23pt**] sets the vertical offset of the counters in the margin.

3.2 Battleship

Try to find the positions of the ships listed below the puzzle. The numbers on the side of the puzzle reveals how many ship segments can be found in the rows and columns. All remaining fields indicate 'water'. Consider the following rules: The ships are arranged horizontally and vertically. No ship touches another ship at any point, not even diagonally.

3.2.1 Example



```
\begin{center}
\begin{battleship}
\placesegment{4}{1}{\ShipR}
\shipH{4,1,2,2,2}
\shipV{3,1,4,0,3}
```



```

\shipbox{3,3,2,2,1}
\end{battleship}
\hspace{1.5cm}
\begin{battleship}
\placeship{V}{1}{1}{3}
\placeship{H}{1}{5}{2}
\placeship{H}{3}{1}{2}
\placeship{H}{3}{3}{3}
\placeship{H}{5}{5}{1}
\shipH{4,1,2,2,2}
\shipV{3,1,4,0,3}
\end{battleship}
\end{center}

```

3.2.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid.

shipcolor [green] sets the color of the ship segments.

width [6cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0.75cm] defines the indent of the title.

titlewidth [5.15cm] specifies the width of the box the title is set in.

sbindent [0.75cm] defines the indent of the ship box below the grid.

sbwidth [5.15cm] specifies the width of the minipage, in which the ships are typeset.

sbshipscale [1] scales the size of the ships in the ship box.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.3 Bokkusu

Black out some of the grid cells. The numbers on the left and the bottom edge of the grid indicate the values of the cells for adding up. The numbers on the right and the top edge of the grid specify the sums of the values of the colored cells.

3.3.1 Example

		7	1	11	9	6	
5							?
4							13
3							5
2							12
1							2
	1	2	3	4	5		

		7	1	11	9	6	
5							?
4							13
3							5
2							12
1							2
	1	2	3	4	5		

```

\begin{center}
\begin{bokkusu}
\valueH{1,2,3,4,5}
\valueV{1,2,3,4,5}
\sumH{7,1,11,9,6}
\sumV{2,12,5,13,?}
\end{bokkusu}
\hspace{1.5cm}
\begin{bokkusu}
\valueH{1,2,3,4,5}
\valueV{1,2,3,4,5}
\sumH{7,1,11,9,6}
\sumV{2,12,5,13,?}
\fillrow{5}{0,0,1,0,0}
\fillrow{4}{1,0,1,1,1}
\fillrow{3}{1,0,0,1,0}
\fillrow{2}{0,0,1,1,1}
\fillrow{1}{0,1,0,0,0}
\end{bokkusu}
\end{center}

```

3.3.2 Options

`rows` [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [6.7cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0.75cm] defines the indent of the title.

titlewidth [5.85cm] specifies the width of the box the title is set in.

color [black] specifies the color for coloring the cells.

bgcolor [] sets the background color of the grid.

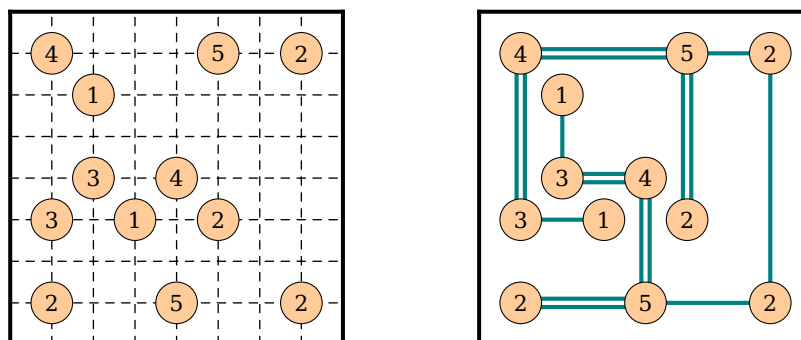
counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-38pt] sets the vertical offset of the counters in the margin.

3.4 Bridges

Connect all the islands (circles) located in the grid by bridges. The bridges may only be routed horizontally and vertically. Islands may be connected by a maximum of two bridges. The bridges must neither overlap nor cross. They may also not be built over islands. The numbers in the islands indicate how many bridges originate from this island. All islands must be fully connected.

3.4.1 Example



```

\colorlet{LP@c@bridge}{Teal}
\begin{center}
\begin{bridges}
\framepuzzle
\bridgesrow{8}{{}},4,{{}},{{}},{{}},5,{{}},2}
\bridgesrow{7}{{}},{{}},1}
\bridgesrow{5}{{}},{{}},3,{{}},4}
\bridgesrow{4}{{}},3,{{}},1,{{}},2}
\bridgesrow{3}{{}},{{}},{{}},{{}},{{}},{{}},{{}}
\bridgesrow{2}{{}},2,{{}},{{}},5,{{}},{{}},2}
\end{bridges}
\hspace{1.5cm}
\begin{bridges}[grid=none]
\framepuzzle
\bridgesrow{8}{{}},4,{{}},{{}},{{}},5,{{}},2}
\bridgesrow{7}{{}},{{}},1}
\bridgesrow{5}{{}},{{}},3,{{}},4}
\bridgesrow{4}{{}},3,{{}},1,{{}},2}
\bridgesrow{3}{{}},{{}},{{}},{{}},{{}},{{}},{{}}
\bridgesrow{2}{{}},2,{{}},{{}},5,{{}},{{}},2}
\bridge[double]{\tikzpath{2}{4}{8,8,8,8,6,6,6,6,2,2,2,2}}
\bridge[double]{\tikzpath{2}{2}{6,6,6,8,8,8,4,4}}
\bridge{\tikzpath{2}{4}{6,6}}
\bridge{\tikzpath{3}{5}{8,8}}
\bridge{\tikzpath{5}{2}{6,6,6,8,8,8,8,8,4,4}}
\end{bridges}
\end{center}

```

3.4.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [6.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [6.1cm] specifies the width of the box the title is set in.

color [green] specifies the color for coloring the islands.

bgcolor [] sets the background color of the grid.

counterstyle [*none*] defines the counter style. Predefined styles: none, left, right

cvoffset [*-23pt*] sets the vertical offset of the counters in the margin.

grid [*dashed*] sets the style of the grid. Possible values: dashed, none, solid

3.5 Chaos Sudoku

Fill the cells of an area with numbers from 1 to N of the N*N grid. Each number can appear only once - in each area, column, row or diagonal if indicated.

3.5.1 Example

4				2
		4	5	
3				

4	3	5	1	2
2	1	3	4	5
5	4	2	3	1
1	2	4	5	3
3	5	1	2	4

```

\begin{center}
\begin{chaossudoku}
\chaossudokucell{1}{1}{3}
\chaossudokucell{1}{5}{4}
\chaossudokucell{3}{2}{4}
\chaossudokucell{4}{2}{5}
\chaossudokucell{5}{5}{2}
\begin{puzzlebackground}
\fillarea{Wheat}{(1,1)--(1,2)--(2,2)--(2,3)--(4,3)--(4,1)
--(1,1)}
\fillarea{HotPink!30}{(1,2)--(1,6)--(3,6)--(3,5)--(2,5)
--(2,2)--(1,2)}
\fillarea{GreenYellow}{(2,3)--(2,5)--(3,5)--(3,4)--(5,4)
--(5,2)--(4,2)--(4,3)--(2,3)}
\fillarea{LightBlue}{(3,4)--(3,6)--(6,6)--(6,5)--(5,5)
--(5,4)--(3,4)}
\fillarea{LightYellow}{(4,1)--(4,2)--(5,2)--(5,5)--(6,5)
--(6,1)--(4,1)}

```

```

\end{puzzlebackground}
\end{chaossudoku}
\hspace{1.5cm}
\begin{chaossudoku}
\setrow{5}{4,3,5,1,2}
\setrow{4}{2,1,3,4,5}
\setrow{3}{5,4,2,3,1}
\setrow{2}{1,2,4,5,3}
\setrow{1}{3,5,1,2,4}
\begin{puzzlebackground}
\fillarea{Wheat}{(1,1)--(1,2)--(2,2)--(2,3)--(4,3)--(4,1)
--(1,1)}
\fillarea{HotPink!30}{(1,2)--(1,6)--(3,6)--(3,5)--(2,5)
--(2,2)--(1,2)}
\fillarea{GreenYellow}{(2,3)--(2,5)--(3,5)--(3,4)--(5,4)
--(5,2)--(4,2)--(4,3)--(2,3)}
\fillarea{LightBlue}{(3,4)--(3,6)--(6,6)--(6,5)--(5,5)
--(5,4)--(3,4)}
\fillarea{LightYellow}{(4,1)--(4,2)--(5,2)--(5,5)--(6,5)
--(6,1)--(4,1)}
\end{puzzlebackground}
\end{chaossudoku}
\end{center}

```

3.5.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bicolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

`cvoffset` [-23pt] sets the vertical offset of the counters in the margin.

3.6 Four Winds

Fill all cells with light rays. These may not intersect. Cells with numbers represent the lighting system that lits horizontally and vertically. The number indicates how many cells are illuminated. Cells with numbers do not count. No cell must remain empty.

3.6.1 Example

2				5
	3			
		3		
2				
			4	

2	T	—	—	5
	3	—	—	
	—	3	—	
2	—		T	
	—	—	4	—

```

\begin{fourwinds}
  \framepuzzle
  \fourwindscell{1}{2}{2}{}
  \fourwindscell{1}{5}{2}{}
  \fourwindscell{2}{4}{3}{}
  \fourwindscell{3}{3}{3}{}
  \fourwindscell{4}{1}{4}{}
  \fourwindscell{5}{5}{5}{}
\end{fourwinds}
\hspace{1.5cm}
\begin{fourwinds}
  \framepuzzle
  \fourwindscell{1}{2}{2}{2/1,6/1}
  \fourwindscell{1}{5}{2}{2/2}
  \fourwindscell{2}{4}{3}{8/1,6/2}
  \fourwindscell{3}{3}{3}{4/1,6/1,2/1}
  \fourwindscell{4}{1}{4}{4/2,6/1,8/1}
  \fourwindscell{5}{5}{5}{4/2,2/3}
\end{fourwinds}

```

3.6.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

color [blue] sets the color of the lines.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.7 Hakyuu

Fill the cells of an area with numbers from 1 to SIZE of the area. If there are two cells with the same number N in a row or a column, there must be at least N cells between those two cells.

3.7.1 Example

```
\begin{center}
\begin{hakyuu}
\hakyuucell{1}{5}{2}
\hakyuucell{3}{5}{6}
\hakyuucell{4}{5}{5}
\hakyuucell{4}{4}{4}
\hakyuucell{1}{3}{3}
\hakyuucell{2}{2}{2}
\hakyuucell{5}{2}{5}
\hakyuucell{4}{1}{1}
\begin{puzzlebackground}
```


2		6	5	
			4	
3				
	2			5
			1	

2	3	6	5	4
1	7	3	4	2
3	1	2	1	3
1	2	1	3	5
2	3	4	1	2

```

\fillarea{Wheat}{(1,1)--(1,4)--(2,4)--(2,1)--(1,1)}
\fillarea{HotPink!30}{(1,4)--(1,6)--(6,6)--(6,5)--(3,5)
--(3,4)--(1,4)}
\fillarea{GreenYellow}{(2,4)--(3,4)--(3,5)--(5,5)--(5,4)
--(4,4)--(4,3)--(2,3)--(2,4)}
\fillarea{LightBlue}{(5,5)--(6,5)--(6,3)--(4,3)--(4,4)
--(5,4)--(5,5)}
\fillarea{LightSalmon!50}{(2,2)--(2,3)--(5,3)--(5,2)
--(2,2)}
\fillarea{LightYellow}{(2,1)--(2,2)--(5,2)--(5,3)--(6,3)
--(6,1)--(2,1)}

\end{puzzlebackground}
\end{hakyuu}
\hspace{1.5cm}
\begin{hakyuu}
\setrow{5}{2,3,6,5,4}
\setrow{4}{1,7,3,4,2}
\setrow{3}{3,1,2,1,3}
\setrow{2}{1,2,1,3,5}
\setrow{1}{2,3,4,1,2}
\begin{puzzlebackground}
\fillarea{Wheat}{(1,1)--(1,4)--(2,4)--(2,1)--(1,1)}
\fillarea{HotPink!30}{(1,4)--(1,6)--(6,6)--(6,5)--(3,5)
--(3,4)--(1,4)}
\fillarea{GreenYellow}{(2,4)--(3,4)--(3,5)--(5,5)--(5,4)
--(4,4)--(4,3)--(2,3)--(2,4)}
\fillarea{LightBlue}{(5,5)--(6,5)--(6,3)--(4,3)--(4,4)
--(5,4)--(5,5)}
\fillarea{LightSalmon!50}{(2,2)--(2,3)--(5,3)--(5,2)
--(2,2)}
\fillarea{LightYellow}{(2,1)--(2,2)--(5,2)--(5,3)--(6,3)
--(6,1)--(2,1)}

\end{puzzlebackground}
\end{hakyuu}

```

```
\end{center}
```

3.7.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bicolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cwoffset [-23pt] sets the vertical offset of the counters in the margin.

3.8 Hitori

Black out some cells according to these specifications: In each row and each column a number may only occur once or can be completely blackened. The blackened cells can touch neither horizontal nor vertical. All non blackened cells must remain connected. Each number has its own color, which otherwise has no meaning.

3.8.1 Example

```
\begin{center}
\begin{hitori}
\framepuzzle
\setcolorrow{5}{2,4,2,1,1}
\setcolorrow{4}{1,3,2,4,1}
\setcolorrow{3}{1,3,3,3,2}
\setcolorrow{2}{4,2,1,3,3}
```

2	4	2	1	1
1	3	2	4	1
1	3	3	3	2
4	2	1	3	3
4	1	2	2	3

2	4		1	
	3	2	4	1
1		3		2
4	2	1	3	
	1		2	3

```

\setcolorrow{1}{4,1,2,2,3}
\end{hitori}
\hspace{1.5cm}
\begin{hitori}
\framepuzzle
\setcolorrow{5}{2,4,0,1,0}
\setcolorrow{4}{0,3,2,4,1}
\setcolorrow{3}{1,0,3,0,2}
\setcolorrow{2}{4,2,1,3,0}
\setcolorrow{1}{0,1,0,2,3}
\end{hitori}
\end{center}

```

3.8.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

`counterstyle` [none] defines the counter style. Predefined styles: none, left, right

`cvoffset` [-23pt] sets the vertical offset of the counters in the margin.

3.9 Kakuro

Enter numbers from 1 to 9 in any order into the blank cells. Here, the given horizontal and vertical sums should result. The zero does not occur. Within a summation, no number can be repeated.

3.9.1 Example

	23	16	10		
14					
16				3	
14					
		8			

	23	16	10		
14	9	1	4		
16	6	5	3	2	
14	8	3	2	1	
		8	7	1	

```

\definecolor{kakuro}{RGB}{155,206,167}
\kakurosetup{color=kakuro}
\begin{center}
\begin{kakuro}
\framepuzzle
\kakurorow{5}{\Black,\KKR{23}{},\KKR{16}{},\KKR{10}{},\Black}
\kakurorow{4}{\KKR{}{14},9,1,4,\KKR{3}{}}
\kakurorow{3}{\KKR{}{16},6,5,3,2}
\kakurorow{2}{\KKR{}{14},8,3,2,1}
\kakurorow{1}{\Black,\KKR{}{8},7,1,\Black}
\end{kakuro}
\hspace{1.5cm}
\begin{kakuro}[solution]
\framepuzzle
\kakurorow{5}{\Black,\KKR{23}{},\KKR{16}{},\KKR{10}{},\Black}
\kakurorow{4}{\KKR{}{14},9,1,4,\KKR{3}{}}
\kakurorow{3}{\KKR{}{16},6,5,3,2}
\kakurorow{2}{\KKR{}{14},8,3,2,1}
\kakurorow{1}{\Black,\KKR{}{8},7,1,\Black}
\end{kakuro}

```

```
\end{center}
```

3.9.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

color [green] specifies the color of the kakuro cells.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

solution [false] You can use the solution also for the puzzle, as the numbers in the cells are only typeset with option solution=true.

3.10 Kendoku

Fill the cells with the numbers from 1 to SIZE of the puzzle. In the top left corner of a framed area, you will find the result of the specified arithmetic function, which is applied on the entered numbers. The numbers may occur only once in each row and column. The numbers of an area may not necessarily be different when they are in different rows or columns.

3.10.1 Example

```
\begin{center}
\begin{kendoku}
```

$4+$	$2\div$	$75\times$		2
			$2\times$	
5	$60\times$			1
$8\times$		$2-$	$1-$	
			$8+$	

$4+$	$2\div$	$75\times$		2
1	4	3	5	2
3	2	5	$2\times$	4
5	$60\times$		1	1
5	3	4	2	1
$8\times$		$2-$	$1-$	
2	5	1	4	3
4	1	2	$8+$	5

```

\framearea{black}{\tikzpath{1}{1}{8,8,6,2,6,2,4,4}}
\framearea{black}{\tikzpath{1}{3}{8,6,2,4}}
\framearea{black}{\tikzpath{1}{4}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{2}{2}{8,8,6,6,2,4,2,4}}
\framearea{black}{\tikzpath{2}{4}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{3}{1}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{3}{4}{8,8,6,6,2,4,2,4}}
\framearea{black}{\tikzpath{4}{1}{8,6,6,2,4,4}}
\framearea{black}{\tikzpath{4}{2}{8,6,6,2,4,4}}
\framearea{black}{\tikzpath{4}{3}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{5}{3}{8,6,2,4}}
\framearea{black}{\tikzpath{5}{4}{8,6,2,4}}
\framearea{black}{\tikzpath{5}{5}{8,6,2,4}}
\setrule{1}{2}{8\times}
\setrule{1}{3}{5}
\setrule{1}{5}{4+}
\setrule{2}{3}{60\times}
\setrule{2}{5}{2\div}
\setrule{3}{2}{2-}
\setrule{3}{5}{75\times}
\setrule{4}{1}{8+}
\setrule{4}{2}{1-}
\setrule{4}{4}{2\times}
\setrule{5}{3}{1}
\setrule{5}{5}{2}
\end{kendoku}
\hspace{1.5cm}
\begin{kendoku}
\framearea{black}{\tikzpath{1}{1}{8,8,6,2,6,2,4,4}}
\framearea{black}{\tikzpath{1}{3}{8,6,2,4}}
\framearea{black}{\tikzpath{1}{4}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{2}{2}{8,8,6,6,2,4,2,4}}
\framearea{black}{\tikzpath{2}{4}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{3}{1}{8,8,6,2,2,4}}

```

```

\framearea{black}{\tikzpath{3}{4}{8,8,6,6,2,4,2,4}}
\framearea{black}{\tikzpath{4}{1}{8,6,6,2,4,4}}
\framearea{black}{\tikzpath{4}{2}{8,6,6,2,4,4}}
\framearea{black}{\tikzpath{4}{3}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{5}{3}{8,6,2,4}}
\framearea{black}{\tikzpath{5}{4}{8,6,2,4}}
\framearea{black}{\tikzpath{5}{5}{8,6,2,4}}
\setrule{1}{2}{8\times}
\setrule{1}{3}{5}
\setrule{1}{5}{4+}
\setrule{2}{3}{60\times}
\setrule{2}{5}{2\div}
\setrule{3}{2}{2-}
\setrule{3}{5}{75\times}
\setrule{4}{1}{8+}
\setrule{4}{2}{1-}
\setrule{4}{4}{2\times}
\setrule{5}{3}{1}
\setrule{5}{5}{2}
\setrow{5}{1,4,3,5,2}
\setrow{4}{3,2,5,1,4}
\setrow{3}{5,3,4,2,1}
\setrow{2}{2,5,1,4,3}
\setrow{1}{4,1,2,3,5}
\end{kendoku}
\end{center}

```

3.10.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid.
Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

`counterstyle` [`none`] defines the counter style. Predefined styles: none, left, right

`cvoffset` [`-23pt`] sets the vertical offset of the counters in the margin.

3.11 Killer Sudoku

Fill the cells with the numbers from 1 to SIZE of the puzzle. The numbers may occur only once in each row, column and colored area if specified. In the top left corner of a framed area, you will find the sum of the entered numbers. The numbers of an area may not necessarily be different, when they are in different rows or columns. But they must be different, when additional colored areas are specified.

3.11.1 Example

7	6	5	
			6
7			
	9		

7	6	5	
3	2	4	1
4	1	3	6
7			
2	4	1	3
1	9	3	2
		2	4

```
\begin{killersudoku}
  \framearea{black}{\tikzpath{1}{1}{8,8,6,6,2,4,2,4}}
  \framearea{black}{\tikzpath{1}{3}{8,8,6,2,2,4}}
  \framearea{black}{\tikzpath{2}{1}{8,6,6,6,2,4,4,4}}
  \framearea{black}{\tikzpath{2}{3}{8,8,6,2,6,2,4,4}}
  \framearea{black}{\tikzpath{3}{2}{8,6,8,6,2,2,4,4}}
  \framearea{black}{\tikzpath{3}{4}{8,6,6,2,4,4}}
  \begin{puzzlebackground}
    \colorarea{orange!20}{\tikzpath{1}{1}{8,8,6,6,2,2,4,4}}
    \colorarea{orange!20}{\tikzpath{3}{3}{8,8,6,6,2,2,4,4}}
  \end{puzzlebackground}
  \setrule{1}{2}{7}
  \setrule{1}{4}{7}
  \setrule{2}{1}{9}
  \setrule{2}{4}{6}
  \setrule{3}{4}{5}
  \setrule{4}{3}{6}
\end{killersudoku}
```



```

\end{killersudoku}
\hspace{1.5cm}
\begin{killersudoku}
  \framearea{black}{\tikzpath{1}{1}{8,8,6,6,2,4,2,4}}
  \framearea{black}{\tikzpath{1}{3}{8,8,6,2,2,4}}
  \framearea{black}{\tikzpath{2}{1}{8,6,6,6,2,4,4,4}}
  \framearea{black}{\tikzpath{2}{3}{8,8,6,2,6,2,4,4}}
  \framearea{black}{\tikzpath{3}{2}{8,6,8,6,2,2,4,4}}
  \framearea{black}{\tikzpath{3}{4}{8,6,6,2,4,4}}
  \begin{puzzlebackground}
    \colorarea{orange!20}{\tikzpath{1}{1}{8,8,6,6,2,2,4,4}}
    \colorarea{orange!20}{\tikzpath{3}{3}{8,8,6,6,2,2,4,4}}
  \end{puzzlebackground}
  \setrule{1}{2}{7}
  \setrule{1}{4}{7}
  \setrule{2}{1}{9}
  \setrule{2}{4}{6}
  \setrule{3}{4}{5}
  \setrule{4}{3}{6}
  \setrow{4}{3,2,4,1}
  \setrow{3}{4,1,3,2}
  \setrow{2}{2,4,1,3}
  \setrow{1}{1,3,2,4}
\end{killersudoku}

```

3.11.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid.
Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bicolor [] sets the background color of the grid.

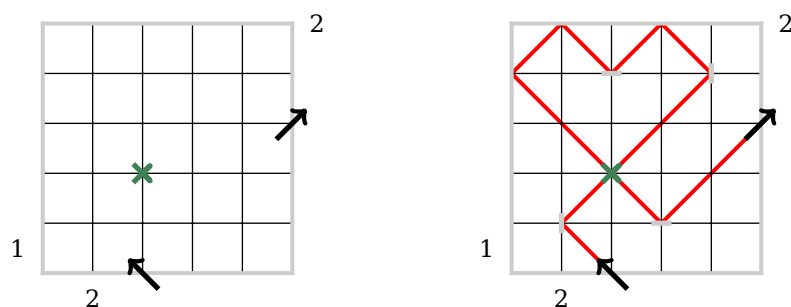
`counterstyle` [`none`] defines the counter style. Predefined styles: none, left, right

`cvoffset` [`-23pt`] sets the vertical offset of the counters in the margin.

3.12 Laser Beam

Draw a laser beam in each grid according to the following guidelines. The beam has to enter or to leave the grid at the arrows. At each intersection, a mirror, on which the laser beam must reflect on one side, can be placed horizontally or vertically. The other side must not be touched by the beam. All locations where the laser crosses are given. The numbers to the left and above the grid indicate how many cells are traversed by the beam in the corresponding row or column. The numbers to the right and below reveal, how many mirrors are found in the intersection of the corresponding row or column.

3.12.1 Example



```
\begin{center}
\begin{laserbeam}
  \laserV{1}
  \laserH{}}
  \mirrorH{,2}
  \mirrorV{,},{,},{,},{,},{,},2}
  \framepuzzle[LP@c@mirror]
  \placearrow{3}{1}{LeftUp}
  \placearrow{6}{4}{RightUp}
  \placecross{3}{3}
\end{laserbeam}
\hspace{1cm}
\begin{laserbeam}
  \laserV{1}
  \laserH{}}
```

```

\mirrorH{{},2}
\mirrorV{{},{},{},{},{},2}
\framepuzzle[LP@c@mirror]
\placearrow{3}{1}{LeftUp}
\placearrow{6}{4}{RightUp}
\placecross{3}{3}
\placemirror{2}{2}{V}
\placemirror{4}{2}{H}
\placemirror{5}{5}{V}
\placemirror{3}{5}{H}
\begin{puzzlebackground}
  \laser{\tikzpath{3}{1}{7,9,9,9,7,1,7,1,3,3,3,9,9}}
\end{puzzlebackground}
\end{laserbeam}
\end{center}

```

3.12.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [6.5cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid.
Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [6.5cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-38pt] sets the vertical offset of the counters in the margin.

3.13 Magic Labyrinth

Enter the numbers 1 to N into the grid. Each number can appear only once in each column and row. Following the labyrinth from the outside inwards, then the given number sequence must be repeated continuously.

3.13.1 Example

			3	
	3			1
			2	
3				

	1	2	3	
2	3			1
		3	1	2
1			2	3
3	2	1		

```

\begin{magiclabyrinth}
  \mline{\xtikzpath{1}{6}{6/5,2/5,4/5,8/4,6/4,2/3,4/3,8/2,6/2,
                    2/1,4/1}}
  \setcells{1/1,2/4,4/5}{3}
  \magiclabyrinthcell{4}{2}{2}
  \magiclabyrinthcell{5}{4}{1}
\end{magiclabyrinth}
\hspace{1.5cm}
\begin{magiclabyrinth}
  \mline{\xtikzpath{1}{6}{6/5,2/5,4/5,8/4,6/4,2/3,4/3,8/2,6/2,
                    2/1,4/1}}
  \setrow{5}{{},1,2,3}
  \setrow{4}{2,3,{}, {},1}
  \setrow{3}{{}, {},3,1,2}
  \setrow{2}{1, {}, {},2,3}
  \setrow{1}{3,2,1}
\end{magiclabyrinth}

```

3.13.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.14 Magnets

Draw magnetic and neutral plates into the grid. The magnetic plates have a positive and a negative pole, neutral plates do not. The same poles must not touch neither horizontal nor vertical. Neutral plates may touch. The numbers to the left and above the grid indicate how many plus or minus poles are present in the respective column or row.

3.14.1 Example

+		2	1	2	2	2	1
	−	2	1	2	2	1	2
1	1		−	+			
2	2					+	−
3	1						+
1	3						−
2	1						
1	2						

+		2	1	2	2	2	1
	−	2	1	2	2	1	2
1	1		−	+			
2	2		+	−		+	−
3	1	+			+	−	+
1	3	−		+	−		−
2	1	+		−	+		
1	2	−			−	+	

```
\magnetssetup{bgcolor=Teal!50}
\begin{magnets}
  \minusH{2,1,2,2,1,2}
  \minusV{2,1,3,1,2,1}
  \plusH{2,1,2,2,2,1}
  \plusV{1,2,1,3,2,1}
  \magnetsH{2/1,2/4,2/5,2/6,3/2,3/3,4/1,4/4,5/5,5/6}
  \magnetsV{1/1,1/3,1/5,2/2,4/5,5/2,6/1,6/3}
  \MPH{2/6}
  \PMH{5/5}
  \MPV{6/3}
```

```

\end{magnets}
\hspace{1.5cm}
\begin{magnets}
  \minusH{2,1,2,2,1,2}
  \minusV{2,1,3,1,2,1}
  \plusH{2,1,2,2,2,1}
  \plusV{1,2,1,3,2,1}
  \MPH{2/6,3/2,4/1}
  \PMH{2/5,5/5,4/4,3/3}
  \MPV{1/1,1/3,6/3}
\end{magnets}

```

3.14.2 Options

rows [6] defines the number of rows in the grid.

columns [6] specifies the number of columns in the grid

width [8.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [8.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid for indicating the neutral areas.

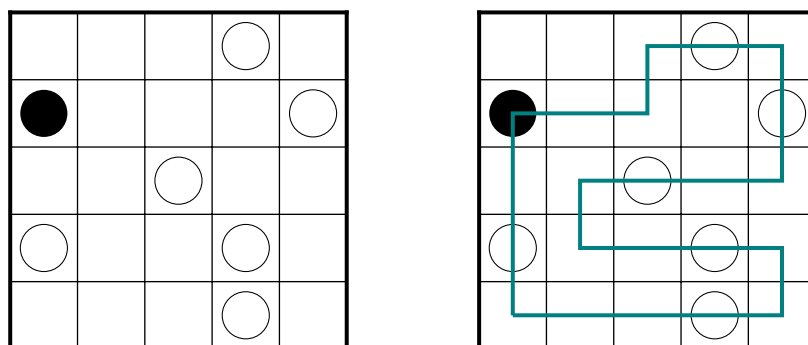
counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.15 Masyu

Draw a line into the grid. The line can only run horizontally and vertically and must pass through all the circles. In cells with a black circle it have to turn in a 90 degree angle and go straight on for at least another cell. The line must go straight through empty circles, but turn left or right in at least one of the neighboring cells. There is no need to go through all cells.

3.15.1 Example



```

\masyusetup{color=Teal}
\begin{masyu}
  \framepuzzle
  \setcells{1/2,3/3,4/1,4/2,4/5,5/4}{\MasyuW}
  \masyucell{1}{4}{\MasyuB}
\end{masyu}
\hspace{1.5cm}
\begin{masyu}
  \framepuzzle
  \setcells{1/2,3/3,4/1,4/2,4/5,5/4}{\MasyuW}
  \masyucell{1}{4}{\MasyuB}
  \masyuline{\xtikzpath{1}{1}{8/3,6/2,8/1,6/2,2/2,4/3,2/1,6/3,
                        2/1,4/4}}
\end{masyu}

```

3.15.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

`titlewidth` [5.1cm] specifies the width of the box the title is set in.

`color` [green] sets the color of the line.

`bgcolor` [] sets the background color of the grid.

`counterstyle` [none] defines the counter style. Predefined styles: none, left, right








`cvoffset` [-23pt] sets the vertical offset of the counters in the margin.

3.16 Minesweeper

Draw a mine in some cells of the grid. The number in a cell indicates how many of the eight neighboring cells contain a mine. A numbered cell does not contain a mine.

3.16.1 Example

	1			
		3	3	
3		4	2	
				0
	2			

	1			
		3	3	
3		4	2	
				0
	2			

```
\begin{center}
\begin{minesweeper}
\framepuzzle
\setrow{5}{{}},1}
\setrow{4}{{}},{ },3,3}
\setrow{3}{3,{ },4,2}
\setrow{2}{{}},{ },{ },{ },0}
\setrow{1}{{}},2}
\end{minesweeper}
\hspace{1.5cm}
\begin{minesweeper}
\framepuzzle
\setrow{5}{{}},1,{ },\Mine,\Mine}
\setrow{4}{{}},\Mine,3,3,\Mine}
\setrow{3}{3,\Mine,4,2}
```



```

\setrow{2}{{}},\Mine,\Mine,{},0}
\setrow{1}{{}},2}
\end{minesweeper}
\end{center}

```

3.16.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.17 Nonogram

Black out some cells of the grid. The black squares form stripes. The number, order, and length is defined by the number sequences on the top and left edge of the grid. Each number represents the length of the strip of black squares in the corresponding row or column. Two stripes are separated by at least one white square.

3.17.1 Example

```

\begin{center}
\begin{nonogram}[rows=10,columns=10,scale=0.35,width=4.8cm,
                fontsize=footnotesize,helpines=5,
                extracells=3]

```

3.17.2 Options

fontsize [Large] specifies the size of the numbers next to the grid.

Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

color [black] sets the color of the lines.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

extracells [5] sets the number of extra cells for the grid.

helplines [5] sets the space between thicker help lines. You can switch off help lines with 0!

solution [false] will switch off extra cells if set to true.

3.18 Number Link

Link all the same numbers or letters in each case by a line. The lines can run horizontally, vertically and in 90 degree angles. Each cell must be traversed by exactly one line. The lines must not intersect.

3.18.1 Example

A		E	D	
B				
		A	E	
	C		D	
	B			C

A		E	D	
B				
		A	E	
	C		D	
	B			C

```
\begin{numberlink}
\framepuzzle
```

```

\numberlinkcell{2}{1}{B}
\numberlinkcell{5}{1}{C}
\numberlinkcell{2}{2}{C}
\numberlinkcell{4}{2}{D}
\numberlinkcell{3}{3}{A}
\numberlinkcell{4}{3}{E}
\numberlinkcell{1}{4}{B}
\numberlinkcell{1}{5}{A}
\numberlinkcell{3}{5}{E}
\numberlinkcell{4}{5}{D}
\end{numberlink}
\hspace{1.5cm}
\begin{numberlink}
  \framepuzzle
  \numberlinkcell{2}{1}{B}
  \numberlinkcell{5}{1}{C}
  \numberlinkcell{2}{2}{C}
  \numberlinkcell{4}{2}{D}
  \numberlinkcell{3}{3}{A}
  \numberlinkcell{4}{3}{E}
  \numberlinkcell{1}{4}{B}
  \numberlinkcell{1}{5}{A}
  \numberlinkcell{3}{5}{E}
  \numberlinkcell{4}{5}{D}
  \link{\tikzpath{4}{3}{8,4,8}}
  \link{\tikzpath{2}{1}{4,8,8,8}}
  \link{\tikzpath{2}{2}{6,2,6,6}}
  \link{\tikzpath{1}{5}{6,2,2,6}}
  \link{\tikzpath{4}{2}{6,8,8,8,4}}
\end{numberlink}

```

3.18.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

color [red] sets the color of the lines.

bgcolor [] sets the background color of the grid.

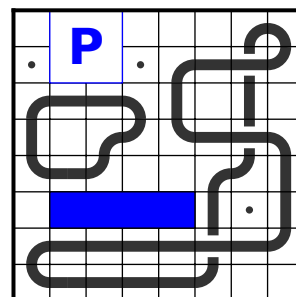
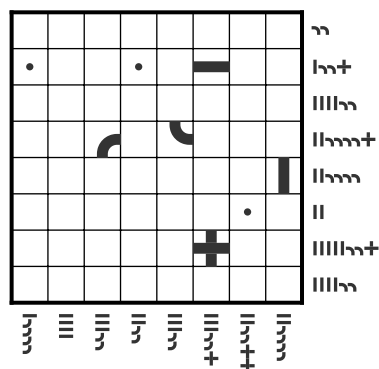
counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.19 Resuko

Complete the given elements in the grid to two race tracks (a race track and a much shorter test track) with pitlane and parking lot. The elements below and to the left of the grid indicate how many straights, curves and intersections are located in the respective columns and rows. The pit lane is always located next to four straights of the circuit on a free area of 1x4 cells. At the end, the parking lot is located on the only free area of 2x2 cells. Both can not be built on a gravel trap.

3.19.1 Example



```
\resukosetup{rows=8,columns=8,width=5.8cm,fontsize=Huge,
  scale=.708}
\begin{resuko}[width=7.4cm]
  \resukocell{1}{7}{\Graveltrap}
  \resukocell{4}{7}{\Graveltrap}
  \resukocell{7}{3}{\Graveltrap}
  \resukocell{6}{2}{\Cross}
  \resukocell{8}{4}{\StraightV}
  \resukocell{6}{7}{\StraightH}
```

```

\resukocell{3}{5}{\CurveBR}
\resukocell{5}{5}{\CurveTR}
\trackH{1/4/0,4/0/0,3/2/0,2/2/0,3/2/0,3/2/1,2/2/2,2/4/0}
\trackV{4/2/0,5/2/1,2/0/0,2/4/0,2/4/1,4/2/0,1/2/1,0/2/0}
\framepuzzle
\end{resuko}
\hspace{1.5cm}
\begin{resuko}
\resukocell{1}{7}{\Graveltrap}
\resukocell{4}{7}{\Graveltrap}
\resukocell{7}{3}{\Graveltrap}
\parkinglot{2}{7}
\pitlane{2}{3}{H}
\track{\tikzpath{2}{4}{6,8,6,8,4,4,4,2,2,6}}
\track{\xtikzpath{2}{1}{6/4,8/3,6/1,8/4,6/1,2/1,4/3,2/2,6/3,
2/3,4/7,2/1,6/1}}

\framepuzzle
\end{resuko}

```

3.19.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

color [blue] sets the color of the pitlane and parking lot.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right








cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.20 Schatzsuche

It's a variant of Minesweeper, just with diamonds instead of mines. Draw a diamond in some cells of the grid. The number in a cell indicates how many of the eight neighboring cells contain a diamond. A numbered cell does not contain a diamond.

3.20.1 Example

	1			
		3	3	
3		4	2	
				0
	2			

	1			
		3	3	
3		4	2	
				0
	2			

```

\begin{center}
\begin{schatzsuche}
\framepuzzle
\setrow{5}{{},1}
\setrow{4}{{},{},3,3}
\setrow{3}{3,{},4,2}
\setrow{2}{{},{},{},{}},0}
\setrow{1}{{},2}
\end{schatzsuche}
\hspace{1.5cm}
\begin{schatzsuche}
\framepuzzle
\setrow{5}{{},1,{},{\Diamond},{\Diamond}}
\setrow{4}{{},{\Diamond},3,3,{\Diamond}}
\setrow{3}{3,{\Diamond},4,2}
\setrow{2}{{},{\Diamond},{\Diamond},{},0}
\setrow{1}{{},2}
\end{schatzsuche}
\end{center}

```

3.20.2 Options

`rows [5]` defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.21 Skyline

There are skyscrapers located in each cell. Try to find out the height of the skyscraper in the respective cell. There are heights of 1 to MAX in every row, every column, and in each of the two diagonals if indicated. Some cells may be empty (parks). The numbers around the grid indicate how many buildings you can see from this position when you look at the skyscraper lineup. Bear in mind that only those skyscrapers are visible which are higher than the ones in front.

3.21.1 Example

		2		3	
3	2				3
4				3	1
		3		3	1

		2		3	
5	4	3	1	2	
4	5	1	2	3	
3	2	3	5	4	3
4	1	2	4	3	5
	3	1	2	5	4
		3		3	1


```

\begin{center}
\begin{skyline}
\skylineB{3, {}, 3, 1, {}}
\skylineL{{}, 4, 3, {}, {}}
\skylineT{{}, {}, 2, {}, 3}
\skylineR{{}, 1, 3, {}, {}}
\skylinecell{1}{3}{2}
\skylinecell{4}{2}{3}
\end{skyline}
\hspace{1cm}
\begin{skyline}
\skylineB{3, {}, 3, 1, {}}
\skylineL{{}, 4, 3, {}, {}}
\skylineT{{}, {}, 2, {}, 3}
\skylineR{{}, 1, 3, {}, {}}
\setrow{5}{5, 4, 3, 1, 2}
\setrow{4}{4, 5, 1, 2, 3}
\setrow{3}{2, 3, 5, 4, 1}
\setrow{2}{1, 2, 4, 3, 5}
\setrow{1}{3, 1, 2, 5, 4}
\end{skyline}
\end{center}

```

3.21.1.1 Variants

3.21.1.1.1 Skyline Sudoku

	4	1	3	2	3	5	3	2	3	
2				8				7		4
3			4			6			8	2
3		2		7					1	3
3					8	2				2
1			2		4		7			4
2				3			4			3
2					1					1
2		3					1	2		3
4			5							3
	4	5	2	5	2	1	2	4	3	

	4	1	3	2	3	5	3	2	3	
2	3	9	6	8	5	1	2	7	4	4
3	1	7	4	9	2	6	3	5	8	2
3	5	2	8	7	3	4	9	6	1	3
3	7	4	3	1	8	2	6	9	5	2
1	9	8	2	6	4	5	7	1	3	4
2	6	5	1	3	9	7	4	8	2	3
2	8	6	7	2	1	3	5	4	9	1
2	4	3	9	5	6	8	1	2	7	3
4	2	1	5	4	7	9	8	3	6	3
	4	5	2	5	2	1	2	4	3	

```

\begin{center}
\begin{skyline}[sudoku, scale=.4]
\skylineB{4, 5, 2, 5, 2, 1, 2, 4, 3}
\skylineL{4, 2, 2, 2, 1, 3, 3, 3, 2}
\skylineT{4, 1, 3, 2, 3, 5, 3, 2, 3}
\skylineR{3, 3, 1, 3, 4, 2, 3, 2, 4}
\end{skyline}
\end{center}

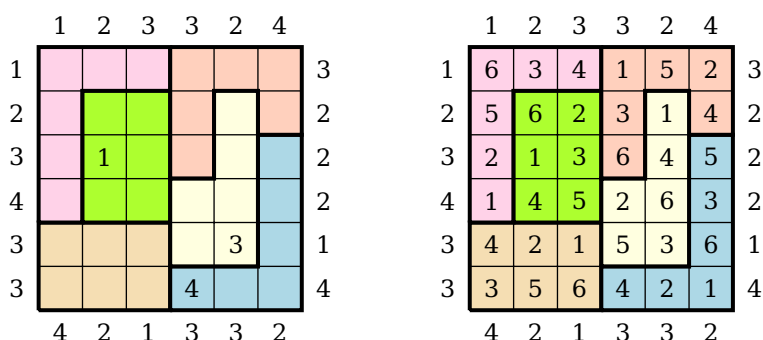
```

```

\setrow{9}{{},{},{},8,{{},{},{},7}
\setrow{8}{{},{},{},4,{{},{},6,{{},{},8}
\setrow{7}{{},{},2,{{},{},7,{{},{},{},1}
\setrow{6}{{},{},{},{},{{},{},8,2}
\setrow{5}{{},{},{},2,{{},{},4,{{},{},7}
\setrow{4}{{},{},{},{},{{},{},3,{{},{},4}
\setrow{3}{{},{},{},{},{},{{},{},1}
\setrow{2}{{},{},3,{{},{},{},{},{{},{},1,2}
\setrow{1}{{},{},{},{},5}
\end{skyline}
\hspace{1cm}
\begin{skyline}[sudoku,scale=.4]
\skylineB{4,5,2,5,2,1,2,4,3}
\skylineL{4,2,2,2,1,3,3,3,2}
\skylineT{4,1,3,2,3,5,3,2,3}
\skylineR{3,3,1,3,4,2,3,2,4}
\setrow{9}{3,9,6,8,5,1,2,7,4}
\setrow{8}{1,7,4,9,2,6,3,5,8}
\setrow{7}{5,2,8,7,3,4,9,6,1}
\setrow{6}{7,4,3,1,8,2,6,9,5}
\setrow{5}{9,8,2,6,4,5,7,1,3}
\setrow{4}{6,5,1,3,9,7,4,8,2}
\setrow{3}{8,6,7,2,1,3,5,4,9}
\setrow{2}{4,3,9,5,6,8,1,2,7}
\setrow{1}{2,1,5,4,7,9,8,3,6}
\end{skyline}
\end{center}

```

3.21.1.1.2 Skyline Sudoku (N*N)



```

\begin{center}
\begin{skyline}[rows=6,columns=6,scale=.58]
\skylineB{4,2,1,3,3,2}
\skylineL{3,3,4,3,2,1}

```

```

\skylineT{1,2,3,3,2,4}
\skylineR{4,1,2,2,2,3}
\skylinecell{2}{4}{1}
\skylinecell{4}{1}{4}
\skylinecell{5}{2}{3}
\begin{puzzlebackground}
  \fillarea{Wheat}{(1,1)--(1,3)--(4,3)--(4,1)--(1,1)}
  \fillarea{HotPink!30}{(1,3)--(1,7)--(4,7)--(4,6)--(2,6)
    --(2,3)--(1,3)}
  \fillarea{GreenYellow}{(2,3)--(2,6)--(4,6)--(4,3)--(2,3)}
  \fillarea{LightBlue}{(4,1)--(7,1)--(7,5)--(6,5)--(6,2)
    --(4,2)--(4,1)}
  \fillarea{LightSalmon!50}{(4,7)--(4,4)--(5,4)--(5,6)--(6,6)
    --(6,5)--(7,5)--(7,7)--(4,7)}
  \fillarea{LightYellow}{(4,2)--(4,4)--(5,4)--(5,6)--(6,6)
    --(6,2)--(4,2)}
\end{puzzlebackground}
\end{skyline}
\hspace{1cm}
\begin{skyline}[rows=6,columns=6,scale=.58]
  \skylineB{4,2,1,3,3,2}
  \skylineL{3,3,4,3,2,1}
  \skylineT{1,2,3,3,2,4}
  \skylineR{4,1,2,2,2,3}
  \setrow{6}{6,3,4,1,5,2}
  \setrow{5}{5,6,2,3,1,4}
  \setrow{4}{2,1,3,6,4,5}
  \setrow{3}{1,4,5,2,6,3}
  \setrow{2}{4,2,1,5,3,6}
  \setrow{1}{3,5,6,4,2,1}
\begin{puzzlebackground}
  \fillarea{Wheat}{(1,1)--(1,3)--(4,3)--(4,1)--(1,1)}
  \fillarea{HotPink!30}{(1,3)--(1,7)--(4,7)--(4,6)--(2,6)
    --(2,3)--(1,3)}
  \fillarea{GreenYellow}{(2,3)--(2,6)--(4,6)--(4,3)--(2,3)}
  \fillarea{LightBlue}{(4,1)--(7,1)--(7,5)--(6,5)--(6,2)
    --(4,2)--(4,1)}
  \fillarea{LightSalmon!50}{(4,7)--(4,4)--(5,4)--(5,6)--(6,6)
    --(6,5)--(7,5)--(7,7)--(4,7)}
  \fillarea{LightYellow}{(4,2)--(4,4)--(5,4)--(5,6)--(6,6)
    --(6,2)--(4,2)}
\end{puzzlebackground}
\end{skyline}
\end{center}

```

3.21.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

sudoku [false] sets rows and columns to 9, in case of true is specified. Additionally the classic Sudoku grid is drawn.

width [6.7cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0.75cm] defines the indent of the title.

titlewidth [5.85cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

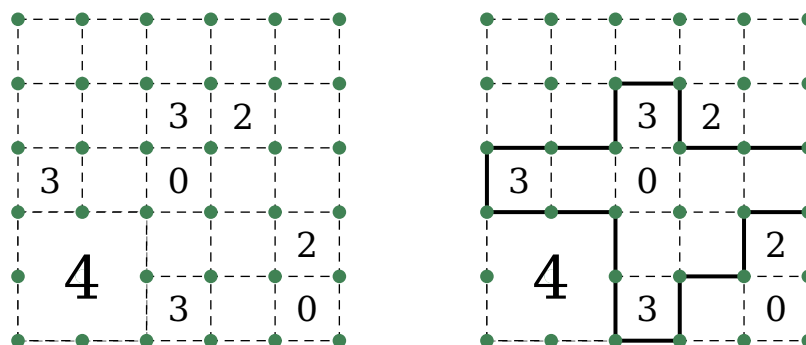
cwoffset [-38pt] sets the vertical offset of the counters in the margin.

3.22 Slitherlink

Draw a closed line into the grid. This line must be on the existing dashed lines, but do not have to go through all grid points. If numbers are present in the grid cells, they indicate how many sides of the cell are touched by the line. The line must not touch or cross itself.

3.22.1 Example

```
\begin{center}
\begin{slitherlink}
\setbigcell{1}{1}{4}
\slitherlinkcell{1}{3}{3}
\slitherlinkcell{3}{1}{3}
\slitherlinkcell{3}{3}{0}
\slitherlinkcell{3}{4}{3}
\slitherlinkcell{4}{4}{2}
\slitherlinkcell{5}{1}{0}
```



```

\slitherlinkcell{5}{2}{2}
\end{slitherlink}
\hspace{1.5cm}
\begin{slitherlink}
\setbigcell{1}{1}{4}
\slitherlinkcell{1}{3}{3}
\slitherlinkcell{3}{1}{3}
\slitherlinkcell{3}{3}{0}
\slitherlinkcell{3}{4}{3}
\slitherlinkcell{4}{4}{2}
\slitherlinkcell{5}{1}{0}
\slitherlinkcell{5}{2}{2}
\framearea{black}{\tikzpath{3}{1}{8,8,4,4,8,6,6,8,6,2,
6,6,2,4,2,4,2,4}}
\end{slitherlink}
\end{center}

```

3.22.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.2cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [**5.2cm**] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

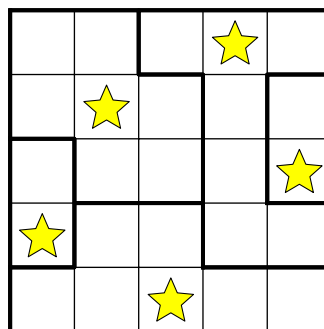
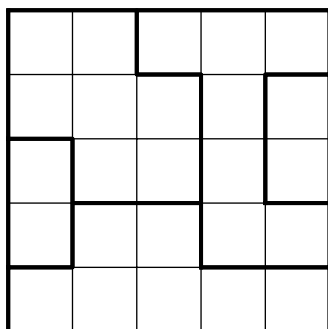
counterstyle [**none**] defines the counter style. Predefined styles: none, left, right

cvoffset [**-23pt**] sets the vertical offset of the counters in the margin.

3.23 Star Battle

Enter exactly one star in each row, each column and each area of the grid. Cells with stars must not touch each other orthogonally or diagonally.

3.23.1 Example



```
\begin{center}
\begin{starbattle}
\framepuzzle
\framearea{black}{\tikzpath{1}{1}{8,6,8,6,6,2,6,6,2,4,4,4,
4,4}}
\framearea{black}{\tikzpath{1}{2}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{1}{4}{8,8,6,6,2,6,2,2,4,4,8,4}}
\framearea{black}{\tikzpath{4}{2}{8,8,8,4,8,6,6,6,2,4,2,2,6,
2,4,4}}
\framearea{black}{\tikzpath{5}{3}{8,8,6,2,2,4}}
\end{starbattle}
\hspace{1.5cm}
\begin{starbattle}
\framepuzzle
\framearea{black}{\tikzpath{1}{1}{8,6,8,6,6,2,6,6,2,4,4,4,
4,4}}
\framearea{black}{\tikzpath{1}{2}{8,8,6,2,2,4}}
\framearea{black}{\tikzpath{1}{4}{8,8,6,6,2,6,2,2,4,4,8,4}}
```

```

\framearea{black}{\tikzpath{4}{2}{8,8,8,4,8,6,6,6,2,4,2,2,6,
                               2,4,4}}
\framearea{black}{\tikzpath{5}{3}{8,8,6,2,2,4}}
\starbattlecell{1}{2}{\Star}
\starbattlecell{2}{4}{\Star}
\starbattlecell{3}{1}{\Star}
\starbattlecell{4}{5}{\Star}
\starbattlecell{5}{3}{\Star}
\end{starbattle}
\end{center}

```

3.23.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.









bgcolor [] sets the background color of the grid.

















counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.24 Stars and Arrows

Enter a star in some empty cells of the grid. Each arrow points to at least one star and every star is referenced by at least one arrow. Arrows point to a whole row, column or diagonal, also through other stars and arrows. The numbers on the left and top of the grid indicate how many stars are located in the row or column.

	1	2	3	2	0
2					
2					
3					
0					
1					

	1	2	3	2	0
2					
2					
3					
0					
1					

3.24.1 Example

```

\begin{center}
\begin{starsandarrows}
\framepuzzle
\starsH{1,2,3,2,0}
\starsV{1,0,3,2,2}
\setrow{5}{\RightDown,{},{},\LeftDown}
\setrow{4}{{},{},\Right,{},{},\LeftUp}
\setrow{2}{{},{},\Up,{},{},{}}
\setrow{1}{\RightUp,{},{},\LeftUp,\Left}
\end{starsandarrows}
\hspace{1.5cm}
\begin{starsandarrows}
\framepuzzle
\starsH{1,2,3,2,0}
\starsV{1,0,3,2,2}
\setrow{5}{\RightDown,\Star,\LeftDown,\Star}
\setrow{4}{{},{},\Right,\Star,\Star,\LeftUp}
\setrow{3}{\Star,\Star,\Star}
\setrow{2}{{},{},\Up,{},{},{}}
\setrow{1}{\RightUp,{},{},\Star,\LeftUp,\Left}
\end{starsandarrows}
\end{center}

```

3.24.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.9cm] sets the width of the minipage, in which the grid is typeset.

scale [**1**] scales the size of the grid in the minipage.

fontsize [**Large**] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [**0cm**] defines the indent of the title.

titlewidth [**5.9cm**] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [**none**] defines the counter style. Predefined styles: none, left, right

cvoffset [**-23pt**] sets the vertical offset of the counters in the margin.

3.25 Sudoku

Well, it's Sudoku – nothing to explain! Fill each row and column with numbers from 1 to 9.

3.25.1 Example

	2	6						
						1	7	
		3	1		6			
	6			5		8		3
		9	2	6	1	7		
5		4		8			6	
			8		4	3		
	4	8						
						9	4	

1	2	6	5	7	8	4	3	9
4	8	5	9	3	2	1	7	6
7	9	3	1	4	6	5	8	2
2	6	1	4	5	7	8	9	3
8	3	9	2	6	1	7	5	4
5	7	4	3	8	9	2	6	1
6	5	2	8	9	4	3	1	7
9	4	8	7	1	3	6	2	5
3	1	7	6	2	5	9	4	8

```
\begin{center}
\begin{lsudoku}
\setrow{9}{{}},2,6,{{}},{{}},{{}},{{}},{{}},{{}}
\setrow{8}{{}},{{}},{{}},{{}},{{}},{{}},1,7,{{}}
\setrow{7}{{}},{{}},3,1,{{}},6,{{}},{{}},{{}}
\setrow{6}{{}},6,{{}},{{}},5,{{}},8,{{}},3
\setrow{5}{{}},{{}},9,2,6,1,7,{{}},{{}}
\setrow{4}{5,{{}},4,{{}},8,{{}},{{}},6,{{}}
```

```

\setrow{3}{{},{},{},8,{{},4,3,{{},{}}
\setrow{2}{{},{},4,8,{{},{},{},{},{}}
\setrow{1}{{},{},{},{},{},{},{},9,4,{{}}
\end{lpsudoku}
\hspace{1.5cm}
\begin{lpsudoku}
\setrow{9}{1,2,6,5,7,8,4,3,9}
\setrow{8}{4,8,5,9,3,2,1,7,6}
\setrow{7}{7,9,3,1,4,6,5,8,2}
\setrow{6}{2,6,1,4,5,7,8,9,3}
\setrow{5}{8,3,9,2,6,1,7,5,4}
\setrow{4}{5,7,4,3,8,9,2,6,1}
\setrow{3}{6,5,2,8,9,4,3,1,7}
\setrow{2}{9,4,8,7,1,3,6,2,5}
\setrow{1}{3,1,7,6,2,5,9,4,8}
\end{lpsudoku}

```

3.25.2 Options

width [9.1cm] sets the width of the minipage, in which the grid is typeset. 9 cells of width 1cm plus a little extra for lines.

scale [1] scales the size of the grid in the minipage. To get a width of 5cm you need to scale by $\frac{5}{9}$

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [9.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.25.3 Supporting bash scripts

3.25.3.1 createlpsudoku

The `createlpsudoku` [2] bash script can transform Sudoku format files into `lpsudoku` environments. It can process files in the so called one line 81 format² (option `-e` (default)) and in simple sudoku format (option `-s`)

Usage: `createlpsudoku [options] [-o output] -i input`

It expects an input file with the option `-i`. You can specify an output file with the option `-o`. Otherwise it writes to `stdout`. Furthermore, the following options are possible:

- `-w` write Windows line endings (CR/LF) to file
- `-v` prints version number
- `-h` prints help

3.25.3.2 lpsmag

With the `lpsmag` [30] bash script you can half automatically produce a Sudoku magazine using the `lpsudoku` environment and the `createlpsudoku` bash script.

Usage: `lpsmag configfile`

The script needs an installed `QQwing` [32] and a config file for defining the magazine's contents:

```
page p1 easy
page p2 easy
startpuzzles
typesetpage p1
typesetpage p2
startsolutions
typesetsolpage p1 p2 last
```

This config file will be sourced into the `lpsmag` bash script and contains calls of `lpsmag` functions. Make sure, that the config file has UNIX line endings (LF). For a detailed documentation I refer to the following [wiki](#) [30] entry.

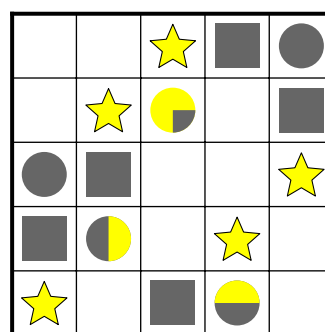
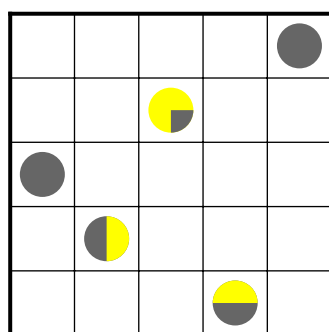
After running `lpsmag` you will find a `lpsmag.tex` in your working directory. Just run `pdflatex lpsmag.tex` twice and you finally get for example this [lpsmag.pdf](#).

²processing of several sudokus in 81 format (one in each line) is possible

3.26 Sun and Moon

Enter exactly one star and one dark cloud in each row and each column of the grid, so that the planets are illuminated as specified. The stars shine horizontally or vertically arbitrarily far, but not through a planet or a dark cloud.

3.26.1 Example



```
\begin{center}
\begin{sunandmoon}
\framepuzzle
\setrow{5}{{},{},{},{},{},\Moon}
\setrow{4}{{},{},{},\MoonTL}
\setrow{3}{{\Moon}}
\setrow{2}{{},{},\MoonR}
\setrow{1}{{},{},{},\MoonT}
\end{sunandmoon}
\hspace{1.5cm}
\begin{sunandmoon}
\framepuzzle
\setrow{5}{{},{},{},\Star,\Cloud,\Moon}
\setrow{4}{{},{},\Star,\MoonTL,{},\Cloud}
\setrow{3}{{\Moon},\Cloud,{},{},\Star}
\setrow{2}{{\Cloud},\MoonR,{},\Star}
\setrow{1}{{\Star},{},\Cloud,\MoonT}
\end{sunandmoon}
\end{center}
```

3.26.2 Options

`rows [5]` defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.1cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.1cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

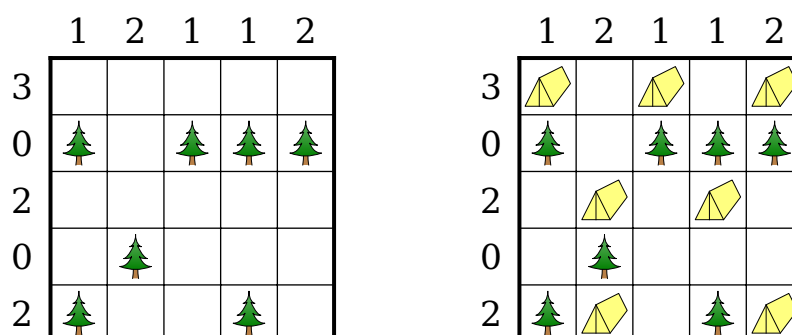
counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.27 Tents and Trees

Draw tents in the grid. Next to each tree, a tent must be entered in a horizontally or vertically adjacent cell, which is associated with this tree. The numbers next to the grid indicate the quantity of tents in each row or column. No tent can stand directly next to another one, not even diagonally.

3.27.1 Example



```
\begin{center}
\begin{tentsandtrees}
\framepuzzle
```

```

\tentH{1,2,1,1,2}
\tentV{2,0,2,0,3}
\setrow{4}{\Tree, {}, \Tree, \Tree, \Tree}
\setrow{2}{{}, \Tree}
\setrow{1}{\Tree, {}, {}, \Tree}
\end{tentsandtrees}
\hspace{1.5cm}
\begin{tentsandtrees}
\framepuzzle
\tentH{1,2,1,1,2}
\tentV{2,0,2,0,3}
\setrow{5}{\Tent, {}, \Tent, {}, \Tent, }
\setrow{4}{\Tree, {}, \Tree, \Tree, \Tree}
\setrow{3}{{}, \Tent, {}, \Tent}
\setrow{2}{{}, \Tree}
\setrow{1}{\Tree, \Tent, {}, \Tree, \Tent}
\end{tentsandtrees}
\end{center}

```

3.27.2 Options

rows [5] defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.9cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.9cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

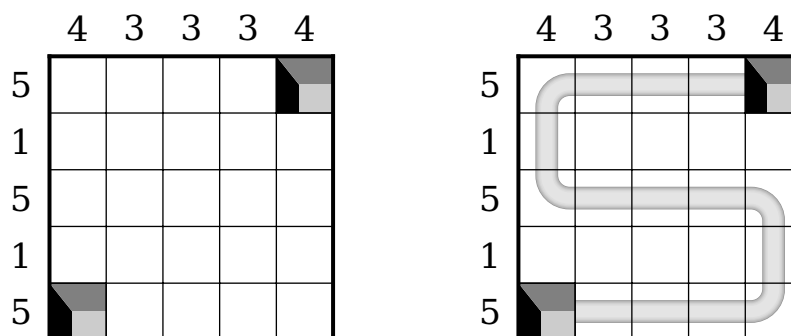
counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.

3.28 Tunnel

Determine the course of the tube. Draw the only possible connection. from the beginning to the end. The numbers indicate how many tube segments (including portals) are present in the corresponding rows and columns. The tube is one cell wide, and does not cross or touch itself!

3.28.1 Example



```
\begin{center}
\begin{tunnel}
\framepuzzle
\tunnelH{4,3,3,3,4}
\tunnelV{5,1,5,1,5}
\portal{1}{1}
\portal{5}{5}
\end{tunnel}
\hspace{1.5cm}
\begin{tunnel}
\framepuzzle
\tunnelH{4,3,3,3,4}
\tunnelV{5,1,5,1,5}
\portal{1}{1}
\portal{5}{5}
\tube{\tikzpath{1}{1}{6,6,6,6,8,8,4,4,4,4,8,8,6,6,6,6}}
\end{tunnel}
\end{center}
```

3.28.2 Options

`rows [5]` defines the number of rows in the grid.

columns [5] specifies the number of columns in the grid

width [5.9cm] sets the width of the minipage, in which the grid is typeset.

scale [1] scales the size of the grid in the minipage.

fontsize [Large] specifies the size of the numbers next to the grid. Here, the usual L^AT_EX sizes are used. Possible values: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

title [] sets the title of a puzzle.

titleindent [0cm] defines the indent of the title.

titlewidth [5.9cm] specifies the width of the box the title is set in.

bgcolor [] sets the background color of the grid.

counterstyle [none] defines the counter style. Predefined styles: none, left, right

cvoffset [-23pt] sets the vertical offset of the counters in the margin.



You can download application examples and their solutions from the [project page](#). The puzzles are originally licensed under .

4 Implementation

4.1 logicpuzzle.sty

```

1 \<package>
2 \NeedsTeXFormat{LaTeX2e}%
3 \ProvidesPackage{logicpuzzle}[2014/06/15 v2.5 logicpuzzle.sty%
4      - Josef Kleber (C) 2013-2014]%

```

4.1.1 Package initialization

First, we load the packages needed by logicpuzzle.sty and the TikZ libraries we need.

```

5 \RequirePackage{xkeyval}%
6 \RequirePackage{ifthen}%
7 \RequirePackage{ragged2e}%
8 \RequirePackage{marginnote}%
9 \RequirePackage{tikz}%
10 \usetikzlibrary{decorations.pathmorphing,decorations.pathreplacing,%
11      calc,shapes.geometric}%

```

We define a set of PGF layers for placing material on them and their order.


```

12 \pgfdeclarelayer{LPdump}%
13 \pgfdeclarelayer{LPbgcolor}%
14 \pgfdeclarelayer{LPbackgroundtwo}%
15 \pgfdeclarelayer{LPbackground}%
16 \pgfdeclarelayer{LPforeground}%
17 \pgfdeclarelayer{LPforegroundtwo}%
18 \pgfsetlayers{LPdump,LPbgcolor,LPbackgroundtwo,LPbackground,main,%
19               LPforeground,LPforegroundtwo}%
20 %

```

We also need some counters and lengths

```

21 \newcounter{LP@rows}%
22 \newcounter{LP@columns}%
23 \newcounter{LP@counter@unique}%
24 \newcounter{LP@counti}%
25 \newcounter{LP@countii}%
26 \newcounter{LP@countiii}%
27 \newcounter{LP@whiledo@i}%
28 \newcounter{LP@whiledo@ii}%
29 \newcounter{LP@count@ig@i}%
30 \newcounter{LP@count@ig@ii}%
31 \newcounter{LP@count@ig@iii}%
32 \newcounter{LP@count@ig@iv}%
33 \newcounter{LP@puzzlecounter}%
34 \setcounter{LP@puzzlecounter}{1}%
35 %
36 \newlength{LP@length}%

```

We define generic macros for puzzle options and add some defaults. Furthermore we define some macros, we will use all over the package.

```

37 \gdef\LP@rows{}%
38 \gdef\LP@columns{}%
39 \gdef\LP@scale{1}%
40 \gdef\LP@color{black}%
41 \gdef\LP@bgcolor{}%
42 \gdef\LP@fontsize\Large%
43 \gdef\LP@cwoffset{0pt}%
44 %
45 \newcommand*\LP@counterstyle{none}%
46 \newcommand*\LP@titleformat{}%
47 \gdef\LP@env@prefix{}%
48 \gdef\LP@package{}%
49 \def\LP@normallines{0.5pt}%
50 \def\LP@thicklines{1.5pt}%
51 \def\LP@grid@linestyle{}%
52 \def\LP@draw@opacity{1}%
53 \def\LP@rel@tikzpath{}%
54 \def\LP@tracks@scale{.3}%
55 \def\LP@fw@linestyle{-}%
56 \newcommand*\LP@Pfive{.5}%

```

4.1.2 Defining options

`\LP@define@key` We define a generic command for the definition of puzzle options with both global (for the `\puzzlesetup` commands) and local scope for the optional argument of the puzzles!

```
\LP@define@key{<puzzle prefix>}{<puzzle>}{<option>}{<default>}
```

```
57 \newcommand*\LP@define@key[4]%
58 {%
59   \expandafter\gdef\csname#1@#3\endcsname{#4}%
60   \define@key{#2.sty}{#3}{#4}%
61   {%
62     \expandafter\gdef\csname#1@#3\endcsname{##1}%
63   }%
64   \define@key{#2}{#3}%
65   {%
66     \expandafter\def\csname#1@#3\endcsname{##1}%
67   }%
68 }%
```

`\LP@define@choicekey@fontsize` We do the same for the more complicated choicekey for the fontsize option.

```
\LP@define@choicekey@fontsize{<puzzle prefix>}{<puzzle>}{<default>}
```

```
69 \newcommand*\LP@define@choicekey@fontsize[3]%
70 {%
71   \expandafter\gdef\csname#1@fontsize\endcsname{\Large}%
72   \define@choicekey*{#2.sty}{fontsize}{\LP@dck@fontsize\nr}{%
73     tiny,scriptsize,footnotesize,small,normalsize,%
74     large,Large,LARGE,huge,Huge}{#3}%
75   {%
76     \ifcase\nr\relax%
77       \expandafter\gdef\csname#1@fontsize\endcsname{\tiny}%
78     \or%
79       \expandafter\gdef\csname#1@fontsize\endcsname{\scriptsize}%
80     \or%
81       \expandafter\gdef\csname#1@fontsize\endcsname{\footnotesize}%
82     \or%
83       \expandafter\gdef\csname#1@fontsize\endcsname{\small}%
84     \or%
85       \expandafter\gdef\csname#1@fontsize\endcsname{\normalsize}%
86     \or%
87       \expandafter\gdef\csname#1@fontsize\endcsname{\large}%
88     \or%
89       \expandafter\gdef\csname#1@fontsize\endcsname{\Large}%
90     \or%
91       \expandafter\gdef\csname#1@fontsize\endcsname{\LARGE}%
92     \or%
93       \expandafter\gdef\csname#1@fontsize\endcsname{\huge}%
94     \or%
95       \expandafter\gdef\csname#1@fontsize\endcsname{\Huge}%
96   }%
97 }
```

```

96   \fi%
97 }%
98 \define@choicekey*{#2}{fontsize}[\LP@dck@fontsize\nr]{%
99     tiny,scriptsize,footnotesize,small,normalsize,%
100     large,Large,LARGE,huge,Huge}[#3]%
101 {%
102   \ifcase\nr\relax%
103     \expandafter\def\csname#1@fontsize\endcsname{\tiny}%
104   \or%
105     \expandafter\def\csname#1@fontsize\endcsname{\scriptsize}%
106   \or%
107     \expandafter\def\csname#1@fontsize\endcsname{\footnotesize}%
108   \or%
109     \expandafter\def\csname#1@fontsize\endcsname{\small}%
110   \or%
111     \expandafter\def\csname#1@fontsize\endcsname{\normalsize}%
112   \or%
113     \expandafter\def\csname#1@fontsize\endcsname{\large}%
114   \or%
115     \expandafter\def\csname#1@fontsize\endcsname{\Large}%
116   \or%
117     \expandafter\def\csname#1@fontsize\endcsname{\LARGE}%
118   \or%
119     \expandafter\def\csname#1@fontsize\endcsname{\huge}%
120   \or%
121     \expandafter\def\csname#1@fontsize\endcsname{\Huge}%
122   \fi%
123 }%
124 }%

```

4.1.3 Defining colors

`\LP@definecolor` We define a command for defining (rgb) colors. For other color models, use `xcolor`'s `\definecolor` command.

```
\LP@definecolor{<name>}{<rgb color>}
```

```

125 \newcommand*\LP@definecolor[2]%
126 {%
127   \definecolor{#1}{rgb}{#2}%
128 }%

```

Predefined colors:

```

129 \LP@definecolor{LP@c@i}{.9,.9,.98}%
130 \LP@definecolor{LP@c@ii}{.688,.932,.932}%
131 \LP@definecolor{LP@c@iii}{.88,1,1}%
132 \LP@definecolor{LP@c@iv}{1,1,.88}%
133 \LP@definecolor{LP@c@v}{1,.855,.725}%
134 \LP@definecolor{LP@c@vi}{.498,1,0}%
135 \LP@definecolor{LP@c@vii}{.53,.808,.98}%

```

```

136 \LP@definecolor{LP@c@viii}{.196,.804,.196}%
137 \LP@definecolor{LP@c@ix}{1,.96,.932}%
138 \LP@definecolor{LP@c@griddots}{.25,.51,.33}%
139 \LP@definecolor{LP@c@cross}{.25,.51,.33}%
140 \LP@definecolor{LP@c@mirror}{.8,.8,.8}%
141 \LP@definecolor{LP@c@tunnel}{.7,.7,.7}%
142 \LP@definecolor{LP@c@tunnel@i}{.8,.8,.8}%
143 \LP@definecolor{LP@c@tunnel@ii}{.5,.5,.5}%
144 \LP@definecolor{LP@c@bridge}{1,0,0}%
145 \colorlet{LP@c@track}{black!80}%

```

4.1.4 Drawing grids

`\LP@drawgrid` We define a command for drawing the standard grid used by all puzzles. In general, this should be a grid with a step of 1cm and thin lines with size (1,1) to (columns + 1, rows + 1). You can influence the grid by redefining the `\LP@grid@linestyle` (default: solid – maybe you want dashed) and `\LP@draw@opacity` (0 (transparent) – 1 (opaque)). They should be changed only within groups, like puzzle environment definitions.

`\LP@drawgrid{<xmin>}{<ymin>}{<xmax>}{<ymax>}{<step>}`

```

146 \newcommand*\LP@drawgrid[5]%
147 {%
148   \setcounter{LP@counti}{#3}% max column
149   \setcounter{LP@countii}{#4}% max row
150   \stepcounter{LP@counti}%
151   \stepcounter{LP@countii}%
152   \draw[step=#5,line width=\LP@normallines,\LP@grid@linestyle,%
153     draw opacity=\LP@draw@opacity]%
154     (#1,#2) grid (\value{LP@counti},\value{LP@countii});%
155 }%

```

4.1.5 Drawing the puzzle background

`\LP@drawbackground` For drawing the puzzle background, we simply draw a rectangle with the size of the puzzle on the `LPbgcolor` layer and fill it with `{<bgcolor>}`.

`\LP@drawbackground{<xmin>}{<ymin>}{<xmax>}{<ymax>}{<bgcolor>}`

```

156 \newcommand*\LP@drawbackground[5]%
157 {%

```

Of course, we only draw a background, if `{<bgcolor>}` is not empty!

```

158   \ifthenelse{\equal{#5}{}}{%
159     }{% no bgcolor
160     {%
161       \setcounter{LP@counti}{#3}% max column
162       \setcounter{LP@countii}{#4}% max row

```

```

163 \stepcounter{LP@counti}%
164 \stepcounter{LP@countii}%
165 \begin{pgfonlayer}{LPbgcolor}%
166 \fill[color=#5] (#1,#2) rectangle%
167 (\value{LP@counti},\value{LP@countii});%
168 \end{pgfonlayer}%
169 }%
170 }%

```

4.1.6 In the grid

`\LP@ingrid` With this macro, we can check if the specified column and row is within the puzzle borders. Otherwise we issue an error message. This macro can be used by higher level commands, which try to place something in the grid.

```
\LP@ingrid{<column>}{<row>}{<max column>}{<max row>}{<package>}
```

```

171 \newcommand*\LP@ingrid[5]%
172 {%

```

First, we define some counters to store the arguments.

```

173 \setcounter{LP@count@ig@i}{#1}% column
174 \setcounter{LP@count@ig@ii}{#2}% row
175 \setcounter{LP@count@ig@iii}{#3}% max column
176 \setcounter{LP@count@ig@iv}{#4}% max row

```

Then, we can check, if the specified coordinate is within the borders of the puzzle.

```

177 \ifnum\value{LP@count@ig@i}<1%
178 \PackageError{#5}%
179 {element outside of the grid}%
180 {You tried to set an element at (#1,#2),\MessageBreak%
181 which is outside the grid (1,1) .. (#3,#4)}%
182 \fi%
183 \ifnum\value{LP@count@ig@ii}<1%
184 \PackageError{#5}%
185 {element outside of the grid}%
186 {You tried to set an element at (#1,#2),\MessageBreak%
187 which is outside the grid (1,1) .. (#3,#4)}%
188 \fi%
189 \ifnum\value{LP@count@ig@i}>\value{LP@count@ig@iii}%
190 \PackageError{#5}%
191 {element outside of the grid}%
192 {You tried to set an element at (#1,#2),\MessageBreak%
193 which is outside the grid (1,1) .. (#3,#4)}%
194 \fi%
195 \ifnum\value{LP@count@ig@ii}>\value{LP@count@ig@iv}%
196 \PackageError{#5}%
197 {element outside of the grid}%
198 {You tried to set an element at (#1,#2),\MessageBreak%

```

```

199         which is outside the grid (1,1) .. (#3,#4)}%
200     \fi%
201 }%

```

`\setrule` For the kendoku and killersudoku environments, we need a command to place a calculation rule in the top left corner of the specified cell.

```
\setrule{<column>}{<row>}{<rule>}
```

```

202 \newcommand*\setrule[3]%
203 {%

```

First, we copy the scale and bgcolor values from the current environment.

```

204 \LP@set@LP@scale{\LP@env@prefix}%
205 \LP@set@LP@bgcolor{\LP@env@prefix}%

```

If no bgcolor is specified, we use white for drawing our helper rectangle. We also step our unique node counter, we need for referencing nodes between different layers.

```

206 \ifthenelse{\equal{\LP@bgcolor}{}}{%
207   {\gdef\LP@sr@bgcolor{white}}%
208   {\gdef\LP@sr@bgcolor{\LP@bgcolor}}%
209   \stepcounter{LP@counter@unique}%

```

First, we draw a (invisible) helper rectangle on the LPdump layer (behind the LPbgcolor layer) in the node `A_theLP@counter@unique`

```

210 \begin{pgfonlayer}{LPdump}%
211   \node [shape=rectangle,inner sep=0pt] (A\_theLP@counter@unique)%
212     at (#1\LP@Pfive,#2\LP@Pfive)%
213     {\tikz\draw[scale=\LP@scale,color=\LP@sr@bgcolor]%
214       (0.08,0.1) rectangle (0.92,0.9)};%
215 \end{pgfonlayer}%

```

Then, we can place a visible node on the main layer in the top left corner of the invisible helper rectangle with the rule (size: $\text{\tiny} \times \text{scale}$).

```

216 \node [shape=rectangle,inner sep=0pt,anchor=north west,%
217       scale=\LP@scale,font=\tiny] at%
218       (A\_theLP@counter@unique.north west) {$#3$};%
219 }%

```

`\LP@G@setcellcontent` Here, we define a generic macro for placing material into nodes placed in the bottom left corner of the grid cell. You can use the options `vcenter` and `hcenter` in the optional argument [`<options>`] of the macro to center the content horizontally and/or vertically.

```
\LP@G@setcellcontent[<options>]{<column>}{<row>}{<content>}
```

```

220 \define@key{LP@G@setcellcontent}{vcenter}{\LP@Pfive}%
221 {%

```

```

222 \def\LP@scc@v{\LP@Pfive}%
223 }%
224 \define@key{LP@G@setcellcontent}{hcenter}{\LP@Pfive}%
225 {%
226 \def\LP@scc@h{\LP@Pfive}%
227 }%
228 %
229 \newcommand*\LP@G@setcellcontent[4][ ]%
230 {%
231 \def\LP@scc@h{ }%
232 \def\LP@scc@v{ }%
233 \setkeys{LP@G@setcellcontent}{#1}%
234 \node at (#2\LP@scc@h,#3\LP@scc@v){#4};%
235 }%

```

`\LP@setcellcontent` For compatibility, we still provide the old `\LP@setcellcontentXY` macros.

```

236 \newcommand*\LP@setcellcontent[3]%
237 {%
238 \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{#3}%
239 }%

```

`\LP@setcellcontentC`

```

240 \newcommand*\LP@setcellcontentC[3]%
241 {%
242 \LP@G@setcellcontent{#1}{#2}{#3}%
243 }%

```

`\LP@setcellcontentVE`

```

244 \newcommand*\LP@setcellcontentVE[3]%
245 {%
246 \LP@G@setcellcontent[vcenter]{#1}{#2}{#3}%
247 }%

```

`\LP@setcellcontentHE`

```

248 \newcommand*\LP@setcellcontentHE[3]%
249 {%
250 \LP@G@setcellcontent[hcenter]{#1}{#2}{#3}%
251 }%

```

`\LP@setrowcontents` We want also be able to set the contents of a complete row.

`\LP@setrowcontents{<csv list>}{<column>}{<row>}`

```

252 \newcommand*\LP@setrowcontents[3]%
253 {%

```

We copy fontsize from the current environment, if we want to typeset numbers or letters.

```

254 \LP@set@LP@fontsize{\LP@env@prefix}%
255 \setcounter{LP@counti}{#2}%
256 \setcounter{LP@countii}{#3}%

```

We loop through the list and create a centered node in cell (column,row). Finally, we step the column counter.

```

257 \foreach \LP@element in {#1}%
258 {%
259   \LP@G@setcellcontent[vcenter,hcenter]%
260   {\arabic{LP@counti}}{\arabic{LP@countii}}%
261   {\LP@fontsize\LP@element}%
262   \stepcounter{LP@counti}%
263 }%
264 }%

```

`\LP@setcolumncontents` Of course, we want to do the same for columns.

```
\LP@setcolumncontents{\<csv list>}{\<column>}{\<row>}
```

```

265 \newcommand*\LP@setcolumncontents[3]%
266 {%
267   \LP@set@LP@fontsize{\LP@env@prefix}%
268   \setcounter{LP@counti}{#2}%
269   \setcounter{LP@countii}{#3}%
270   \foreach \LP@element in {#1}%
271   {%
272     \LP@G@setcellcontent[vcenter,hcenter]%
273     {\arabic{LP@counti}}{\arabic{LP@countii}}%
274     {\LP@fontsize\LP@element}%
275     \stepcounter{LP@countii}%
276   }%
277 }%

```

`\LP@setrowcontents@edge` For environments like `laserbeam`, we need to typeset row contents on the left border of the cell, instead of centered. Therefore, we only use `vcenter`.

```
\LP@setrowcontents@edge{\<csv list>}{\<column>}{\<row>}
```

```

278 \newcommand*\LP@setrowcontents@edge[3]%
279 {%
280   \LP@set@LP@fontsize{\LP@env@prefix}%
281   \setcounter{LP@counti}{#2}%
282   \setcounter{LP@countii}{#3}%
283   \foreach \LP@element in {#1}%
284   {%
285     \LP@G@setcellcontent[vcenter]%
286     {\arabic{LP@counti}}{\arabic{LP@countii}}%
287     {\LP@fontsize\LP@element}%
288     \stepcounter{LP@counti}%
289   }%
290 }%

```


`\LP@setcolumncontents@edge` Furthermore, we need the ability to typeset a column on the bottom border of a cell (hcenter).

`\LP@setcolumncontents@edge{\langle csv list \rangle}{\langle column \rangle}{\langle row \rangle}`

```

291 \newcommand*\LP@setcolumncontents@edge[3]%
292 {%
293   \LP@set@LP@fontsize{\LP@env@prefix}%
294   \setcounter{LP@counti}{#2}%
295   \setcounter{LP@countii}{#3}%
296   \foreach \LP@element in {#1}%
297   {%
298     \LP@G@setcellcontent[hcenter]%
299     {\arabic{LP@counti}}{\arabic{LP@countii}}%
300     {\LP@fontsize\LP@element}%
301     \stepcounter{LP@countii}%
302   }%
303 }%
```

`\setrow` The user command for typesetting row contents.

`\setrow{\langle row \rangle}{\langle csv list \rangle}`

```

304 \newcommand*\setrow[2]%
305 {%
```

First, we copy `scale`, which might be necessary for graphical objects, like `\Mine`. Then, we hand over the arguments to the generic macro for typesetting row contents and start with column 1.

```

306   \LP@set@LP@scale{\LP@env@prefix}%
307   \LP@setrowcontents{#2}{1}{#1}%
308 }%
```

`\setcolumn` Again, we do the same for columns.

`\setcolumn{\langle column \rangle}{\langle csv list \rangle}`

```

309 \newcommand*\setcolumn[2]%
310 {%
311   \LP@set@LP@scale{\LP@env@prefix}%
312   \LP@setcolumncontents{#2}{#1}{1}%
313 }%
```

`\setcell` We need to set numbers, letters or a graphical object into a central node in grid cell (column,row)

`\setcell{\langle column \rangle}{\langle row \rangle}{\langle element \rangle}`

```

314 \newcommand*\setcell[3]%
315 {%
```

First, we copy scale, fontsize, rows and columns from the current environment.

```
316 \LP@set@LP@scale{\LP@env@prefix}%
317 \LP@set@LP@fontsize{\LP@env@prefix}%
318 \LP@set@LP@rows{\LP@env@prefix}%
319 \LP@set@LP@columns{\LP@env@prefix}%
```

Then, we test if $(\langle column \rangle, \langle row \rangle)$ is within the borders of the puzzle. Finally, we typeset $\langle element \rangle$ into a central node.

```
320 \LP@ingrid{\#1}{\#2}{\LP@columns}{\LP@rows}{\LP@package}%
321 \LP@G@setcellcontent[hcenter,vcenter]{\#1}{\#2}{\LP@fontsize\#3}%
322 }%
```

`\setcells` We also want to typeset the same element into several grid cells. Therefore, we use a $\langle csv list \rangle$ with the format: $\langle column/row, ... \rangle$

`\setcells` $\langle csv list \rangle \langle element \rangle$

```
323 \newcommand*\setcells[2]%
324 {%
325   \LP@set@LP@scale{\LP@env@prefix}%
326   \LP@set@LP@fontsize{\LP@env@prefix}%
327   \LP@set@LP@rows{\LP@env@prefix}%
328   \LP@set@LP@columns{\LP@env@prefix}%
329   \foreach \LP@sc@column/\LP@sc@row in {#1}%
330   {%
331     \LP@ingrid{\LP@sc@column}{\LP@sc@row}%
332     {\LP@columns}{\LP@rows}{\LP@package}%
333     \LP@G@setcellcontent[hcenter,vcenter]%
334     {\LP@sc@column}{\LP@sc@row}{\LP@fontsize\#2}%
335   };%
336 }%
```

`\setbigcell` For the slitherlink environment, we need to typeset a huge (2×2) grid cell.

`\setbigcell` $[\langle fontsize \rangle] \langle column \rangle \langle row \rangle \langle element \rangle$

```
337 \newcommand*\setbigcell[4][Huge]%
338 {%
```

First, we copy scale and bgcolor. If bgcolor is not specified, we assume bgcolor is white!

```
339 \LP@set@LP@scale{\LP@env@prefix}%
340 \LP@set@LP@bgcolor{\LP@env@prefix}%
341 \ifthenelse{equal{\LP@bgcolor}{}}{%
342   {\gdef\LP@sbc@bgcolor{white}}%
343   {\gdef\LP@sbc@bgcolor{\LP@bgcolor}}%
}
```

The center of (2×2) cell is the bottom left corner of $(column + 1, row + 1)$

```

344 \setcounter{LP@counti}{#2}%
345 \setcounter{LP@countii}{#3}%
346 \stepcounter{LP@counti}%
347 \stepcounter{LP@countii}%

```

First, we 'clear' (2×2) area with `bgcolor` and respect the grid line style.

```

348 \draw[line width=\LP@normallines,fill=\LP@sbc@bgcolor,%
349       \LP@grid@linestyle]%
350       (#2,#3) rectangle ++(2,2);%

```

As the grid dots were drawn on the `LPforeground` layer, we have to 'overpaint' the center dot on the `LPforegroundtwo` layer with `bgcolor`. Finally, we can typeset `{\langle element \rangle}`.

```

351 \begin{pgfonlayer}{LPforegroundtwo}%
352   \fill[color=\LP@sbc@bgcolor]%
353     (\arabic{LP@counti},\arabic{LP@countii})%
354     circle [radius=3.5pt*\LP@scale];%
355   \node at (\arabic{LP@counti},\arabic{LP@countii})%
356     {\csname#1\endcsname#4};%
357 \end{pgfonlayer}%
358 }%

```

`\setcolorrow` For the `hitori` environment, we need to typeset row contents with numbers associated to background colors.

```

\setcolorrow{\langle row \rangle}{\langle csv list \rangle}

359 \newcommand*\setcolorrow[2]%
360 {%

```

We start at column 1 and loop through `{\langle csv list \rangle}`

```

361 \setcounter{LP@counti}{1}%
362 \setcounter{LP@countii}{#1}%
363 \foreach \LP@element in {#2}%
364 {%

```

If the list element is 0, we fill the cell black on the `LPbackground` layer.

```

365   \ifthenelse{\equal{\LP@element}{0}}%
366   {%
367     \gdef\LP@HT@color{black}%
368     \begin{puzzlebackground}%
369       \fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}%
370     \end{puzzlebackground}%
371   }%
372   {%

```

Otherwise, we fill the cell with predefined color `LP@c@romannumber` on the `LPbackground` layer and typeset the list element into the grid cell.

```

373     \expandafter\gdef\expandafter\LP@HT@color%
374         {LP@cc@\romannumeral\LP@element}%
375     \begin{puzzlebackground}%
376         \fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}%
377     \end{puzzlebackground}%
378     \setcell{\arabic{LP@counti}}{\arabic{LP@countii}}{\LP@element}%
379 }%
380 \stepcounter{LP@counti}%
381 }%
382 }%

```

`\setcolorcolumn` Again, the same for columns.

```

383 \newcommand*\setcolorcolumn[2]%
384 {%
385     \setcounter{LP@counti}{#1}%
386     \setcounter{LP@countii}{1}%
387     \foreach \LP@element in {#2}%
388     {%
389         \ifthenelse{\equal{\LP@element}{0}}%
390         {%
391             \gdef\LP@HT@color{black}%
392             \begin{puzzlebackground}%
393                 \fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}%
394             \end{puzzlebackground}%
395         }%
396         {%
397             \expandafter\gdef\expandafter\LP@HT@color%
398                 {LP@cc@\romannumeral\LP@element}%
399             \begin{puzzlebackground}%
400                 \fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}%
401             \end{puzzlebackground}%
402             \setcell{\arabic{LP@counti}}{\arabic{LP@countii}}{\LP@element}%
403         }%
404         \stepcounter{LP@countii}%
405     }%
406 }%

```

`\fillcell` Sometimes, we need to ‘blacken’ a grid cell.

`\fillcell{<column>}{<row>}`

```

407 \newcommand*\fillcell[2]%
408 {%
409     \LP@set@LP@scale{\LP@env@prefix}%
410     \LP@set@LP@color{\LP@env@prefix}%
411     \LP@set@LP@rows{\LP@env@prefix}%
412     \LP@set@LP@columns{\LP@env@prefix}%

```

After a border check, we typeset a black block (`\LP@Block`) into the grid cell.

```

413     \LP@ingrid{#1}{#2}{\LP@columns}{\LP@rows}{\LP@package}%

```

```

414 \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{\LP@Block}%
415 }%

```

`\fillrow` We also want to allow the filling of (parts of a) row.

```

\fillrow{<row>}{<csv list>}

416 \newcommand*\fillrow[2]%
417 {%
418   \setcounter{LP@counti}{1}%
419   \setcounter{LP@countii}{#1}%
420   \foreach \LP@element in {#2}%
421   {%

```

We loop through the list and if element is 1, we fill this grid cell.

```

422     \ifthenelse{\equal{\LP@element}{1}}%
423     {\fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}}%
424     }%
425     \stepcounter{LP@counti}%
426   }%
427 }%

```

`\fillcolumn` Again the same, for columns!

```

\fillcolumn{<column>}{<csv list>}

428 %
429 \newcommand*\fillcolumn[2]%
430 {%
431   \setcounter{LP@counti}{#1}%
432   \setcounter{LP@countii}{1}%
433   \foreach \LP@element in {#2}%
434   {%
435     \ifthenelse{\equal{\LP@element}{1}}%
436     {\fillcell{\arabic{LP@counti}}{\arabic{LP@countii}}}%
437     }%
438     \stepcounter{LP@countii}%
439   }%
440 }%

```

`\framearea` Sometimes, we need to frame a specified area.

```

\framearea{<color>}{<TikZ path>}

441 \newcommand*\framearea[2]%
442 {%
443   \draw[line width=\LP@thicklines,color=#1] #2;%
444 }%

```

`\fillarea` Sometimes, we even want to fill the area.

```

\fillarea{<color>}{<TikZ path>}

```

```

445 \newcommand*\fillarea[2]%
446 {%
447   \draw[line width=\LP@thicklines,fill=#1] #2;%
448 }%

```

`\colorarea` In some cases it might be better just to fill the area without drawing a frame.

`\colorarea{<color>}{<TikZ path>}`

```

449 \newcommand*\colorarea[2]%
450 {%
451   \fill[color=#1] #2 ;%
452 }%

```

`\tikzpath` Using a `{<TikZ path>}` can be cumbersome. `\tikzpath` construct a path starting at the bottom left corner of grid cell (column,row). If want to start in the center of the cell, redefine `\LP@rel@tikzpath` to `.5` inside a group! Inside `{<csv list of relative moves>}`, you can specify relative movements from one grid cell to the next based on num pad (4 → one cell right, 2 → one cell down and 9 → one cell right up).

`\tikzpath{<column>}{<row>}{<csv list of relative movement>}`

```

453 \newcommand*\tikzpath[3]%
454 {%

```

starting point

```

455   (#1\LP@rel@tikzpath,#2\LP@rel@tikzpath)%

```

We loop through the list and add a relative path segment based on the direction indicator.

```

456   \foreach \LP@direction in {#3}%
457   {%
458     \ifnum\LP@direction=1%
459       --++(-1,-1)%
460     \fi%
461     \ifnum\LP@direction=2%
462       --++(0,-1)%
463     \fi%
464     \ifnum\LP@direction=3%
465       --++(1,-1)%
466     \fi%
467     \ifnum\LP@direction=4%
468       --++(-1,0)%
469     \fi%
470     \ifnum\LP@direction=6%
471       --++(1,0)%
472     \fi%
473     \ifnum\LP@direction=7%
474       --++(-1,1)%

```

```

475 \fi%
476 \ifnum\LP@direction=8%
477 --++(0,1)%
478 \fi%
479 \ifnum\LP@direction=9%
480 --++(1,1)%
481 \fi%
482 };%
483 }%

```

`\xtikzpath` `\xtikzpath` is based on `\tikzpath` with a slightly different format in `{(csv list of relative movements)}`. It allows pairs of direction/length, e.g. (6/2) means two cells right.

`\xtikzpath{<column>}{<row>}{<csv list of relative movements>}`

```

484 \newcommand*\xtikzpath[3]%
485 {%
486 (#1\LP@rel@tikzpath,#2\LP@rel@tikzpath)%
487 \foreach \LP@dir/\LP@plength in {#3}%
488 {%
489 \ifnum\LP@dir=1%
490 --++(-\LP@plength,-\LP@plength)%
491 \fi%
492 \ifnum\LP@dir=2%
493 --++(0,-\LP@plength)%
494 \fi%
495 \ifnum\LP@dir=3%
496 --++(\LP@plength,-\LP@plength)%
497 \fi%
498 \ifnum\LP@dir=4%
499 --++(-\LP@plength,0)%
500 \fi%
501 \ifnum\LP@dir=6%
502 --++(\LP@plength,0)%
503 \fi%
504 \ifnum\LP@dir=7%
505 --++(-\LP@plength,\LP@plength)%
506 \fi%
507 \ifnum\LP@dir=8%
508 --++(0,\LP@plength)%
509 \fi%
510 \ifnum\LP@dir=9%
511 --++(\LP@plength,\LP@plength)%
512 \fi%
513 };%
514 }%

```

`\filldiagonals` For some puzzles we need colored diagonals indicating that also the diagonals are relevant, not just rows and columns.

`\filldiagonals[⟨color⟩]`

```
515 \newcommand*\filldiagonals[1][yellow!20]%
516 {%
```

We copy and get scale, \$rows+1\$ and \$columns+1\$,

```
517 \LP@set@LP@columns{\LP@env@prefix}%
518 \LP@set@LP@rows{\LP@env@prefix}%
519 \LP@set@LP@scale{\LP@env@prefix}%
520 \setcounter{LP@counti}{\LP@columns}%
521 \setcounter{LP@countii}{\LP@rows}%
522 \stepcounter{LP@counti}%
523 \stepcounter{LP@countii}%
```

We only color the diagonals, if the puzzle is quadratic. Otherwise, coloring diagonals doesn't make sense and we issue an error.

```
524 \ifnum\value{LP@counti}=\value{LP@countii}%
```

We color the diagonals on the LPbackground layer and redefine `\LP@color` (for `\LP@Block`) inside a group with local scope.

```
525 \begin{puzzlebackground}%
526 \def\LP@color{#1}%
```

We use two counters running from (1 .. rows) and (rows .. 1) and color the cells on the diagonals in a loop.

```
527 \setcounter{LP@whiledo@i}{1}%
528 \setcounter{LP@whiledo@ii}{\LP@rows}%
529 \whiledo{\value{LP@whiledo@i}<\value{LP@counti}}%
530 {%
531 \LP@G@setcellcontent[hcenter,vcenter]%
532 {\arabic{LP@whiledo@i}}{\arabic{LP@whiledo@ii}}{\LP@Block}%
533 \LP@G@setcellcontent[hcenter,vcenter]%
534 {\arabic{LP@whiledo@i}}{\arabic{LP@whiledo@ii}}{\LP@Block}%
535 \stepcounter{LP@whiledo@i}%
536 \addtocounter{LP@whiledo@ii}{-1}%
537 }%
538 \end{puzzlebackground}%
539 \else%
540 \PackageError{\LP@package}%
541 {non quadratic grid (\filldiagonals)}%
542 {You tried to fill diagonals,\MessageBreak in a non quadratic%
543 grid (1,1) .. (\arabic{LP@counti},\arabic{LP@countii})}%
544 \MessageBreak doesn't make sense!}%
545 \fi%
546 }%
```

`\framepuzzle` We might want to frame the puzzle with a thicker line.

`\framepuzzle[⟨color⟩]`


```

547 \newcommand*\framepuzzle[1][black]%
548 {%
549   \LP@set@LP@columns{\LP@env@prefix}%
550   \LP@set@LP@rows{\LP@env@prefix}%
551   \setcounter{LP@counti}{\LP@columns}%
552   \setcounter{LP@countii}{\LP@rows}%
553   \stepcounter{LP@counti}%
554   \stepcounter{LP@countii}%

```

We copied rows and columns to get the top right corner of the puzzle. Now, we can draw the thicker lines.

```

555   \draw[line width=\LP@thicklines,color=#1]%
556     (1,1)--(1,\arabic{LP@countii});%
557   \draw[line width=\LP@thicklines,color=#1]%
558     (1,\arabic{LP@countii})--%
559     (\arabic{LP@counti},\arabic{LP@countii});%
560   \draw[line width=\LP@thicklines,color=#1]%
561     (\arabic{LP@counti},\arabic{LP@countii})--%
562     (\arabic{LP@counti},1);%
563   \draw[line width=\LP@thicklines,color=#1]%
564     (\arabic{LP@counti},1)--(1,1);%
565 }%

```

`puzzlebackground` Sometimes, we want to move material to the LPbackground layer.

```

566 \newenvironment{puzzlebackground}%
567 {%
568   \begin{pgfonlayer}{LPbackground}%
569 }%
570 {%
571   \end{pgfonlayer}%
572 }%

```

`puzzleforeground` Sometimes, we want to move material to the LPforeground layer.

```

573 \newenvironment{puzzleforeground}%
574 {%
575   \begin{pgfonlayer}{LPforeground}%
576 }%
577 {%
578   \end{pgfonlayer}%
579 }%

```

4.1.7 Around the grid

For most puzzles, we need to put numbers below, above, to the right or to the left of the puzzle.

`\LP@bottomrow` Put numbers below the puzzle.

`\LP@bottomrow{<csv list>}`

```
580 \newcommand*\LP@bottomrow[1]%
581 {%
582   \LP@setrowcontents{#1}{1}{0}%
583 }%
```

`\LP@bottomrow@edge` Put numbers below the puzzle, but on the edge.

`\LP@bottomrow@edge{<csv list>}`

```
584 \newcommand*\LP@bottomrow@edge[1]%
585 {%
586   \LP@setrowcontents@edge{#1}{1}{0}%
587 }%
```

`\LP@leftcolumn` Put numbers left to the puzzle.

`\LP@leftcolumn{<csv list>}`

```
588 \newcommand*\LP@leftcolumn[1]%
589 {%
590   \LP@setcolumncontents{#1}{0}{1}%
591 }%
```

`\LP@toprow` Put number on top of the puzzle.

`\LP@toprow{<csv list>}`

```
592 \newcommand*\LP@toprow[1]%
593 {%
594   \LP@setrowcontents{#1}{1}{\theLP@rows}%
595 }%
```

`\LP@rightcolumn` Put numbers right to the puzzle.

`\LP@rightcolumn{<csv list>}`

```
596 \newcommand*\LP@rightcolumn[1]%
597 {%
598   \LP@setcolumncontents{#1}{\theLP@columns}{1}%
599 }%
```

`\LP@rightcolumn@edge` Put numbers right next to the puzzle, but on the edge.

`\LP@rightcolumn@edge{<csv list>}`

```
600 \newcommand*\LP@rightcolumn@edge[1]%
601 {%
602   \LP@setcolumncontents@edge{#1}{\theLP@columns}{1}%
603 }%
```

4.1.8 Presentation

`\titleformat` You can redefine the format of the puzzle at any time.

```
604 \newcommand*\titleformat[1]%
605 {%
606   \renewcommand*\LP@titleformat{#1}%
607 }%
```

We define the default title format:

```
608 \titleformat{\centering\Large\color{blue}}%
```

`\puzzlecounter` User command to get the current puzzle counter.

```
609 \newcommand*\puzzlecounter{\theLP@puzzlecounter}%
```

`\setpuzzlecounter` User command to reset the puzzle counter, e.g. before the solution section.

```
610 \newcommand*\setpuzzlecounter[1]%
611 {%
612   \setcounter{LP@puzzlecounter}{#1}%
613 }%
```

`\LP@drawcounter` Maybe we want to draw puzzle counter, or not (`counterstyle=none`).

```
\LP@drawcounter{\counterstyle}
```

```
614 \newcommand*\LP@drawcounter[1]%
615 {%
```

We copy `cvoffset` and execute the macro we defined for the respective `counterstyle`.

```
616   \LP@set@LP@cvoffset{\LP@env@prefix}%
617   \csname LP@cs@#1\endcsname%
618 }%
```

`\definecounterstyle` Maybe a user want to define his/her own counter style.

```
\definecounterstyle{\counterstyle}{\definition}
```

```
619 \newcommand*\definecounterstyle[2]%
620 {%
```

We define a counterstyle macro and store its definition.

```
621   \expandafter\gdef\csname LP@cs@#1\endcsname{#2}%
622 }%
```

We define three predefined counterstyles none, left and right.

```

623 \definecounterstyle{none}{}%
624 \definecounterstyle{left}{}%
625 \beginngroup\reversemarginpar%
626   \marginnote{%
627     \tikz\node[shape=rectangle,fill=yellow!40,inner sep=7pt,draw,%
628       rounded corners=3pt,thick]%
629       {\Huge\puzzlecounter}};][\LP@cwoffset]%
630 \endgroup%
631 }%
632 \definecounterstyle{right}{}%
633 \marginnote{%
634   \tikz\node[shape=rectangle,fill=yellow!40,inner sep=7pt,draw,%
635     rounded corners=3pt,thick]%
636     {\Huge\puzzlecounter}};%
637 }][\LP@cwoffset]%
638 }%

```

4.1.9 Misc macros

`\LP@Block` We define a TikZ picture with the size of a grid cell ($1\text{cm} \times 1\text{cm}$) and fill it with color.

```

639 \newcommand*\LP@Block{%
640   \tikz[scale=\LP@scale]%
641     \draw[line width=\LP@normallines,fill=\LP@color]%
642       (0,0) rectangle (1,1);}%

```

`\LP@Line` A generic command for drawing lines with options `double`, `color` and `linewidth`.

`\LP@Line[<options>]{<TikZ path>}`

```

643 \define@key{\LP@Line}{double}[double]%
644 {%
645   \def\LP@Line@double{double}%
646 }%
647 %
648 \define@key{\LP@Line}{color}[\LP@c@bridge]%
649 {%
650   \def\LP@Line@linecolor{#1}%
651 }%
652 %
653 \define@key{\LP@Line}{linewidth}[1mm]%
654 {%
655   \def\LP@Line@linewidth{#1}%
656 }%
657 %
658 \newcommand*\LP@Line[2][]%
659 {%
660   \def\LP@Line@double{}%
661   \def\LP@Line@linecolor{\LP@c@bridge}%

```

```

662 \def\LP@Line@linewidth{1mm}%
663 \setkeys{LP@Line}{#1}%

```

By default we draw a single 1mm wide line with color LP@c@bridge. But we can change that with [*options*].

```

664 \LP@set@LP@scale{\LP@env@prefix}%

```

Then we can draw a single or double line based on the defined options or defaults.

```

665 \begin{pgfonlayer}{LPbackgroundtwo}%
666 \ifthenelse{\equal{\LP@Line@double}{double}}{%
667   {\draw[double,double distance=\LP@Line@linewidth*\LP@scale,%
668     color=\LP@Line@linecolor,%
669     line width=\LP@Line@linewidth*\LP@scale] #2;}%
670   {\draw[color=\LP@Line@linecolor,%
671     line width=\LP@Line@linewidth*\LP@scale] #2;}%
672 \end{pgfonlayer}%
673 }%

```

\LP@set@LP@scale

```

674 \newcommand*\LP@set@LP@scale[1]%
675 {%
676   \expandafter\xdef\expandafter\LP@scale{\csname #1@scale\endcsname}%
677 }%

```

\LP@set@LP@color

```

678 \newcommand*\LP@set@LP@color[1]%
679 {%
680   \expandafter\xdef\expandafter\LP@color{\csname #1@color\endcsname}%
681 }%

```

\LP@set@LP@rows

```

682 \newcommand*\LP@set@LP@rows[1]%
683 {%
684   \expandafter\xdef\expandafter\LP@rows{\csname #1@rows\endcsname}%
685 }%

```

\LP@set@LP@columns

```

686 \newcommand*\LP@set@LP@columns[1]%
687 {%
688   \expandafter\xdef\expandafter\LP@columns{\csname #1@columns\endcsname}%
689 }%

```

\LP@set@LP@cwoffset

```

690 \newcommand*\LP@set@LP@cwoffset[1]%
691 {%
692   \expandafter\xdef\expandafter\LP@cwoffset{\csname #1@cwoffset\endcsname}%
693 }%

```

\LP@set@LP@bgcolor

```

694 \newcommand*\LP@set@LP@bgcolor[1]%
695 {%
696   \expandafter\xdef\expandafter\LP@bgcolor{\csname #1@bgcolor\endcsname}%
697 }%

```

\LP@set@LP@extracells

```

698 \newcommand*\LP@set@LP@extracells[1]%
699 {%
700   \expandafter\xdef\expandafter\LP@extracells{\csname #1@extracells\endcsname}%
701 }%

```

\LP@set@LP@fontsize

```

702 \newcommand*\LP@set@LP@fontsize[1]%
703 {%
704   can't expand \Large
705   \expandafter\gdef\expandafter\LP@fontsize{\csname #1@fontsize\endcsname}%
706 }%

```

\LP@set@env@prefix

```

707 \newcommand*\LP@set@env@prefix[1]%
708 {%
709   \gdef\LP@env@prefix{#1}%
710 }%

```

\LP@set@package

```

711 \newcommand*\LP@set@package[1]%
712 {%
713   \gdef\LP@package{#1}%
714 }%

```

\setgridlinestyle

```

715 \newcommand*\setgridlinestyle[1]%
716 {%
717   \def\LP@grid@linestyle{#1}%
718 }%

```

\setnormallinewidth

```

719 \newcommand*\setnormallinewidth[1]%

```

```

720 {%
721   \def\LP@normallines{#1}%
722 }%

\setthicklinewidth

723 \newcommand*\setthicklinewidth[1]%
724 {%
725   \def\LP@thicklines{#1}%
726 }%

\puzzlestrut

727 \newcommand*\puzzlestrut%
728 {%
729   \LP@set@LP@rows{\LP@env@prefix}%
730   \ifthenelse{\equal{\LP@package}{nonogram}}%
731   {%
732     \LP@set@LP@extracells{\LP@env@prefix}%
733     \setcounter{LP@counti}{\LP@rows}% max row
734     \stepcounter{LP@counti}%
735     \addtocounter{LP@counti}{\LP@NG@extracells}%
736     \draw[line width=\LP@thicklines, transparent]%
737       (1,1) -- (1,\arabic{LP@counti});%
738   }{%
739 }%

\setTikZpreset

740 \newcommand*\setTikZpreset[1]%
741 {%
742   \tikzset{%
743     LPpreset/.style={#1}%
744   }%
745 }%
746 %
747 \setTikZpreset{line cap=rect,line join=round}%

logicpuzzle

748 \newcommand*\LP@LP@init@prefix{LP@LP}%
749 \newcommand*\LP@LP@init@package{logicpuzzle}%
750 %
751 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
752 {rows}{5}%
753 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
754 {columns}{5}%
755 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
756 {scale}{1}%
757 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
758 {counterstyle}{none}%

```

```

759 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
760 {color}{}%
761 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
762 {bgcolor}{}%
763 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
764 {width}{5.1cm}%
765 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
766 {cvmoffset}{-23pt}%
767 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
768 {title}{}%
769 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
770 {titleindent}{0cm}%
771 \LP@define@key{\LP@LP@init@prefix}{\LP@LP@init@package}%
772 {titlewidth}{5.1cm}%
773 \LP@define@choicekey@fontsize{\LP@LP@init@prefix}%
774 {\LP@LP@init@package}{Large}%
775 %
776 \ExecuteOptionsX{rows,columns,width,fontsize,scale,bgcolor%
777 ,cvmoffset,counterstyle,title,titleindent,titlewidth}%
778 %
779 \ProcessOptionsX\relax%
780 %
781 \let\logicpuzzleccl\setcell%
782 %
783 \newcommand*\logicpuzzlesetup[1]%
784 {%
785   \setkeys{logicpuzzle.sty}{#1}%
786 }%
787 %
788 \newenvironment{logicpuzzle}[1][]%
789 {%
790   \setkeys{logicpuzzle}{#1}%
791   \LP@set@package{logicpuzzle}%
792   \LP@set@env@prefix{LP@LP}%
793   \setcounter{LP@rows}{\LP@LP@rows}%
794   \setcounter{LP@columns}{\LP@LP@columns}%
795   \stepcounter{LP@rows}%
796   \stepcounter{LP@columns}%
797   \begin{minipage}[t]{\LP@LP@width}%
798     \ifthenelse{\equal{\LP@LP@title}{}}{%
799       {\par\enspace\par}% empty
800       {\enspace\par\noindent\hspace{\LP@LP@titleindent}%
801         \parbox{\LP@LP@titlewidth}{\strut\LP@titleformat\LP@LP@title}%
802         \vspace{3mm}\par}%
803     \begin{tikzpicture}[scale=\LP@LP@scale]%
804       \LP@drawbackground{1}{1}{\LP@LP@columns}{\LP@LP@rows}%
805       {\LP@LP@bgcolor}%
806       \LP@drawgrid{1}{1}{\LP@LP@columns}{\LP@LP@rows}{1cm}%
807     }%
808   }%
809   \end{tikzpicture}%

```



```

810 \LP@drawcounter{\LP@LP@counterstyle}%
811 \stepcounter{LP@puzzlecounter}%
812 \end{minipage}%
813 }%
814 %
815 \RequirePackage{lpenv}%

816 \</package>

```

4.2 lpenv.sty

Here's just a flat copy of `lpenv.sty`! The code for the puzzles are more or less simple copy & paste. Take a look at section 1 for a detailed explanation of code needed for a puzzle environment.

```

817 \<{*lpenv}

818 %
819 % battleship environment and options
820 %
821 \newcommand*\LP@BS@init@prefix{\LP@BS}%
822 \newcommand*\LP@BS@init@package{battleship}%
823 %
824 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{rows}{5}%
825 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{columns}{5}%
826 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{shipcolor}{green}%
827 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{scale}{1}%

828 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{counterstyle}{none}%

829 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{bgcolor}{}%
830 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{width}{6cm}%

831 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{cvoffset}{-23pt}%

832 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{title}{}%
833 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{titleindent}{0.75cm}%
834 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{titlewidth}{5.15cm}%
835 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{sbindent}{0.75cm}%
836 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{sbwidth}{5.15cm}%
837 \LP@define@key{\LP@BS@init@prefix}{\LP@BS@init@package}{sbshipscale}{1}%
838 \LP@define@choicekey@fontsize{\LP@BS@init@prefix}{\LP@BS@init@package}{{Large}}%
839 %
840 \gdef\LP@BS@shipbox{}%
841 \tikzstyle{island} = [fill=yellow!30,draw, decorate, decoration={random steps,segment len
842 %
843 \newcommand*\Ship{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor] (0,0) c
844 \newcommand*\ShipC{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor] (0,0)-
845 \newcommand*\ShipT{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor] (2,1)-

```

```

846 \newcommand*\ShipB{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor](2,1)--
847 \newcommand*\ShipL{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor](1,2)--
848 \newcommand*\ShipR{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill=\LP@BS@shipcolor](1,2)--
849 \newcommand*\Island{\tikz[scale=\LP@BS@scale]\draw[scale=.36,island] (0,0) rectangle (2,2)
850 \newcommand*\Water{\tikz[scale=\LP@BS@scale]\draw[scale=.36,fill,blue!40] (1,1) circle (0.5)
851 % versions for \shipbox without second scale
852 \newcommand*\@Ship{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor] (0,0) circle (1);}%
853 \newcommand*\@ShipC{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor] (0,0)--(0,2)--(2,2)--(2,0)--(0,2)--(0,0)}%
854 \newcommand*\@ShipT{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor](2,1)--(2,0)--(0,0)--(0,1)--(0,2)--(0,0)}%
855 \newcommand*\@ShipB{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor](2,1)--(2,2)--(0,2)--(0,1)--(0,2)--(0,0)}%
856 \newcommand*\@ShipL{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor](1,2)--(2,2)--(2,0)--(1,0)--(1,2)--(1,0)}%
857 \newcommand*\@ShipR{\tikz\draw[scale=.144,fill=\LP@BS@shipcolor](1,2)--(0,2)--(0,0)--(1,0)--(1,2)--(1,0)}%
858 %
859 \newcommand*\LP@BS@printship[1]%
860 {%
861   \ifthenelse{\equal{#1}{1}}{\scalebox{\LP@BS@sbshipscale}{\@Ship}\space\allowbreak}{}%
862   \ifthenelse{\equal{#1}{2}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipR}\space\allowbreak}{}%
863   \ifthenelse{\equal{#1}{3}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipR}\space\allowbreak}{}%
864   \ifthenelse{\equal{#1}{4}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipR}\space\allowbreak}{}%
865   \ifthenelse{\equal{#1}{5}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC}\space\allowbreak}{}%
866   \ifthenelse{\equal{#1}{6}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC,\@ShipR}\space\allowbreak}{}%
867   \ifthenelse{\equal{#1}{7}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC,\@ShipC}\space\allowbreak}{}%
868   \ifthenelse{\equal{#1}{8}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC,\@ShipC,\@ShipR}\space\allowbreak}{}%
869   \ifthenelse{\equal{#1}{9}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC,\@ShipC,\@ShipC}\space\allowbreak}{}%
870   \ifthenelse{\equal{#1}{10}}{\scalebox{\LP@BS@sbshipscale}{\@ShipL,\@ShipC,\@ShipC,\@ShipC,\@ShipC,\@ShipC,\@ShipR}\space\allowbreak}{}%
871 }%
872 %
873 \newcommand*\LP@BS@printshipbox[1]%
874 {%
875   % no {#1}! to avoid expansion
876   \foreach \LP@element in #1%
877   {%
878     \LP@BS@printship{\LP@element}%
879   }%
880 }%
881 %
882 \newcommand*\shipH[1]%
883 {%
884   \LP@setrowcontents{#1}{1}{\value{LP@rows}}%
885 }%
886 %
887 \newcommand*\shipV[1]%
888 {%
889   \LP@setcolumncontents{#1}{0}{1}%
890 }%
891 %
892 \newcommand*\placesegment[3]%
893 {%
894   \LP@ingrid{#1}{#2}{\LP@BS@columns}{\LP@BS@rows}{battleship}%
895   \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{#3}%
896 }%

```

```

896 %
897 \let\ship\placesegment%

898 \newcommand*\placeisland[2]%
899 {%
900   \LP@ingrid{#1}{#2}{\LP@BS@columns}{\LP@BS@rows}{battleship}%
901   \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{\Island}%
902 }%

903 \newcommand*\placewater[2]%
904 {%
905   \LP@ingrid{#1}{#2}{\LP@BS@columns}{\LP@BS@rows}{battleship}%
906   \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{\Water}%
907 }%

908 %
909 \newcommand*\shipbox[2][ ]%
910 {%
911   \setkeys{shipbox}{#1}%
912   \gdef\LP@BS@shipbox{#2}%
913 }%

914 \newcommand*\placeship[4]%
915 {%
916   \setcounter{LP@counti}{#4}% length
917   \ifnum\value{LP@counti}<1%
918     \PackageError{battleship}%
919       {ship length < 1}%
920       {The length of your ship should be at least 1}%
921   \fi%
922   \ifnum\value{LP@counti}>10%
923     \PackageError{battleship}%
924       {ship length > 10}%
925       {The supported max length of ships is 10!}%
926   \fi%
927   \ifnum\value{LP@counti}=1%
928     \placesegment{#2}{#3}{\Ship}%
929   \else%
930     \setcounter{LP@whiledo@i}{\value{LP@counti}}%
931     \addtocounter{LP@whiledo@i}{-2}% length of middle ship
932     \ifthenelse{\equal{#1}{V}}%
933     {%
934       \placesegment{#2}{#3}{\ShipB}%
935       \setcounter{LP@countii}{#3}%
936       \whiledo{\value{LP@whiledo@i}>0}%
937       {%
938         \addtocounter{LP@whiledo@i}{-1}%
939         \stepcounter{LP@countii}%
940         \placesegment{#2}{\theLP@countii}{\ShipC}%
941       }%
942     \stepcounter{LP@countii}%
943     \placesegment{#2}{\theLP@countii}{\ShipT}%

```

```

944 }%
945 {%
946 \ifthenelse{\equal{#1}{H}}{%
947 {%
948 \placesegment{#2}{#3}{\ShipL}%
949 \setcounter{LP@counti}{#2}%
950 \whiledo{\value{LP@whiledo@i}>0}%
951 {%
952 \addtocounter{LP@whiledo@i}{-1}%
953 \stepcounter{LP@counti}%
954 \placesegment{\theLP@counti}{#3}{\ShipC}%
955 }%
956 \stepcounter{LP@counti}%
957 \placesegment{\theLP@counti}{#3}{\ShipR}%
958 }%
959 {\PackageError{battleship}%
960 {invalid direction (H/V)}%
961 {You can place your ship only\MessageBreak%
962 horizontally (H) or vertically (V)!}%
963 }%
964 }%
965 \fi%
966 }%
967 %

968 \newcommand*\battleshipsetup[1]%
969 {%
970 \setkeys{battleship.sty}{#1}%
971 }%
972 %

973 \newcommand{\classicgame}[1]%
974 {%
975 \begin{center}%
976 \begin{battleship}[rows=10,columns=10,width=6.5cm,title=Me,sbindent=0.65cm,titleinden
977 \shipV{J,I,H,G,F,E,D,C,B,A}%
978 \shipH{1,2,3,4,5,6,7,8,9,10}%
979 \shipbox{#1}%
980 \end{battleship}%
981 \hspace{1cm}%
982 \begin{battleship}[rows=10,columns=10,width=6.5cm,title=Enemy,sbindent=0.65cm,titleinden
983 \shipV{J,I,H,G,F,E,D,C,B,A}%
984 \shipH{1,2,3,4,5,6,7,8,9,10}%
985 \end{battleship}%
986 \end{center}%
987 \par\vspace{1cm}%
988 \begin{center}%
989 \begin{battleship}[rows=10,columns=10,width=6.5cm,title=Me,sbindent=0.65cm,titleinden
990 \shipV{J,I,H,G,F,E,D,C,B,A}%
991 \shipH{1,2,3,4,5,6,7,8,9,10}%
992 \shipbox{#1}%

```

```

993   \end{battleship}%
994   \hspace{1cm}%
995   \begin{battleship}[rows=10,columns=10,width=6.5cm,title=Enemy,sbindent=0.65cm,titlein
996     \shipV{J,I,H,G,F,E,D,C,B,A}%
997     \shipH{1,2,3,4,5,6,7,8,9,10}%
998   \end{battleship}%
999   \end{center}%
1000 }%

```

battleship

```

1001 \newenvironment{battleship}[1][]%
1002 {%
1003   \setkeys{battleship}{#1}%
1004   \LP@set@package{battleship}%
1005   \LP@set@env@prefix{LP@BS}%
1006   \shipbox{}% clear shipbox
1007   \setcounter{LP@rows}{\LP@BS@rows}%
1008   \setcounter{LP@columns}{\LP@BS@columns}%
1009   \stepcounter{LP@rows}%
1010   \stepcounter{LP@columns}%
1011   \begin{minipage}[t]{\LP@BS@width}%
1012     \ifthenelse{\equal{\LP@BS@title}{}}{%
1013       {\par\enspace\par}% empty
1014       {\enspace\par\noindent\hspace{\LP@BS@titleindent}\parbox{\LP@BS@titlewidth}{\strut\LP
1015     \begin{tikzpicture}[LP@preset,scale=\LP@BS@scale]%
1016       \LP@drawbackground{1}{1}{\LP@BS@columns}{\LP@BS@rows}{\LP@BS@bgcolor}%
1017       \LP@drawgrid{1}{1}{\LP@BS@columns}{\LP@BS@rows}{1cm}%
1018   }%
1019   {%
1020     \end{tikzpicture}%
1021     \LP@drawcounter{\LP@BS@counterstyle}%
1022     \par\hbadness=10000\medskip\noindent\hspace{\LP@BS@sbindent}\begin{minipage}{\LP@BS@s
1023     \stepcounter{LP@puzzlecounter}%
1024   \end{minipage}%
1025   }%

1026 %
1027 % bokkusu environment and options
1028 %
1029 \newcommand*\LP@BK@init@prefix{LP@BK}%
1030 \newcommand*\LP@BK@init@package{bokkusu}%
1031 %
1032 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{rows}{5}%
1033 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{columns}{5}%
1034 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{scale}{1}%
1035 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{counterstyle}{none}%
1036 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{color}{black}%
1037 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{bgcolor}{}%
1038 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{width}{6.7cm}%
1039 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{cvoffset}{-38pt}%

```

```

1040 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{title}{}%
1041 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{titleindent}{0.75cm}%
1042 \LP@define@key{\LP@BK@init@prefix}{\LP@BK@init@package}{titlewidth}{5.85cm}%
1043 \LP@define@choicekey@fontsize{\LP@BK@init@prefix}{\LP@BK@init@package}{Large}%
1044 %
1045 \let\valueH\LP@bottomrow%
1046 \let\valueV\LP@leftcolumn%
1047 \let\sumH\LP@toprow%
1048 \let\sumV\LP@rightcolumn%
1049 %
1050 \newcommand*\bokkususetup[1]%
1051 {%
1052   \setkeys{bokkusu.sty}{#1}%
1053 }%

```

bokkusu

```

1054 \newenvironment{bokkusu}[1][]%
1055 {%
1056   \setkeys{bokkusu}{#1}%
1057   \LP@set@package{bokkusu}%
1058   \LP@set@env@prefix{\LP@BK}%
1059   \setcounter{LP@rows}{\LP@BK@rows}%
1060   \setcounter{LP@columns}{\LP@BK@columns}%
1061   \stepcounter{LP@rows}%
1062   \stepcounter{LP@columns}%
1063   \begin{minipage}[t]{\LP@BK@width}%
1064     \ifthenelse{\equal{\LP@BK@title}{}}{%
1065       {\par\enspace\par}% empty
1066       {\enspace\par\noindent\hspace{\LP@BK@titleindent}\parbox{\LP@BK@titlewidth}{\strut\LP@BK@title}%
1067     }{\begin{tikzpicture}[LP@preset,scale=\LP@BK@scale]%
1068       \LP@drawbackground{1}{1}{\LP@BK@columns}{\LP@BK@rows}{\LP@BK@bgcolor}%
1069       \LP@drawgrid{1}{1}{\LP@BK@columns}{\LP@BK@rows}{1cm}%
1070     }%
1071   }%
1072   \end{tikzpicture}%
1073   \LP@drawcounter{\LP@BK@counterstyle}%
1074   \stepcounter{LP@puzzlecounter}%
1075   \end{minipage}%
1076 }%

1077 %
1078 % bridges environment and options
1079 %
1080 \newcommand*\LP@BG@init@prefix{\LP@BG}%
1081 \newcommand*\LP@BG@init@package{bridges}%
1082 %
1083 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{rows}{5}%
1084 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{columns}{5}%
1085 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{scale}{1}%
1086 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{counterstyle}{none}%
1087 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{grid}{dashed}%

```

```

1088 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{bgcolor}{}%
1089 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{color}{green}%
1090 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{width}{6.1cm}%
1091 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{cvoffset}{-23pt}%
1092 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{title}{}%
1093 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{titleindent}{0cm}%
1094 \LP@define@key{\LP@BG@init@prefix}{\LP@BG@init@package}{titlewidth}{6.1cm}%
1095 \LP@define@choicekey@fontsize{\LP@BG@init@prefix}{\LP@BG@init@package}{Large}%
1096 %
1097 \let\bridgescell\setcell%
1098 %
1099 \newcommand*\bridgesetup[1]%
1100 {%
1101   \setkeys{bridges.sty}{#1}%
1102 }%

```

`\bridgesrow` For the bridges environment, we need a special row command for drawing the islands with the numbers of bridges.

`\bridgesrow{<row>}{<csv list>}`

```

1103 \newcommand*\bridgesrow[2]%
1104 {%
1105   \setcounter{LP@counti}{1}%
1106   \setcounter{LP@countii}{#1}%
1107   \LP@set@LP@fontsize{\LP@env@prefix}%
1108   \LP@set@LP@color{\LP@env@prefix}%
1109   \foreach \LP@element in {#2}%
1110   {%

```

We loop through the list and if element is not empty, we draw an island (circle filled with bgcolor) and typeset the number of bridges into to center of the circle.

```

1111     \ifthenelse{\equal{\LP@element}{}}{%
1112     }{%
1113     {%
1114       \draw[fill=\LP@color] (\arabic{LP@counti},\arabic{LP@countii})%
1115       circle (0.5cm);%
1116       \node[scale=\LP@scale,font=\LP@fontsize] at%
1117       (\arabic{LP@counti},\arabic{LP@countii}){\LP@element};%
1118     }%
1119     \stepcounter{LP@counti}%
1120   };%
1121 }%

```

`\bridgescolumn` And again, for columns!

```

1122 \newcommand*\bridgescolumn[2]%
1123 {%
1124   \setcounter{LP@counti}{#1}%
1125   \setcounter{LP@countii}{1}%

```

```

1126 \LP@set@LP@fontsize{\LP@env@prefix}%
1127 \LP@set@LP@color{\LP@env@prefix}%
1128 \foreach \LP@element in {#2}%
1129 {%
1130   \ifthenelse{\equal{\LP@element}{}}%
1131   {%
1132     {%
1133       \draw[fill=\LP@color] (\arabic{LP@counti},\arabic{LP@countii})%
1134       circle (0.5cm);%
1135       \node[scale=\LP@scale,font=\LP@fontsize]%
1136       at (\arabic{LP@counti},\arabic{LP@countii})%
1137       {\LP@element};%
1138     }%
1139     \stepcounter{LP@countii}%
1140   };%
1141 }%

```

`\bridge` For the `bridges` environment, we need to draw bridges. Keep in mind that you can influence the appearance of the bridge with the options `double`, `color` and `linewidth`.

```
\bridge[<options>]{<Tikz path>}
```

```

1142 \newcommand*\bridge[2][ ]%
1143 {%
1144   \LP@Line[#1]{#2}%
1145 }%

```

`bridges`

```

1146 \newenvironment{bridges}[1][ ]%
1147 {%
1148   \setkeys{bridges}{#1}%
1149   \LP@set@package{bridges}%
1150   \LP@set@env@prefix{LP@BG}%
1151   \setcounter{LP@rows}{\LP@BG@rows}%
1152   \setcounter{LP@columns}{\LP@BG@columns}%
1153   \stepcounter{LP@rows}%
1154   \stepcounter{LP@columns}%
1155   \begin{minipage}[t]{\LP@BG@width}%
1156     \ifthenelse{\equal{\LP@BG@title}{}}%
1157     {\par\enspace\par}% empty
1158     {\enspace\par\noindent\hspace{\LP@BG@titleindent}\parbox{\LP@BG@titlewidth}{\strut\LP
1159   \begin{tikzpicture}[LP@preset,scale=\LP@BG@scale]%
1160     \LP@drawbackground{1}{1}{\LP@BG@columns}{\LP@BG@rows}{\LP@BG@bgcolor}%
1161     \ifthenelse{\equal{\LP@BG@grid}{none}}%
1162     {}%
1163     {%
1164       \ifthenelse{\equal{\LP@BG@grid}{dashed}}%
1165       {%
1166         \setgridlinestyle{dashed}%
1167         \LP@drawgrid{1}{1}{\LP@BG@columns}{\LP@BG@rows}{1cm}%

```



```

1168     }%
1169     {%
1170     \LP@drawgrid{1}{1}{\LP@BG@columns}{\LP@BG@rows}{1cm}%
1171     }%
1172     }%
1173 }%
1174 {%
1175     \end{tikzpicture}%
1176     \LP@drawcounter{\LP@BG@counterstyle}%
1177     \stepcounter{\LP@puzzlecounter}%
1178     \end{minipage}%
1179 }%

1180 %
1181 % chaossudoku environment and options
1182 %
1183 \newcommand*\LP@CS@init@prefix{\LP@CS}%
1184 \newcommand*\LP@CS@init@package{chaossudoku}%
1185 %
1186 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{rows}{5}%
1187 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{columns}{5}%
1188 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{scale}{1}%
1189 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{counterstyle}{none}%
1190 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{bgcolor}{}%
1191 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{width}{5.1cm}%
1192 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{cwoffset}{-23pt}%
1193 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{title}{}%
1194 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{titleindent}{0cm}%
1195 \LP@define@key{\LP@CS@init@prefix}{\LP@CS@init@package}{titlewidth}{5.1cm}%
1196 \LP@define@choicekey@fontsize{\LP@CS@init@prefix}{\LP@CS@init@package}{Large}%
1197 %
1198 \let\chaossudokucell\setcell%
1199 %
1200 \newcommand*\chaossudoku@setup[1]%
1201 {%
1202     \setkeys{chaossudoku.sty}{#1}%
1203 }%

```

chaossudoku

```

1204 \newenvironment{chaossudoku}[1][ ]%
1205 {%
1206     \setkeys{chaossudoku}{#1}%
1207     \LP@set@package{chaossudoku}%
1208     \LP@set@env@prefix{\LP@CS}%
1209     \setcounter{\LP@rows}{\LP@CS@rows}%
1210     \setcounter{\LP@columns}{\LP@CS@columns}%
1211     \stepcounter{\LP@rows}%
1212     \stepcounter{\LP@columns}%
1213     \begin{minipage}[t]{\LP@CS@width}%
1214         \ifthenelse{\equal{\LP@CS@title}{}}{%
1215             {\par\enspace\par}% empty

```

```

1216     {\enspace\par\noindent\hspace{\LP@CS@titleindent}\parbox{\LP@CS@titlewidth}{\strut\LP
1217     \begin{tikzpicture}[LPpreset,scale=\LP@CS@scale]%
1218         \LP@drawbackground{1}{1}{\LP@CS@columns}{\LP@CS@rows}{\LP@CS@bgcolor}%
1219         \LP@drawgrid{1}{1}{\LP@CS@columns}{\LP@CS@rows}{1cm}%
1220     }%
1221     {%
1222         \end{tikzpicture}%
1223         \LP@drawcounter{\LP@CS@counterstyle}%
1224         \stepcounter{LP@puzzlecounter}%
1225     \end{minipage}%
1226 }%

1227 %
1228 % ddsudoku environment and options
1229 %
1230 \newcommand*\LP@DDS@init@prefix{\LP@DDS}%
1231 \newcommand*\LP@DDS@init@package{ddsudoku}%
1232 %
1233 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{rows}{5}%
1234 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{columns}{5}%
1235 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{scale}{1}%
1236 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{counterstyle}{none}%
1237 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{bgcolor}{}%
1238 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{width}{5.1cm}%
1239 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{cvmoffset}{-23pt}%
1240 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{title}{}%
1241 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{titleindent}{0cm}%
1242 \LP@define@key{\LP@DDS@init@prefix}{\LP@DDS@init@package}{titlewidth}{5.1cm}%
1243 \LP@define@choicekey@fontsize{\LP@DDS@init@prefix}{\LP@DDS@init@package}{Large}%
1244 %
1245 \let\ddsudokucell\setcell%
1246 %
1247 \newcommand*\ddsudoku[1]%
1248 {%
1249     \setkeys{ddsudoku}{#1}%
1250 }%

```

ddsudoku

```

1251 \newenvironment{ddsudoku}[1][]%
1252 {%
1253     \setkeys{ddsudoku}{#1}%
1254     \LP@set@package{ddsudoku}%
1255     \LP@set@env@prefix{\LP@DDS}%
1256     \setcounter{LP@rows}{\LP@DDS@rows}%
1257     \setcounter{LP@columns}{\LP@DDS@columns}%
1258     \stepcounter{LP@rows}%
1259     \stepcounter{LP@columns}%
1260     \begin{minipage}[t]{\LP@DDS@width}%
1261         \ifthenelse{\equal{\LP@DDS@title}{} }%
1262         {\par\enspace\par}% empty
1263         {\enspace\par\noindent\hspace{\LP@DDS@titleindent}\parbox{\LP@DDS@titlewidth}{\strut\

```

```

1264 \begin{tikzpicture}[LPpreset,scale=\LP@DDS@scale]%
1265 \LP@drawbackground{1}{1}{\LP@DDS@columns}{\LP@DDS@rows}{\LP@DDS@bgcolor}%
1266 \LP@drawgrid{1}{1}{\LP@DDS@columns}{\LP@DDS@rows}{1cm}%
1267 }%
1268 {%
1269 \end{tikzpicture}%
1270 \LP@drawcounter{\LP@DDS@counterstyle}%
1271 \stepcounter{LP@puzzlecounter}%
1272 \end{minipage}%
1273 }%

1274 %
1275 % fourwinds environment and options
1276 %
1277 \newcommand*\LP@FW@init@prefix{LP@FW}%
1278 \newcommand*\LP@FW@init@package{fourwinds}%
1279 %
1280 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{rows}{5}%
1281 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{columns}{5}%
1282 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{scale}{1}%
1283 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{counterstyle}{none}%
1284 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{bgcolor}{}%
1285 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{color}{blue}%
1286 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{width}{5.1cm}%
1287 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{cvoffset}{-23pt}%
1288 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{title}{}%
1289 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{titleindent}{0cm}%
1290 \LP@define@key{\LP@FW@init@prefix}{\LP@FW@init@package}{titlewidth}{5.1cm}%
1291 \LP@define@choicekey@fontsize{\LP@FW@init@prefix}{\LP@FW@init@package}{Large}%
1292 %
1293 \newcommand*\fourwindscell[4]%
1294 {%
1295 \LP@set@LP@fontsize{\LP@env@prefix}%
1296 \LP@set@LP@color{\LP@env@prefix}%
1297 \LP@set@LP@bgcolor{\LP@env@prefix}%
1298 \LP@set@LP@scale{\LP@env@prefix}%
1299 \ifthenelse{\equal{\LP@bgcolor}{}}{%
1300 {\gdef\LP@c@bg@fw{white}}%
1301 {\gdef\LP@c@bg@fw{\LP@bgcolor}}%
1302 \foreach \LP@fw@dir/\LP@fw@length in {#4}%
1303 {%
1304 \begin{pgfonlayer}{LPbackgroundtwo}%
1305 \def\LP@rel@tikzpath{.5}%
1306 \draw[\LP@fw@linestyle,color=\LP@color,line width=.1cm*\LP@scale,shorten >=-3mm*\LP
1307 \xtikzpath{#1}{#2}{\LP@fw@dir/\LP@fw@length};%
1308 \end{pgfonlayer}%
1309 };%
1310 \begin{puzzlebackground}%
1311 \node[fill=\LP@c@bg@fw,font=\LP@fontsize] at (#1.5,#2.5) {#3};%
1312 \end{puzzlebackground}%
1313 }%
1314 %

```

```

1315 \newcommand*\fourwindssetup[1]%
1316 {%
1317   \setkeys{fourwinds.sty}{#1}%
1318 }%

```

fourwinds

```

1319 \newenvironment{fourwinds}[1][1]%
1320 {%
1321   \setkeys{fourwinds}{#1}%
1322   \LP@set@package{fourwinds}%
1323   \LP@set@env@prefix{LP@FW}%
1324   \setcounter{LP@rows}{\LP@FW@rows}%
1325   \setcounter{LP@columns}{\LP@FW@columns}%
1326   \stepcounter{LP@rows}%
1327   \stepcounter{LP@columns}%
1328   \begin{minipage}[t]{\LP@FW@width}%
1329     \ifthenelse{\equal{\LP@FW@title}{}}{%
1330       {\par\enspace\par}% empty
1331       {\enspace\par\noindent\hspace{\LP@FW@titleindent}\parbox{\LP@FW@titlewidth}{\strut\LP
1332     \begin{tikzpicture}[LP@preset,scale=\LP@FW@scale]%
1333       \LP@drawbackground{1}{1}{\LP@FW@columns}{\LP@FW@rows}{\LP@FW@bgcolor}%
1334       \LP@drawgrid{1}{1}{\LP@FW@columns}{\LP@FW@rows}{1cm}%
1335     }%
1336   {%
1337     \end{tikzpicture}%
1338     \LP@drawcounter{\LP@FW@counterstyle}%
1339     \stepcounter{LP@puzzlecounter}%
1340   \end{minipage}%
1341 }%

1342 %
1343 % hakyuu environment and options
1344 %
1345 \newcommand*\LP@HY@init@prefix{LP@HY}%
1346 \newcommand*\LP@HY@init@package{hakyuu}%
1347 %
1348 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{rows}{5}%
1349 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{columns}{5}%
1350 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{scale}{1}%
1351 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{counterstyle}{none}%
1352 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{bgcolor}{}%
1353 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{width}{5.1cm}%
1354 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{cvoffset}{-23pt}%
1355 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{title}{}%
1356 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{titleindent}{0cm}%
1357 \LP@define@key{\LP@HY@init@prefix}{\LP@HY@init@package}{titlewidth}{5.1cm}%
1358 \LP@define@choicekey@fontsize{\LP@HY@init@prefix}{\LP@HY@init@package}{Large}%
1359 %
1360 \let\hakyuucell\setcell%
1361 %
1362 \newcommand*\hakyuussetup[1]%

```

```

1363 {%
1364   \setkeys{hakyuu.sty}{#1}%
1365 }%

```

hakyuu

```

1366 \newenvironment{hakyuu}[1][]%
1367 {%
1368   \setkeys{hakyuu}{#1}%
1369   \LP@set@package{hakyuu}%
1370   \LP@set@env@prefix{LP@HY}%
1371   \setcounter{LP@rows}{\LP@HY@rows}%
1372   \setcounter{LP@columns}{\LP@HY@columns}%
1373   \stepcounter{LP@rows}%
1374   \stepcounter{LP@columns}%
1375   \begin{minipage}[t]{\LP@HY@width}%
1376     \ifthenelse{\equal{\LP@HY@title}{}}{%
1377       {\par\enspace\par}% empty
1378       {\enspace\par\noindent\hspace{\LP@HY@titleindent}\parbox{\LP@HY@titlewidth}{\strut\LP
1379       \begin{tikzpicture}[LP@preset,scale=\LP@HY@scale]%
1380         \LP@drawbackground{1}{1}{\LP@HY@columns}{\LP@HY@rows}{\LP@HY@bgcolor}%
1381         \LP@drawgrid{1}{1}{\LP@HY@columns}{\LP@HY@rows}{1cm}%
1382       }%
1383     }%
1384     \end{tikzpicture}%
1385     \LP@drawcounter{\LP@HY@counterstyle}%
1386     \stepcounter{LP@puzzlecounter}%
1387   \end{minipage}%
1388 }%

1389 %
1390 % hitori environment and options
1391 %
1392 \newcommand*\LP@HT@init@prefix{LP@HT}%
1393 \newcommand*\LP@HT@init@package{hitori}%
1394 %
1395 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{rows}{5}%
1396 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{columns}{5}%
1397 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{scale}{1}%
1398 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{counterstyle}{none}%
1399 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{bgcolor}{}%
1400 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{width}{5.1cm}%
1401 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{cvoffset}{-23pt}%
1402 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{title}{}%
1403 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{titleindent}{0cm}%
1404 \LP@define@key{\LP@HT@init@prefix}{\LP@HT@init@package}{titlewidth}{5.1cm}%
1405 \LP@define@choicekey@fontsize{\LP@HT@init@prefix}{\LP@HT@init@package}{Large}%
1406 %
1407 \let\hitoricell\setcell%
1408 %
1409 \newcommand*\hitorisetup[1]%
1410 {%

```

```

1411 \setkeys{hitori.sty}{#1}%
1412 }%

```

hitori

```

1413 \newenvironment{hitori}[1][]%
1414 {%
1415   \setkeys{hitori}{#1}%
1416   \LP@set@package{hitori}%
1417   \LP@set@env@prefix{LP@HT}%
1418   \setcounter{LP@rows}{\LP@HT@rows}%
1419   \setcounter{LP@columns}{\LP@HT@columns}%
1420   \stepcounter{LP@rows}%
1421   \stepcounter{LP@columns}%
1422   \begin{minipage}[t]{\LP@HT@width}%
1423     \ifthenelse{\equal{\LP@HT@title}{}}{%
1424       {\par\enspace\par}% empty
1425       {\enspace\par\noindent\hspace{\LP@HT@titleindent}\parbox{\LP@HT@titlewidth}{\strut\LP@HT@title}%
1426       \begin{tikzpicture}[LP@preset,scale=\LP@HT@scale]%
1427         \LP@drawbackground{1}{1}{\LP@HT@columns}{\LP@HT@rows}{\LP@HT@bgcolor}%
1428         \LP@drawgrid{1}{1}{\LP@HT@columns}{\LP@HT@rows}{1cm}%
1429       }%
1430     }%
1431     \end{tikzpicture}%
1432     \LP@drawcounter{\LP@HT@counterstyle}%
1433     \stepcounter{LP@puzzlecounter}%
1434   \end{minipage}%
1435 }%

1436 %
1437 % kakuro environment and options
1438 %
1439 \newcommand*\LP@KKR@init@prefix{LP@KKR}%
1440 \newcommand*\LP@KKR@init@package{kakuro}%
1441 %
1442 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{rows}{5}%
1443 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{columns}{5}%
1444 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{scale}{1}%
1445 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{counterstyle}{none}%
1446 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{bgcolor}{}%
1447 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{color}{green}%
1448 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{width}{5.1cm}%
1449 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{cvmoffset}{-23pt}%
1450 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{title}{}%
1451 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{titleindent}{0cm}%
1452 \LP@define@key{\LP@KKR@init@prefix}{\LP@KKR@init@package}{titlewidth}{5.1cm}%
1453 \LP@define@choicekey@fontsize{\LP@KKR@init@prefix}{\LP@KKR@init@package}{Large}%
1454 %
1455 \newif\ifLP@KKR@solution\LP@KKR@solutionfalse%
1456 %
1457 \define@choicekey*{kakuro.sty}{solution}[\LP@KKR@solution\nr]{true,false}[true]%
1458 {%

```

```

1459 \ifcase\nr\relax%
1460   \renewcommand*\LP@KKR@solution{true}%
1461   \LP@KKR@solutiontrue%
1462 \or%
1463   \renewcommand*\LP@KKR@solution{false}%
1464   \LP@KKR@solutionfalse%
1465 \fi%
1466 }%
1467 %
1468 \define@choicekey*{kakuro}{solution}[\LP@KKR@solution\nr]{true,false}[true]%
1469 {%
1470   \ifcase\nr\relax%
1471     \renewcommand*\LP@KKR@solution{true}%
1472     \LP@KKR@solutiontrue%
1473   \or%
1474     \renewcommand*\LP@KKR@solution{false}%
1475     \LP@KKR@solutionfalse%
1476   \fi%
1477 }%
1478 %
1479 \let\kakurocell\setcell%
1480 %
1481 \newcommand*\kakurosetup[1]%
1482 {%
1483   \setkeys{kakuro.sty}{#1}%
1484 }%

```

`\LP@KKR@print@element` For the kakuro environment, we need to typeset numbers and special Kakuro cells (`\KKR`).

`\LP@KKR@print@element{<element>}`

```

1485 \def\LP@KKR@print@element#1%
1486 {%

```

First, we test if `{<element>}` is a number and typeset `{<element>}` into the grid cell, if we are in solution mode. Otherwise, we execute the `\KKR` command.

```

1487 \if!\ifnum9<1#1!\else_\fi%
1488   \ifLP@KKR@solution%
1489     \setcell{\arabic{LP@counti}}{\arabic{LP@countii}}{#1}%
1490   \else%
1491     \fi%
1492   \else%
1493     #1%
1494   \fi%
1495 }%

```

`\kakurorow` For the kakuro environment, we need special row and column commands, which can also handle the `\KKR` commands.

```

1496 \newcommand*\kakurorow[2]%

```

```

1497 {%
1498   \setcounter{LP@counti}{1}%
1499   \setcounter{LP@countii}{#1}%
1500   \foreach \LP@element in {#2}%
1501   {%
1502     \LP@KKR@print@element{\LP@element}%
1503     \stepcounter{LP@counti}%
1504   };%
1505 }%

```

\kakurocolumn

```

1506 \newcommand*\kakurocolumn[2]%
1507 {%
1508   \setcounter{LP@counti}{#1}%
1509   \setcounter{LP@countii}{1}%
1510   \foreach \LP@element in {#2}%
1511   {%
1512     \LP@KKR@print@element{\LP@element}%
1513     \stepcounter{LP@countii}%
1514   };%
1515 }%

```

\KKR For the kakuro environment, we need to draw complex kakuro cells. They consist of a diagonally divided cell with the sums of the cells below and right of the current cell. They also have a special background color.

\KKR{*vertical sum*}{*horizontal sum*}

```

1516 \newcommand*\KKR[2]%
1517 {%

```

First of all, we copy scale, bgcolor and color.

```

1518 \LP@set@LP@scale{\LP@env@prefix}%
1519 \LP@set@LP@bgcolor{\LP@env@prefix}%
1520 \LP@set@LP@color{\LP@env@prefix}%

```

If bgcolor is undefined, we assume white.

```

1521 \ifthenelse{\equal{\LP@bgcolor}{}}{%
1522   {\gdef\LP@sr@bgcolor{white}}%
1523   {\gdef\LP@sr@bgcolor{\LP@bgcolor}}%

```

To get unique node names, we step our unique counter.

```

1524 \stepcounter{LP@counter@unique}%

```

In a first step, we draw a rectangular helper node in bgcolor color on the LPdump layer.

```

1525 \begin{pgfonlayer}{LPdump}%
1526   \node [shape=rectangle,inner sep=0pt] (A_\theLP@counter@unique)%

```



```

1527      at (\arabic{LP@counti}\LP@Pfive,\arabic{LP@countii}\LP@Pfive)%
1528      {\tikz\draw[scale=\LP@scale,color=\LP@sr@bgcolor]%
1529      (0.08,0.1) rectangle (0.92,0.9);};%
1530 \end{pgfonlayer}%

```

In the second step, we fill the cell with color color and draw a diagonal line.

```

1531 \begin{puzzlebackground}%
1532   \fill[color=\LP@color]%
1533   (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(1,1);%
1534   \draw[line width=\LP@normallines]%
1535   (\arabic{LP@counti},\arabic{LP@countii}) -- ++(0,1) -- ++(1,-1);%
1536 \end{puzzlebackground}%

```

Finally, we use the corners of the helper node to place the sums.

```

1537 \node [shape=rectangle,inner sep=0pt,anchor=south west,%
1538       scale=\LP@scale,font=\small]%
1539       at (A_\theLP@counter@unique.south west) {\#1};%
1540 \node [shape=rectangle,inner sep=0pt,anchor=north east,%
1541       scale=\LP@scale,font=\small]%
1542       at (A_\theLP@counter@unique.north east) {\#2};%
1543 }%

```

\Black Sometimes, we need black cells.

```

1544 \newcommand*\Black%
1545 {%
1546   \LP@set@LP@scale{\LP@env@prefix}%
1547   \begin{puzzlebackground}%
1548     \fill[color=black]%
1549     (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(1,1);%
1550   \end{puzzlebackground}%
1551 }%

```

kakuro

```

1552 \newenvironment{kakuro}[1][]%
1553 {%
1554   \setkeys{kakuro}{#1}%
1555   \LP@set@package{kakuro}%
1556   \LP@set@env@prefix{LP@KKR}%
1557   \setcounter{LP@rows}{\LP@KKR@rows}%
1558   \setcounter{LP@columns}{\LP@KKR@columns}%
1559   \stepcounter{LP@rows}%
1560   \stepcounter{LP@columns}%
1561   \begin{minipage}[t]{\LP@KKR@width}%
1562     \ifthenelse{\equal{\LP@KKR@title}{}}%
1563     {\par\enspace\par}% empty
1564     {\enspace\par\noindent\hspace{\LP@KKR@titleindent}\parbox{\LP@KKR@titlewidth}{\strut\
1565     \begin{tikzpicture}[LP@preset,scale=\LP@KKR@scale]%
1566       \LP@drawbackground{1}{1}{\LP@KKR@columns}{\LP@KKR@rows}{\LP@KKR@bgcolor}%

```

```

1567      \LP@drawgrid{1}{1}{\LP@KKR@columns}{\LP@KKR@rows}{1cm}%
1568 }%
1569 {%
1570   \end{tikzpicture}%
1571   \LP@drawcounter{\LP@KKR@counterstyle}%
1572   \stepcounter{\LP@puzzlecounter}%
1573   \end{minipage}%
1574 }%

1575 %
1576 % kendoku environment and options
1577 %
1578 \newcommand*\LP@KD@init@prefix{\LP@KD}%
1579 \newcommand*\LP@KD@init@package{kendoku}%
1580 %
1581 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{rows}{5}%
1582 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{columns}{5}%
1583 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{scale}{1}%
1584 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{counterstyle}{none}%
1585 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{bgcolor}{}%
1586 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{width}{5.1cm}%
1587 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{cvoffset}{-23pt}%
1588 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{title}{}%
1589 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{titleindent}{0cm}%
1590 \LP@define@key{\LP@KD@init@prefix}{\LP@KD@init@package}{titlewidth}{5.1cm}%
1591 \LP@define@choicekey@fontsize{\LP@KD@init@prefix}{\LP@KD@init@package}{Large}%
1592 %
1593 \let\kendokucell\setcell%
1594 %
1595 \newcommand*\kendokusetup[1]%
1596 {%
1597   \setkeys{kendoku.sty}{#1}%
1598 }%

```

kendoku

```

1599 \newenvironment{kendoku}[1][]%
1600 {%
1601   \setkeys{kendoku}{#1}%
1602   \LP@set@package{kendoku}%
1603   \LP@set@env@prefix{\LP@KD}%
1604   \setcounter{\LP@rows}{\LP@KD@rows}%
1605   \setcounter{\LP@columns}{\LP@KD@columns}%
1606   \stepcounter{\LP@rows}%
1607   \stepcounter{\LP@columns}%
1608   \begin{minipage}[t]{\LP@KD@width}%
1609     \ifthenelse{\equal{\LP@KD@title}{}}{%
1610       {\par\enspace\par}% empty
1611     }{\enspace\par\noindent\hspace{\LP@KD@titleindent}\parbox{\LP@KD@titlewidth}{\strut\LP@KD@title}%
1612     }
1613     \begin{tikzpicture}[LP@preset,scale=\LP@KD@scale]%
1614       \LP@drawbackground{1}{1}{\LP@KD@columns}{\LP@KD@rows}{\LP@KD@bgcolor}%
1615       \LP@drawgrid{1}{1}{\LP@KD@columns}{\LP@KD@rows}{1cm}%

```

```

1615 }%
1616 {%
1617     \end{tikzpicture}%
1618     \LP@drawcounter{\LP@KD@counterstyle}%
1619     \stepcounter{LP@puzzlecounter}%
1620 \end{minipage}%
1621 }%

1622 %
1623 % killersudoku environment and options
1624 %
1625 \newcommand*\LP@KSDK@init@prefix{\LP@KSDK}%
1626 \newcommand*\LP@KSDK@init@package{killersudoku}%
1627 %
1628 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{rows}{5}%
1629 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{columns}{5}%
1630 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{scale}{1}%
1631 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{counterstyle}{none}%
1632 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{bgcolor}{}%
1633 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{width}{5.1cm}%
1634 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{cvoffset}{-23pt}%
1635 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{title}{}%
1636 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{titleindent}{0cm}%
1637 \LP@define@key{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{titlewidth}{5.1cm}%
1638 \LP@define@choicekey@fontsize{\LP@KSDK@init@prefix}{\LP@KSDK@init@package}{Large}%
1639 %
1640 \let\killersudokucell\setcell%
1641 %
1642 \newcommand*\killersudokusetup[1]%
1643 {%
1644     \setkeys{killersudoku.sty}{#1}%
1645 }%

```

killersudoku

```

1646 \newenvironment{killersudoku}[1][]%
1647 {%
1648     \setkeys{killersudoku}{#1}%
1649     \LP@set@package{killersudoku}%
1650     \LP@set@env@prefix{\LP@KSDK}%
1651     \setcounter{LP@rows}{\LP@KSDK@rows}%
1652     \setcounter{LP@columns}{\LP@KSDK@columns}%
1653     \stepcounter{LP@rows}%
1654     \stepcounter{LP@columns}%
1655     \begin{minipage}[t]{\LP@KSDK@width}%
1656         \ifthenelse{\equal{\LP@KSDK@title}{} }%
1657         {\par\enspace\par}% empty
1658         {\enspace\par\noindent\hspace{\LP@KSDK@titleindent}\parbox{\LP@KSDK@titlewidth}{\stru
1659     \begin{tikzpicture}[LP@preset,scale=\LP@KSDK@scale]%
1660         \LP@drawbackground{1}{1}{\LP@KSDK@columns}{\LP@KSDK@rows}{\LP@KSDK@bgcolor}%
1661         \LP@drawgrid{1}{1}{\LP@KSDK@columns}{\LP@KSDK@rows}{1cm}%
1662 }%

```

```

1663 {%
1664     \end{tikzpicture}%
1665     \LP@drawcounter{\LP@KSDK@counterstyle}%
1666     \stepcounter{\LP@puzzlecounter}%
1667     \end{minipage}%
1668 }%

1669 %
1670 % laserbeam environment and options
1671 %
1672 \newcommand*\LP@LB@init@prefix{\LP@LB}%
1673 \newcommand*\LP@LB@init@package{laserbeam}%
1674 %
1675 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{rows}{5}%
1676 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{columns}{5}%
1677 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{scale}{1}%
1678 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{counterstyle}{none}%
1679 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{bgcolor}{}%
1680 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{width}{6.5cm}%
1681 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{cvoffset}{-38pt}%
1682 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{title}{}%
1683 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{titleindent}{0cm}%
1684 \LP@define@key{\LP@LB@init@prefix}{\LP@LB@init@package}{titlewidth}{6.5cm}%
1685 \LP@define@choicekey@fontsize{\LP@LB@init@prefix}{\LP@LB@init@package}{Large}%
1686 %
1687 \let\laserH\LP@toprow%
1688 \let\laserV\LP@leftcolumn%
1689 \let\mirrorH\LP@bottomrow@edge%
1690 \let\mirrorV\LP@rightcolumn@edge%
1691 %
1692 \newcommand*\laser[2][red]%
1693 {%
1694 %   avoid little laser peaks outside grid
1695   \clip (1,1) rectangle (\arabic{\LP@columns},\arabic{\LP@rows});%
1696   \framearea{#1}{#2}%
1697 }%
1698 %
1699 \newcommand*\laserbeamsetup[1]%
1700 {%
1701   \setkeys{laserbeam.sty}{#1}%
1702 }%

```

`\placecross` Place a cross in the bottom left corner of the grid cell.

`\placecross{<column>}{<row>}`

```

1703 \newcommand*\placecross[2]%
1704 {%
1705   \LP@G@setcellcontent{#1}{#2}{\LP@Cross}%
1706 }%

```

`\placemirror` Place a mirror in the bottom left corner of the grid cell.

`\placemirror{<column>}{<row>}`

```
1707 \newcommand*\placemirror[3]%
1708 {%
1709   \LP@G@setcellcontent{#1}{#2}{\LP@Mirror{#3}}%
1710 }%
```

`\placearrow` Place an arrow in the bottom left corner of the grid cell. `{<direction>}` may be: RightUp, LeftUp, LeftDown or RightDown

`\placearrow{<column>}{<row>}{<direction>}`

```
1711 \newcommand*\placearrow[3]%
1712 {%
1713   \LP@G@setcellcontent{#1}{#2}{\LP@Arrow{#3}}%
1714 }%
```

`\LP@Arrow` We define a TikZ picture for an arrow in four directions: RightUp, LeftUp, LeftDown or RightDown

```
1715 \newcommand*\LP@Arrow[1]%
1716 {%
1717   \LP@set@LP@scale{\LP@env@prefix}%
1718   \def\LP@rotate{45}%
1719   \ifthenelse{\equal{#1}{LeftUp}}{\def\LP@rotate{135}}{}%
1720   \ifthenelse{\equal{#1}{LeftDown}}{\def\LP@rotate{225}}{}%
1721   \ifthenelse{\equal{#1}{RightDown}}{\def\LP@rotate{315}}{}%
```

We define the line width of the arrow base on scale. Line widths are absolut and not influenced by a scale factor of the picture.

```
1722 \pgfmathsetmacro{\LPlinewidth}{3pt*\LP@scale}%
1723 \tikz\draw[->,line width=\LPlinewidth,%
1724           rotate=\LP@rotate,scale=\LP@scale]%
1725           (0.1,.5) -- (0.9,.5);%
1726 }%
```

`\LP@Cross` We define a TikZ picture of a cross.

```
1727 \newcommand*\LP@Cross%
1728 {%
1729   \LP@set@LP@scale{\LP@env@prefix}%
1730   \begin{tikzpicture}%
```

We ensure that the cross is on top of all elements on the main layer.

```
1731   \begin{pgfonlayer}{LPforeground}%
1732   \pgfmathsetmacro{\LPlinewidth}{3pt*\LP@scale}%
```

For drawing the cross we defined a line width. Now we can draw the cross with the predefined color `LP@cc@cross`.

```
1733   \draw[line width=\LPlinewidth,scale=\LP@scale,color=LP@cc@cross]%
```

```

1734      (.35,.35) -- (0.65,.65) -- (.5,.5) -- (.65,.35) -- (.35,.65);%
1735      \end{pgfonlayer}%
1736      \end{tikzpicture}%
1737 }%

```

`\LP@Mirror` We define a TikZ picture for a mirror.

```

\LP@Mirror{<direction>}

1738 \newcommand*\LP@Mirror[1]%
1739 {%

```

Based on `{<direction>}`, we define the angle of the mirror.

```

1740   \def\LP@rotate{0}%
1741   \ifthenelse{\equal{#1}{V}}{\def\LP@rotate{90}}{}%
1742   \LP@set@LP@scale{\LP@env@prefix}%
1743   \begin{tikzpicture}%

```

Mirrors should be on to of everything, therefore we draw them on the `LPforegroundtwo` layer and use the predefined color `LP@c@mirror`.

```

1744     \begin{pgfonlayer}{LPforegroundtwo}%
1745     \pgfmathsetmacro{\LPlinewidth}{3pt*\LP@scale}%
1746     \draw[line width=\LPlinewidth,scale=\LP@scale,%
1747           color=LP@c@mirror,rotate=\LP@rotate]%
1748           (.35,.5) -- (0.65,.5);%
1749     \end{pgfonlayer}%
1750   \end{tikzpicture}%
1751 }%

```

`laserbeam`

```

1752 \newenvironment{laserbeam}[1][]%
1753 {%
1754   \setkeys{laserbeam}{#1}%
1755   \LP@set@package{laserbeam}%
1756   \LP@set@env@prefix{LP@LB}%
1757   \setcounter{LP@rows}{\LP@LB@rows}%
1758   \setcounter{LP@columns}{\LP@LB@columns}%
1759   \stepcounter{LP@rows}%
1760   \stepcounter{LP@columns}%
1761   \begin{minipage}[t]{\LP@LB@width}%
1762     \ifthenelse{\equal{\LP@LB@title}{}}{}%
1763     {\par\enspace\par}% empty
1764     {\enspace\par\noindent\hspace{\LP@LB@titleindent}\parbox{\LP@LB@titlewidth}{\strut\LP@LB@title}}%
1765     \begin{tikzpicture}[LP@preset,scale=\LP@LB@scale]%
1766       \LP@drawbackground{1}{1}{\LP@LB@columns}{\LP@LB@rows}{\LP@LB@bgcolor}%
1767       \LP@drawgrid{1}{1}{\LP@LB@columns}{\LP@LB@rows}{1cm}%
1768     }%
1769   {%
1770     \end{tikzpicture}%
1771     \LP@drawcounter{\LP@LB@counterstyle}%

```

```

1772 \stepcounter{LP@puzzlecounter}%
1773 \end{minipage}%
1774 }%

1775 %
1776 % lpsudoku environment and options
1777 %
1778 \newcommand*{LP@SDK@init@prefix}{LP@SDK}%
1779 \newcommand*{LP@SDK@init@package}{lpsudoku}%
1780 %
1781 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{rows}{9}%
1782 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{columns}{9}%
1783 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{scale}{1}%
1784 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{counterstyle}{none}%
1785 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{bgcolor}{}%
1786 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{width}{9.1cm}%
1787 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{cvmoffset}{-23pt}%
1788 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{title}{}%
1789 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{titleindent}{0cm}%
1790 \LP@define@key{LP@SDK@init@prefix}{LP@SDK@init@package}{titlewidth}{9.1cm}%
1791 \LP@define@choicekey@fontsize{LP@SDK@init@prefix}{LP@SDK@init@package}{Large}%
1792 %
1793 \let\lpsudokucell\setcell%
1794 %
1795 \newcommand*{lpsudokusetup[1]}%
1796 {%
1797 \setkeys{lpsudoku.sty}{#1}%
1798 }%

```

`\LP@drawsudokugrid` For the lpsudoku and skyscrapers environments, we need to add the typical thick Sudoku lines to the standard grid.

```

1799 \newcommand*{LP@drawsudokugrid}%
1800 {%
1801 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1802 (1,1) -- (1,10);%
1803 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1804 (4,1) -- (4,10);%
1805 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1806 (7,1) -- (7,10);%
1807 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1808 (10,1) -- (10,10);%
1809 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1810 (1,1) -- (10,1);%
1811 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1812 (1,4) -- (10,4);%
1813 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1814 (1,7) -- (10,7);%
1815 \draw[line width=\LP@thicklines,draw opacity=\LP@draw@opacity]%
1816 (1,10) -- (10,10);%
1817 }%

```

`\LP@magnetsgrid` For the magnets environment, we need to add some lines to the standard grid.

```
1818 \newcommand*\LP@magnetsgrid%
1819 {%
```

First, we copy the values of `fontsize`, `columns` and `rows` from the environment we are in at the moment. Then, we step `columns` and `rows` to get the upper right grid coordinate.

```
1820 \LP@set@LP@fontsize{\LP@env@prefix}%
1821 \LP@set@LP@columns{\LP@env@prefix}%
1822 \LP@set@LP@rows{\LP@env@prefix}%
1823 \setcounter{LP@counti}{\LP@columns}% max column
1824 \setcounter{LP@countii}{\LP@rows}% max row
1825 \stepcounter{LP@counti}%
1826 \stepcounter{LP@countii}%
```

Now, we can draw the additional lines and the + and - signs.

```
1827 \draw[step=1cm,line width=\LP@normallines]%
1828 (-1,1) grid (1,\arabic{LP@countii});%
1829 \draw[step=1cm,line width=\LP@normallines]%
1830 (1,\arabic{LP@countii}) grid ++(\LP@columns,2);%
1831 \draw[line width=\LP@normallines]%
1832 (0,\arabic{LP@countii}) -- ++(0,1) -- ++(1,0);%
1833 \draw[line width=\LP@thicklines]%
1834 (-1,1) rectangle (1,\arabic{LP@countii});%
1835 \draw[line width=\LP@thicklines]%
1836 (1,\arabic{LP@countii}) rectangle ++(\LP@columns,2);%
1837 \draw[line width=\LP@thicklines]%
1838 (1,\arabic{LP@countii}) rectangle ++(-2,2);%
1839 \node[font=\LP@fontsize\bfseries] at (0.5,\arabic{LP@counti}.5)%
1840 {$-$};%
1841 \stepcounter{LP@counti}%
1842 \node[font=\LP@fontsize\bfseries] at (-0.5,\arabic{LP@counti}.5)%
1843 {$+$};%
1844 }%
```

`lpsudoku`

```
1845 \newenvironment{lpsudoku}[1][]%
1846 {%
1847 \setkeys{lpsudoku}{#1}%
1848 \LP@set@package{lpsudoku}%
1849 \LP@set@env@prefix{LP@SDK}%
1850 \setcounter{LP@rows}{\LP@SDK@rows}%
1851 \setcounter{LP@columns}{\LP@SDK@columns}%
1852 \stepcounter{LP@rows}%
1853 \stepcounter{LP@columns}%
1854 \begin{minipage}[t]{\LP@SDK@width}%
1855 \ifthenelse{\equal{\LP@SDK@title}{}}%
1856 {\par\enspace\par}% empty
```



```

1857     {\enspace\par\noindent\hspace{\LP@SDK@titleindent}\parbox{\LP@SDK@titlewidth}{\strut\
1858     \begin{tikzpicture}[LPpreset,scale=\LP@SDK@scale]%
1859         \LP@drawbackground{1}{1}{\LP@SDK@columns}{\LP@SDK@rows}{\LP@SDK@bgcolor}%
1860         \LP@drawgrid{1}{1}{\LP@SDK@columns}{\LP@SDK@rows}{1cm}%
1861         \LP@drawsudokugrid%
1862     }%
1863 {%
1864     \end{tikzpicture}%
1865     \LP@drawcounter{\LP@SDK@counterstyle}%
1866     \stepcounter{LP@puzzlecounter}%
1867 \end{minipage}%
1868 }%

1869 %
1870 % magiclabyrinth environment and options
1871 %
1872 \newcommand*\LP@ML@init@prefix{LP@ML}%
1873 \newcommand*\LP@ML@init@package{magiclabyrinth}%
1874 %
1875 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{rows}{5}%
1876 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{columns}{5}%
1877 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{scale}{1}%
1878 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{counterstyle}{none}%
1879 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{bgcolor}{}%
1880 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{width}{5.1cm}%
1881 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{cvoffset}{-23pt}%
1882 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{title}{}%
1883 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{titleindent}{0cm}%
1884 \LP@define@key{\LP@ML@init@prefix}{\LP@ML@init@package}{titlewidth}{5.1cm}%
1885 \LP@define@choicekey@fontsize{\LP@ML@init@prefix}{\LP@ML@init@package}{Large}%
1886 %
1887 \let\magiclabyrinthcell\setcell%
1888 %
1889 \newcommand*\magiclabyrinthsetup[1]%
1890 {%
1891     \setkeys{magiclabyrinth.sty}{#1}%
1892 }%

\mlline

1893 \newcommand*\mlline[1]%
1894 {%
1895     \draw[color=black, line width=\LP@thicklines] #1;%
1896 }%

\MasyuW

1897 \newcommand*\MasyuW%
1898 {%
1899     \begin{tikzpicture}[scale=0.7*\LP@scale]%
1900         \draw[color=black,fill=white] (0.5,0.5) circle (0.5cm);%
1901     \end{tikzpicture}%

```

```

1902 }%

magiclabyrinth

1903 \newenvironment{magiclabyrinth}[1][]%
1904 {%
1905   \setkeys{magiclabyrinth}{#1}%
1906   \LP@set@package{magiclabyrinth}%
1907   \LP@set@env@prefix{LP@ML}%
1908   \setcounter{LP@rows}{\LP@ML@rows}%
1909   \setcounter{LP@columns}{\LP@ML@columns}%
1910   \stepcounter{LP@rows}%
1911   \stepcounter{LP@columns}%
1912   \begin{minipage}[t]{\LP@ML@width}%
1913     \ifthenelse{\equal{\LP@ML@title}{}}{%
1914       {\par\enspace\par}% empty
1915       {\enspace\par\noindent\hspace{\LP@ML@titleindent}\parbox{\LP@ML@titlewidth}{\strut\LP@ML@title}%
1916       \begin{tikzpicture}[LP@preset,scale=\LP@ML@scale]%
1917         \LP@drawbackground{1}{1}{\LP@ML@columns}{\LP@ML@rows}{\LP@ML@bgcolor}%
1918         \LP@drawgrid{1}{1}{\LP@ML@columns}{\LP@ML@rows}{1cm}%
1919       }%
1920     }%
1921     \end{tikzpicture}%
1922     \LP@drawcounter{\LP@ML@counterstyle}%
1923     \stepcounter{LP@puzzlecounter}%
1924   \end{minipage}%
1925 }%

1926 %
1927 % magnets environment and options
1928 %
1929 \newcommand*\LP@MN@init@prefix{LP@MN}%
1930 \newcommand*\LP@MN@init@package{magnets}%
1931 %
1932 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{rows}{6}%
1933 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{columns}{6}%
1934 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{scale}{1}%
1935 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{counterstyle}{none}%
1936 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{bgcolor}{}%
1937 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{width}{8.1cm}%
1938 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{cvooffset}{-23pt}%
1939 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{title}{}%
1940 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{titleindent}{0cm}%
1941 \LP@define@key{\LP@MN@init@prefix}{\LP@MN@init@package}{titlewidth}{8.1cm}%
1942 \LP@define@choicekey@fontsize{\LP@MN@init@prefix}{\LP@MN@init@package}{Large}%
1943 %
1944 \let\magnetscell\setcell%
1945 %
1946 \let\minusH\LP@toprow%
1947 \let\minusV\LP@leftcolumn%
1948 %
1949 \newcommand*\plusV[1]%

```

```

1950 {%
1951 \LP@set@LP@fontsize{\LP@env@prefix}%
1952 \setcounter{LP@counti}{1}%
1953 \foreach \LP@element in{#1}%
1954 {%
1955 \node at (-0.5,\arabic{LP@counti}\LP@Pfive){\LP@fontsize\LP@element};%
1956 \stepcounter{LP@counti}%
1957 };%
1958 }%
1959 %
1960 \newcommand*\plusH[1]%
1961 {%
1962 \LP@set@LP@fontsize{\LP@env@prefix}%
1963 \LP@set@LP@rows{\LP@env@prefix}%
1964 \setcounter{LP@counti}{1}%
1965 \setcounter{LP@countii}{\LP@rows}%
1966 \addtocounter{LP@countii}{2}%
1967 \foreach \LP@element in{#1}%
1968 {%
1969 \node at (\arabic{LP@counti}\LP@Pfive,\arabic{LP@countii}\LP@Pfive){\LP@fontsize\LP@element};%
1970 \stepcounter{LP@counti}%
1971 };%
1972 }%
1973 %
1974 \newcommand*\magnetssetup[1]%
1975 {%
1976 \setkeys{magnets.sty}{#1}%
1977 }%

```

\PMH

```

1978 \newcommand*\PMH[1]%
1979 {%
1980 \LP@set@LP@fontsize{\LP@env@prefix}%
1981 \foreach \LP@fe@column/\LP@fe@row in {#1}%
1982 {%
1983 \setcounter{LP@counti}{\LP@fe@column}%
1984 \setcounter{LP@countii}{\LP@fe@row}%
1985 \draw[line width=\LP@thicklines,fill=white]%
1986 (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(2,1);%
1987 \node[font=\LP@fontsize\bfseries]%
1988 at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$+}$;%
1989 \stepcounter{LP@counti}%
1990 \node[font=\LP@fontsize\bfseries]%
1991 at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$-$};%
1992 };%
1993 }%

```

\MPH

```

1994 \newcommand*\MPH[1]%
1995 {%

```

```

1996 \LP@set@LP@fontsize{\LP@env@prefix}%
1997 \foreach \LP@fe@column/\LP@fe@row in {#1}%
1998 {%
1999   \setcounter{LP@counti}{\LP@fe@column}%
2000   \setcounter{LP@countii}{\LP@fe@row}%
2001   \draw[line width=\LP@thicklines,fill=white]%
2002     (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(2,1);%
2003   \node[font=\LP@fontsize\bfseries]%
2004     at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$-$};%
2005   \stepcounter{LP@counti}%
2006   \node[font=\LP@fontsize\bfseries]%
2007     at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$+$};%
2008 };%
2009 }%

```

\PMV

```

2010 \newcommand*\PMV[1]%
2011 {%
2012   \LP@set@LP@fontsize{\LP@env@prefix}%
2013   \foreach \LP@fe@column/\LP@fe@row in {#1}%
2014   {%
2015     \setcounter{LP@counti}{\LP@fe@column}%
2016     \setcounter{LP@countii}{\LP@fe@row}%
2017     \draw[line width=\LP@thicklines,fill=white]%
2018       (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(1,2);%
2019     \node[font=\LP@fontsize\bfseries]%
2020       at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$+$};%
2021     \stepcounter{LP@countii}%
2022     \node[font=\LP@fontsize\bfseries]%
2023       at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$-$};%
2024   };%
2025 }%

```

\MPV

```

2026 \newcommand*\MPV[1]%
2027 {%
2028   \LP@set@LP@fontsize{\LP@env@prefix}%
2029   \foreach \LP@fe@column/\LP@fe@row in {#1}%
2030   {%
2031     \setcounter{LP@counti}{\LP@fe@column}%
2032     \setcounter{LP@countii}{\LP@fe@row}%
2033     \draw[line width=\LP@thicklines,fill=white]%
2034       (\arabic{LP@counti},\arabic{LP@countii}) rectangle ++(1,2);%
2035     \node[font=\LP@fontsize\bfseries]%
2036       at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$-$};%
2037     \stepcounter{LP@countii}%
2038     \node[font=\LP@fontsize\bfseries]%
2039       at (\arabic{LP@counti}.5,\arabic{LP@countii}.5) {$+$};%
2040   };%
2041 }%

```

\magnetsH

```
2042 \newcommand*\magnetsH[1]%
2043 {%
2044   \foreach \LP@fe@column/\LP@fe@row in {#1}%
2045   {%
2046     \draw[line width=\LP@thicklines,fill=white]%
2047       (\LP@fe@column,\LP@fe@row) rectangle ++(2,1);%
2048   };%
2049 }%
```

\magnetsV

```
2050 \newcommand*\magnetsV[1]%
2051 {%
2052   \foreach \LP@fe@column/\LP@fe@row in {#1}%
2053   {%
2054     \draw[line width=\LP@thicklines,fill=white]%
2055       (\LP@fe@column,\LP@fe@row) rectangle ++(1,2);%
2056   };%
2057 }%
```

magnets

```
2058 \newenvironment{magnets}[1][]%
2059 {%
2060   \setkeys{magnets}{#1}%
2061   \LP@set@package{magnets}%
2062   \LP@set@env@prefix{LP@MN}%
2063   \setcounter{LP@rows}{\LP@MN@rows}%
2064   \setcounter{LP@columns}{\LP@MN@columns}%
2065   \stepcounter{LP@rows}%
2066   \stepcounter{LP@columns}%
2067   \begin{minipage}[t]{\LP@MN@width}%
2068     \ifthenelse{\equal{\LP@MN@title}{}}{%
2069       {\par\enspace\par}% empty
2070     }{\enspace\par\noindent\hspace{\LP@MN@titleindent}\parbox{\LP@MN@titlewidth}{\strut\LP
2071       \begin{tikzpicture}[LP@preset,scale=\LP@MN@scale]%
2072         \LP@drawbackground{1}{1}{\LP@MN@columns}{\LP@MN@rows}{\LP@MN@bgcolor}%
2073         \LP@drawgrid{1}{1}{\LP@MN@columns}{\LP@MN@rows}{1cm}%
2074         \LP@magnetsgrid%
2075         \framepuzzle%
2076       }%
2077     }%
2078     \end{tikzpicture}%
2079     \LP@drawcounter{\LP@MN@counterstyle}%
2080     \stepcounter{LP@puzzlecounter}%
2081   \end{minipage}%
2082 }%

2083 %
2084 % masyu environment and options
```

```

2085 %
2086 \newcommand*\LP@MY@init@prefix{LP@MY}%
2087 \newcommand*\LP@MY@init@package{masyu}%
2088 %
2089 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{rows}{5}%
2090 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{columns}{5}%
2091 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{scale}{1}%
2092 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{counterstyle}{none}%
2093 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{color}{green}%
2094 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{bgcolor}{}%
2095 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{width}{5.1cm}%
2096 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{cvoffset}{-23pt}%
2097 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{title}{}%
2098 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{titleindent}{0cm}%
2099 \LP@define@key{\LP@MY@init@prefix}{\LP@MY@init@package}{titlewidth}{5.1cm}%
2100 \LP@define@choicekey@fontsize{\LP@MY@init@prefix}{\LP@MY@init@package}{Large}%
2101 %
2102 \let\masyucell\setcell%
2103 %
2104 \newcommand*\masyusetup[1]%
2105 {%
2106   \setkeys{masyu.sty}{#1}%
2107 }%

```

\MasyuB

```

2108 \newcommand*\MasyuB%
2109 {%
2110   \begin{tikzpicture}[scale=0.7*\LP@scale]%
2111     \fill[color=black] (0.5,0.5) circle (0.5cm);%
2112   \end{tikzpicture}%
2113 }%

```

\masyuline

```

2114 \newcommand*\masyuline[1]%
2115 {%
2116   \LP@set@LP@color{\LP@env@prefix}%
2117   \begin{puzzleforeground}%
2118     \bgroup%
2119     \def\LP@rel@tikzpath{.5}%
2120     \draw[color=\LP@color, line width=\LP@thicklines] #1;%
2121   \egroup%
2122 \end{puzzleforeground}%
2123 }%

```

masyu

```

2124 \newenvironment{masyu}[1][]%
2125 {%
2126   \setkeys{masyu}{#1}%

```

```

2127 \LP@set@package{masyu}%
2128 \LP@set@env@prefix{LP@MY}%
2129 \setcounter{LP@rows}{\LP@MY@rows}%
2130 \setcounter{LP@columns}{\LP@MY@columns}%
2131 \stepcounter{LP@rows}%
2132 \stepcounter{LP@columns}%
2133 \begin{minipage}[t]{\LP@MY@width}%
2134 \ifthenelse{\equal{\LP@MY@title}{}}{%
2135 \par\enspace\par}% empty
2136 {\enspace\par\noindent\hspace{\LP@MY@titleindent}\parbox{\LP@MY@titlewidth}{\strut\LP
2137 \begin{tikzpicture}[LP@preset,scale=\LP@MY@scale]%
2138 \LP@drawbackground{1}{1}{\LP@MY@columns}{\LP@MY@rows}{\LP@MY@bgcolor}%
2139 \LP@drawgrid{1}{1}{\LP@MY@columns}{\LP@MY@rows}{1cm}%
2140 }%
2141 {%
2142 \end{tikzpicture}%
2143 \LP@drawcounter{\LP@MY@counterstyle}%
2144 \stepcounter{LP@puzzlecounter}%
2145 \end{minipage}%
2146 }%

2147 %
2148 % minesweeper environment and options
2149 %
2150 \newcommand*\LP@MS@init@prefix{LP@MS}%
2151 \newcommand*\LP@MS@init@package{minesweeper}%
2152 %
2153 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{rows}{5}%
2154 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{columns}{5}%
2155 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{scale}{1}%
2156 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{counterstyle}{none}%
2157 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{bgcolor}{}%
2158 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{width}{5.1cm}%
2159 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{cvoffset}{-23pt}%
2160 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{title}{}%
2161 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{titleindent}{0cm}%
2162 \LP@define@key{\LP@MS@init@prefix}{\LP@MS@init@package}{titlewidth}{5.1cm}%
2163 \LP@define@choicekey@fontsize{\LP@MS@init@prefix}{\LP@MS@init@package}{Large}%
2164 %
2165 \let\minesweepercell\setcell%
2166 %
2167 \newcommand*\minesweepersetup[1]%
2168 {%
2169 \setkeys{minesweeper.sty}{#1}%
2170 }%

```

\LP@Mine We define a TikZ picture for a mine.

```

2171 \newcommand*\LP@Mine%
2172 {%
2173 \begin{tikzpicture}[scale=\LP@scale]%

```

We simply draw a shaded ball with four spikes.

```

2174 \fill[color=black] (.2,.5) -- (.5,.6) -- (.5,.4) -- cycle;%
2175 \fill[color=black] (.8,.5) -- (.5,.6) -- (.5,.4) -- cycle;%
2176 \fill[color=black] (.5,.8) -- (.4,.5) -- (.6,.5) -- cycle;%
2177 \fill[color=black] (.5,.2) -- (.4,.5) -- (.6,.5) -- cycle;%
2178 \shade[ball color=black] (.5,.5) circle (.225cm);%
2179 \end{tikzpicture}%
2180 }%

```

\Mine A user command for mines.

```

2181 \let\Mine\LP@Mine%

```

minesweeper

```

2182 \newenvironment{minesweeper}[1][]%
2183 {%
2184 \setkeys{minesweeper}{#1}%
2185 \LP@set@package{minesweeper}%
2186 \LP@set@env@prefix{LP@MS}%
2187 \setcounter{LP@rows}{\LP@MS@rows}%
2188 \setcounter{LP@columns}{\LP@MS@columns}%
2189 \stepcounter{LP@rows}%
2190 \stepcounter{LP@columns}%
2191 \begin{minipage}[t]{\LP@MS@width}%
2192 \ifthenelse{\equal{\LP@MS@title}{}}{%
2193 {\par\enspace\par}% empty
2194 {\enspace\par\noindent\hspace{\LP@MS@titleindent}\parbox{\LP@MS@titlewidth}{\strut\LP@
2195 \begin{tikzpicture}[LP@preset,scale=\LP@MS@scale]%
2196 \LP@drawbackground{1}{1}{\LP@MS@columns}{\LP@MS@rows}{\LP@MS@bgcolor}%
2197 \LP@drawgrid{1}{1}{\LP@MS@columns}{\LP@MS@rows}{1cm}%
2198 }%
2199 }%
2200 \end{tikzpicture}%
2201 \LP@drawcounter{\LP@MS@counterstyle}%
2202 \stepcounter{LP@puzzlecounter}%
2203 \end{minipage}%
2204 }%

2205 %
2206 % nonogram environment and options
2207 %
2208 \newcommand*\LP@NG@init@prefix{LP@NG}%
2209 \newcommand*\LP@NG@init@package{nonogram}%
2210 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{rows}{5}%
2211 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{columns}{5}%
2212 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{extracells}{5}%
2213 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{helplines}{5}%
2214 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{solution}{false}%
2215 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{scale}{1}%
2216 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{counterstyle}{none}%

```



```

2217 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{bgcolor}{}%
2218 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{color}{black}%
2219 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{width}{5.1cm}%
2220 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{cvooffset}{-23pt}%
2221 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{title}{}%
2222 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{titleindent}{0cm}%
2223 \LP@define@key{\LP@NG@init@prefix}{\LP@NG@init@package}{titlewidth}{5.1cm}%
2224 \LP@define@choicekey@fontsize{\LP@NG@init@prefix}{\LP@NG@init@package}{Large}%
2225 %

```

\nonogramrow

```

2226 \newcommand*\nonogramrow[2]%
2227 {%
2228   \foreach \LP@fe@column/\LP@fe@length in {#2}%
2229   {%
2230     \setcounter{LP@whiledo@i}{\LP@fe@length}%
2231     \setcounter{LP@countiii}{\LP@fe@column}%
2232     \whiledo{\value{LP@whiledo@i}>0}%
2233     {%
2234       \fillcell{\arabic{LP@countiii}}{#1}%
2235       \addtocounter{LP@countiii}{1}%
2236       \addtocounter{LP@whiledo@i}{-1}%
2237     }%
2238   };%
2239 }%

```

\nonogramcolumn

```

2240 \newcommand*\nonogramcolumn[2]%
2241 {%
2242   \foreach \LP@fe@row/\LP@fe@length in {#2}%
2243   {%
2244     \setcounter{LP@whiledo@i}{\LP@fe@length}%
2245     \setcounter{LP@countiii}{\LP@fe@row}%
2246     \whiledo{\value{LP@whiledo@i}>0}%
2247     {%
2248       \fillcell{#1}{\arabic{LP@countiii}}%
2249       \addtocounter{LP@countiii}{1}%
2250       \addtocounter{LP@whiledo@i}{-1}%
2251     }%
2252   };%
2253 }%

```

\nonogramV

```

2254 \newcommand*\nonogramV[1]%
2255 {%
2256   \LP@set@LP@fontsize{\LP@env@prefix}%
2257   \setcounter{LP@whiledo@i}{0}%
2258   \foreach \LP@line in {#1}%

```

```

2259 {%
2260   \setcounter{LP@countiii}{0}%
2261   \addtocounter{LP@whiledo@i}{1}%
2262   \foreach \LP@element in \LP@line%
2263   {%
2264     \ifthenelse{\value{LP@countiii}=0}%
2265     {%
2266       {\LP@fontsize\node at (0.5,\arabic{LP@whiledo@i}.5){\LP@element};}%
2267     }%
2268     {%
2269       \ifthenelse{\value{LP@countiii}=-1}%
2270       {%
2271         {\LP@fontsize\node at (-0.5,\arabic{LP@whiledo@i}.5){\LP@element};}%
2272       }%
2273       {%
2274         \addtocounter{LP@countiii}{1}%
2275         {\LP@fontsize\node at
2276          (\arabic{LP@countiii}.5,\arabic{LP@whiledo@i}.5){\LP@element};}%
2277         \addtocounter{LP@countiii}{-1}%
2278       }%
2279     }%
2280     \addtocounter{LP@countiii}{-1}%
2281   }%
2282 };%
2283 }%

```

\nonogramH

```

2284 \newcommand*\nonogramH[1]%
2285 {%
2286   \setcounter{LP@whiledo@i}{0}%
2287   \foreach \LP@line in {#1}%
2288   {%
2289     \setcounter{LP@countiii}{\LP@rows}%
2290     \addtocounter{LP@countiii}{1}%
2291     \addtocounter{LP@whiledo@i}{1}%
2292     \foreach \LP@element in \LP@line%
2293     {%
2294       {\LP@fontsize\node at%
2295        (\arabic{LP@whiledo@i}.5,\arabic{LP@countiii}.5){\LP@element};}%
2296       \addtocounter{LP@countiii}{1}%
2297     }%
2298   };%
2299 }%

```

\LP@nonogramgrid

```

2300 \newcommand*\LP@nonogramgrid%
2301 {%
2302   \LP@set@LP@columns{\LP@env@prefix}%
2303   \LP@set@LP@rows{\LP@env@prefix}%
2304   \setcounter{LP@counti}{\LP@columns}% max column

```

```

2305 \setcounter{LP@countii}{\LP@rows}% max row
2306 \setcounter{LP@countiii}{\LP@NG@extracells}%
2307 \stepcounter{LP@countii}%
2308 \stepcounter{LP@countii}%
2309 \addtocounter{LP@countiii}{-1}%
2310 \setcounter{LP@whiledo@i}{1}%
2311 \setcounter{LP@whiledo@ii}{\LP@NG@extracells}%
2312 \addtocounter{LP@whiledo@ii}{\arabic{LP@countii}}%
2313 \setcounter{LP@countiii}{\arabic{LP@countii}}%
2314 \addtocounter{LP@countiii}{\LP@NG@extracells}%
2315 \addtocounter{LP@countii}{1}%
2316 \whiledo{\value{LP@whiledo@i}<\value{LP@countii}}%
2317 {%
2318   \draw[line width=\LP@normallines]%
2319     (\arabic{LP@whiledo@i},1) --%
2320     (\arabic{LP@whiledo@i},\arabic{LP@countiii});%
2321   \addtocounter{LP@whiledo@i}{1}%
2322 }%
2323 \addtocounter{LP@countii}{-1}%
2324 \setcounter{LP@whiledo@i}{1}%
2325 \setcounter{LP@whiledo@ii}{-\LP@NG@extracells}%
2326 \addtocounter{LP@whiledo@ii}{1}%
2327 \addtocounter{LP@countii}{1}%
2328 \whiledo{\value{LP@whiledo@i}<\value{LP@countii}}%
2329 {%
2330   \draw[line width=\LP@normallines]%
2331     (\arabic{LP@whiledo@ii},\arabic{LP@whiledo@i}) --%
2332     (\arabic{LP@countii},\arabic{LP@whiledo@i});%
2333   \addtocounter{LP@whiledo@i}{1}%
2334 }%
2335 \addtocounter{LP@countii}{-1}%
2336 \setcounter{LP@countiii}{\LP@NG@helplines}%
2337 \ifthenelse{\arabic{LP@countiii}>0}%
2338 {%
2339   \setcounter{LP@whiledo@i}{1}%
2340   \addtocounter{LP@whiledo@i}{\LP@NG@helplines}%
2341   \setcounter{LP@whiledo@ii}{\LP@NG@extracells}%
2342   \addtocounter{LP@whiledo@ii}{\arabic{LP@countii}}%
2343   \setcounter{LP@countiii}{\arabic{LP@countii}}%
2344   \addtocounter{LP@countiii}{\LP@NG@extracells}%
2345   \whiledo{\value{LP@whiledo@i}<\value{LP@countii}}%
2346   {%
2347     \draw[line width=\LP@thicklines]%
2348       (\arabic{LP@whiledo@i},1) --%
2349       (\arabic{LP@whiledo@i},\arabic{LP@countiii});%
2350     \addtocounter{LP@whiledo@i}{\LP@NG@helplines}%
2351   }%
2352   \draw[line width=\LP@thicklines] (1,1) -- (1,\arabic{LP@countiii});%
2353   \draw[line width=\LP@thicklines]%
2354     (\arabic{LP@countii},1) --%
2355     (\arabic{LP@countii},\arabic{LP@countiii});%

```

```

2356 \setcounter{LP@whiledo@i}{1}%
2357 \addtocounter{LP@whiledo@i}{\LP@NG@helplines}%
2358 \setcounter{LP@whiledo@ii}{-\LP@NG@extracells}%
2359 \addtocounter{LP@whiledo@ii}{1}%
2360 \whiledo{\value{LP@whiledo@i}<\value{LP@countii}}%
2361 {%
2362   \draw[line width=\LP@thicklines]%
2363     (\arabic{LP@whiledo@ii},\arabic{LP@whiledo@i}) --%
2364     (\arabic{LP@counti},\arabic{LP@whiledo@i});%
2365   \addtocounter{LP@whiledo@i}{\LP@NG@helplines}%
2366 }%
2367 \draw[line width=\LP@thicklines]%
2368   (\arabic{LP@whiledo@ii},1) -- (\arabic{LP@counti},1);%
2369 \draw[line width=\LP@thicklines]%
2370   (\arabic{LP@whiledo@ii},\arabic{LP@countii}) --%
2371   (\arabic{LP@counti},\arabic{LP@countii});%
2372 }%
2373 {}%
2374 }%

```

\nonogramsetup

```

2375 \newcommand*\nonogramsetup[1]%
2376 {%
2377   \setkeys{nonogram.sty}{#1}%
2378 }%

```

nonogram

```

2379 \newenvironment{nonogram}[1][]%
2380 {%
2381   \setkeys{nonogram}{#1}%
2382   \LP@set@package{nonogram}%
2383   \LP@set@env@prefix{LP@NG}%
2384   \setcounter{LP@rows}{\LP@NG@rows}%
2385   \setcounter{LP@columns}{\LP@NG@columns}%
2386   \stepcounter{LP@rows}%
2387   \stepcounter{LP@columns}%
2388   \begin{minipage}[t]{\LP@NG@width}%
2389     \ifthenelse{\equal{\LP@NG@title}{}}{%
2390       {\par\enspace\par}% empty
2391       {\enspace\par\noindent\hspace{\LP@NG@titleindent}\parbox{\LP@NG@titlewidth}{\strut\LP@
2392         \begin{tikzpicture}[LP@preset,scale=\LP@NG@scale]%
2393           \LP@drawbackground{1}{1}{\LP@NG@columns}{\LP@NG@rows}{\LP@NG@bgcolor}%
2394           \LP@drawgrid{1}{1}{\LP@NG@columns}{\LP@NG@rows}{1cm}%
2395           \framepuzzle%
2396           \ifthenelse{\equal{\LP@NG@solution}{false}}{%
2397             {%
2398               \LP@nonogramgrid%
2399             }%
2400           }%
2401 }%

```

```

2402 {%
2403     \end{tikzpicture}%
2404     \LP@drawcounter{\LP@NG@counterstyle}%
2405     \stepcounter{LP@puzzlecounter}%
2406     \end{minipage}%
2407 }%

2408 %
2409 % numberlink environment and options
2410 %
2411 \newcommand*\LP@NL@init@prefix{LP@NL}%
2412 \newcommand*\LP@NL@init@package{numberlink}%
2413 %
2414 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{rows}{5}%
2415 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{columns}{5}%
2416 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{scale}{1}%
2417 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{counterstyle}{none}%
2418 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{bgcolor}{}%
2419 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{color}{red}%
2420 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{width}{5.1cm}%
2421 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{cvmoffset}{-23pt}%
2422 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{title}{}%
2423 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{titleindent}{0cm}%
2424 \LP@define@key{\LP@NL@init@prefix}{\LP@NL@init@package}{titlewidth}{5.1cm}%
2425 \LP@define@choicekey@fontsize{LP@NL@init@prefix}{\LP@NL@init@package}{Large}%
2426 %
2427 \newcommand*\numberlinkcell[3]%
2428 {%
2429     \fourwindsell{#1}{#2}{#3}{}%
2430 }%
2431 %
2432 %
2433 \newcommand*\numberlinksetup[1]%
2434 {%
2435     \setkeys{numberlink.sty}{#1}%
2436 }%

```

\link

```

2437 \newcommand*\link[2][]%
2438 {%
2439     \LP@set@LP@color{\LP@env@prefix}%
2440     \bgroup%
2441         \def\LP@rel@tikzpath{.5}%
2442         \LP@Line[color=\LP@color,#1]{#2}%
2443     \egroup%
2444 }%

```

numberlink

```

2445 \newenvironment{numberlink}[1][]%
2446 {%

```

```

2447 \setkeys{numberlink}{#1}%
2448 \LP@set@package{numberlink}%
2449 \LP@set@env@prefix{LP@NL}%
2450 \setcounter{LP@rows}{\LP@NL@rows}%
2451 \setcounter{LP@columns}{\LP@NL@columns}%
2452 \stepcounter{LP@rows}%
2453 \stepcounter{LP@columns}%
2454 \begin{minipage}[t]{\LP@NL@width}%
2455   \ifthenelse{\equal{\LP@NL@title}{}}{%
2456     {\par\enspace\par}% empty
2457     {\enspace\par\noindent\hspace{\LP@NL@titleindent}\parbox{\LP@NL@titlewidth}{\strut\LP
2458       \begin{tikzpicture}[LP@preset,scale=\LP@NL@scale]%
2459         \LP@drawbackground{1}{1}{\LP@NL@columns}{\LP@NL@rows}{\LP@NL@bgcolor}%
2460         \LP@drawgrid{1}{1}{\LP@NL@columns}{\LP@NL@rows}{1cm}%
2461 }%
2462 {%
2463   \end{tikzpicture}%
2464   \LP@drawcounter{\LP@NL@counterstyle}%
2465   \stepcounter{LP@puzzlecounter}%
2466 \end{minipage}%
2467 }%

2468 %
2469 % resuko environment and options
2470 %
2471 \newcommand*\LP@RSK@init@prefix{LP@RSK}%
2472 \newcommand*\LP@RSK@init@package{resuko}%
2473 %
2474 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{rows}{5}%
2475 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{columns}{5}%
2476 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{scale}{1}%
2477 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{counterstyle}{none}%
2478 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{bgcolor}{}%
2479 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{color}{blue}%
2480 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{width}{5.1cm}%
2481 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{cvmoffset}{-23pt}%
2482 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{title}{}%
2483 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{titleindent}{0cm}%
2484 \LP@define@key{\LP@RSK@init@prefix}{\LP@RSK@init@package}{titlewidth}{5.1cm}%
2485 \LP@define@choicekey@fontsize{\LP@RSK@init@prefix}{\LP@RSK@init@package}{Large}%
2486 %
2487 \let\resukocell\setcell%
2488 %
2489 \newcommand*\resukosetup[1]%
2490 {%
2491   \setkeys{resuko.sty}{#1}%
2492 }%

```

\LP@trackC For the resuko environment, we need to draw different tiles of track segments. First, we start with a curve from the bottom to the left.

```
2493 \newcommand*\LP@trackC%
```

```

2494 {%
2495   \LP@set@LP@scale{\LP@env@prefix}%
2496   \begin{tikzpicture}[scale=\LP@scale]%
2497     \pgfsetcornersarced{\pgfpoint{4mm*\LP@scale}{4mm*\LP@scale}}%
2498     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2499       (0,.5)--(.5,.5)--(.5,0);%
2500   \end{tikzpicture}%
2501 }%

```

`\LP@G@trackC` A generic command for drawing track curves with a rotation [$\langle angle \rangle$].

`\LP@G@trackC[$\langle angle \rangle$]`

```

2502 \newcommand*\LP@G@trackC[1][0]%
2503 {%
2504   \LP@set@LP@scale{\LP@env@prefix}%

```

We draw a TikZ picture and apply a rotation.

```

2505   \begin{tikzpicture}[scale=\LP@scale,rotate=#1]%

```

We want tiles fitting into a cell, so we clip the picture. Furthermore, we want rounded corners for the race track, of course!

```

2506     \clip (0,0) rectangle (1,1);%
2507     \pgfsetcornersarced{\pgfpoint{4mm*\LP@scale}{4mm*\LP@scale}}%

```

Then we draw a track curve from the bottom to the left. Rotation applies for the complete picture!

```

2508     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2509       (0,.5)--(.5,.5)--(.5,0);%
2510   \end{tikzpicture}%
2511 }%

```

`\CurveBL`

```

2512 \let\CurveBL\LP@G@trackC%

```

`\CurveBR`

```

2513 \newcommand*\CurveBR{\LP@G@trackC[90]}%

```

`\CurveTR`

```

2514 \newcommand*\CurveTR{\LP@G@trackC[180]}%

```

`\CurveTL`

```

2515 \newcommand*\CurveTL{\LP@G@trackC[270]}%

```

`\LP@trackS` Of course, we also need straights.

```

2516 \newcommand*\LP@trackS%
2517 {%
2518   \LP@set@LP@scale{\LP@env@prefix}%
2519   \begin{tikzpicture}[scale=\LP@scale]%
2520     \draw[draw opacity=0] (.2,0) -- (.8,1);%
2521     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2522       (.5,0)--(.5,1);%
2523   \end{tikzpicture}%
2524 }%

```

\LP@G@trackS We need horizontal and vertical straights, therefore we allow an rotate angle.

\LP@G@trackS[$\langle angle \rangle$]

```

2525 \newcommand*\LP@G@trackS[1][0]%
2526 {%
2527   \LP@set@LP@scale{\LP@env@prefix}%
2528   \begin{tikzpicture}[scale=\LP@scale,rotate=#1]%
2529     \clip (0,0) rectangle (1,1);%
2530     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2531       (.5,0)--(.5,1);%
2532   \end{tikzpicture}%
2533 }%

```

\StraightV

```
2534 \newcommand*\StraightV{\LP@G@trackS}%
```

\StraightH

```
2535 \newcommand*\StraightH{\LP@G@trackS[90]}%
```

\Straight

```
2536 \let\Straight\StraightV%
```

\LP@trackCR Crossings are needed as well!

```

2537 \newcommand*\LP@trackCR%
2538 {%
2539   \LP@set@LP@scale{\LP@env@prefix}%
2540   \begin{tikzpicture}[scale=\LP@scale]%
2541     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2542       (0,0.5)--(1,.5);%
2543     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2544       (0.5,0)--(.5,.2);%
2545     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2546       (0.5,1)--(.5,.8);%
2547   \end{tikzpicture}%
2548 }%

```


`\LP@G@trackCR` Same game again, we need also rotated versions.

```

2549 \newcommand*\LP@G@trackCR[1][0]%
2550 {%
2551   \LP@set@LP@scale{\LP@env@prefix}%
2552   \begin{tikzpicture}[scale=\LP@scale,rotate=#1]%
2553     \clip (0,0) rectangle (1,1);%
2554     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2555       (0,0.5)--(1,.5);%
2556     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2557       (0.5,0)--(.5,.2);%
2558     \draw[color=LP@c@track, line width=.3cm*\LP@scale]%
2559       (0.5,1)--(.5,.8);%
2560   \end{tikzpicture}%
2561 }%
2562 %
2563 \newcommand*\CrossH{\LP@G@trackCR}%
2564 \newcommand*\CrossV{\LP@G@trackCR[90]}%
2565 \let\Cross\CrossH%
```

`\LP@graveltrap` We also need a gravel trap.

```

2566 \newcommand*\LP@graveltrap%
2567 {%
2568   \LP@set@LP@scale{\LP@env@prefix}%
2569   \begin{tikzpicture}[scale=\LP@scale]%
2570     \clip (0,0) rectangle (1,1);%
2571     \fill[color=LP@c@track] (.5,.5) circle (.1cm);%
2572   \end{tikzpicture}%
2573 }%
2574 %
2575 \let\Graveltrap\LP@graveltrap%
```

`\parkinglot` And a parking lot!

```

2576 \newcommand*\parkinglot[2]%
2577 {%
2578   \LP@set@LP@scale{\LP@env@prefix}%
2579   \LP@set@LP@color{\LP@env@prefix}%
2580   \LP@set@LP@bgcolor{\LP@env@prefix}%
2581   \LP@set@LP@fontsize{\LP@env@prefix}%
2582   \ifthenelse{\equal{\LP@color}{}}{%
2583     {\gdef\LP@c@parkinglot{black}}%
2584     {\gdef\LP@c@parkinglot{\LP@color}}%
2585   \ifthenelse{\equal{\LP@bgcolor}{}}{%
2586     {\gdef\LP@cbg@parkinglot{white}}%
2587     {\gdef\LP@cbg@parkinglot{\LP@bgcolor}}%
2588   \setcounter{LP@counti}{#1}%
2589   \setcounter{LP@countii}{#2}%
2590   \stepcounter{LP@counti}%
2591   \stepcounter{LP@countii}%
2592   \draw[color=\LP@c@parkinglot,line width=\LP@normallines,%
```

```

2593         fill=\LP@cc@bg@parkinglot] (#1,#2) rectangle ++(2,2);%
2594 \node[color=\LP@cc@parkinglot,font=\sffamily\bfseries\LP@fontsize]%
2595     at (\arabic{LP@counti},\arabic{LP@countii}){P};%
2596 }%

```

\pitlane And finally, the pitlane!

```

2597 \newcommand*\pitlane[3]%
2598 {%
2599     \LP@set@LP@scale{\LP@env@prefix}%
2600     \LP@set@LP@color{\LP@env@prefix}%
2601     \ifthenelse{\equal{\LP@color}{}}{%
2602         {\gdef\LP@cc@pitlane{black}}%
2603         {\gdef\LP@cc@pitlane{\LP@color}}%
2604     \ifthenelse{\equal{#3}{V}}{%
2605         {\draw[color=black,line width=\LP@normallines,fill=\LP@cc@pitlane]%
2606             (#1,#2) rectangle ++(1,4);}%
2607         {\draw[color=black,line width=\LP@normallines,fill=\LP@cc@pitlane]%
2608             (#1,#2) rectangle ++(4,1);}%
2609     }%

```

\LP@trackline This macro provides the "box" placed next to the grid, showing the straights, curves and crossings, which are needed in that line!

\LP@trackline{<# straights>}{<# curves>}{<# crossings>}

```

2610 \newcommand*\LP@trackline[3]%
2611 {%
2612     \LP@set@LP@scale{\LP@env@prefix}%
2613     \pgfmathsetlength{\LP@length}{.1cm*\LP@scale}%
2614     \scalebox{\LP@tracks@scale}%
2615     {%
2616         \hspace{\LP@length}%
2617         \setcounter{LP@whiledo@i}{#1}%
2618         \whiledo{\arabic{LP@whiledo@i}>0}%
2619         {%
2620             \LP@trackS%
2621             \addtocounter{LP@whiledo@i}{-1}%
2622         }%
2623         \setcounter{LP@whiledo@i}{#2}%
2624         \whiledo{\arabic{LP@whiledo@i}>0}%
2625         {%
2626             \LP@trackC%
2627             \addtocounter{LP@whiledo@i}{-1}%
2628         }%
2629         \hspace{\LP@length}%
2630         \setcounter{LP@whiledo@i}{#3}%
2631         \whiledo{\arabic{LP@whiledo@i}>0}%
2632         {%
2633             \LP@trackCR%
2634             \addtocounter{LP@whiledo@i}{-1}%
2635         }%

```

```
2636 }%
2637 }%
```

`\trackV` This macro places the vertical track lines and expects a csv list in the format
straights/curves/crossings!

```
\trackV{<csv list>}
```

```
2638 \newcommand*\trackV[1]%
2639 {%
2640   \LP@set@LP@columns{\LP@env@prefix}%
2641   \setcounter{LP@counti}{\LP@columns}%
2642   \stepcounter{LP@counti}%
2643   \setcounter{LP@countii}{1}%
2644   \foreach \LP@c@straight/\LP@c@curve/\LP@c@cross in {#1}%
2645   {%
2646     \node[anchor=west] at (\arabic{LP@counti},\arabic{LP@countii}.5)%
2647       {\LP@trackline{\LP@c@straight}{\LP@c@curve}{\LP@c@cross}};%
2648     \stepcounter{LP@countii}%
2649   };%
2650 }%
```

`\trackH` The same for the horizontal track lines!

```
\trackH{<csv list>}
```

```
2651 \newcommand*\trackH[1]%
2652 {%
2653   \setcounter{LP@counti}{1}%
2654   \setcounter{LP@countii}{1}%
2655   \foreach \LP@c@straight/\LP@c@curve/\LP@c@cross in {#1}%
2656   {%
2657     \node[anchor=west,rotate=-90]%
2658       at (\arabic{LP@counti}.5,\arabic{LP@countii})%
2659       {\LP@trackline{\LP@c@straight}{\LP@c@curve}{\LP@c@cross}};%
2660     \stepcounter{LP@counti}%
2661   };%
2662 }%
```

`\track` We still need to draw the race track. The used decoration - which allows
automatic over-/underbridges - was provided by [Frédéric](#) in this [question](#)!

```
\track{<Tikz path>}
```

```
2663 \newcommand*\track[1]%
2664 {%
2665   \LP@set@LP@scale{\LP@env@prefix}%
2666   \LP@set@LP@bgcolor{\LP@env@prefix}%
2667   \ifthenelse{\equal{\LP@bgcolor}{}}{%
2668     {\gdef\LP@c@bg@track{white}}%
2669     {\gdef\LP@c@bg@track{\LP@bgcolor}}%
2670   }%
2671   \bgroup%
```

```

2671 \def\LP@rel@tikzpath{.5}%
2672 \begin{pgfonlayer}{LPbackgroundtwo}%
2673 \pgfsetcornersarced{\pgfpoint{4.9mm*\LP@scale}{4.9mm*\LP@scale}}%
2674 \draw[decorate,decoration={show path construction,%
2675   lineto code={%
2676     \draw [LP@c@bg@track,line width=0.15cm*\LP@scale,%
2677       double=LP@c@track,double distance=.3cm*\LP@scale]%
2678       (\tikzinputsegmentfirst) -- (\tikzinputsegmentlast);},%
2679   curveto code={%
2680     \draw [LP@c@track,line width=0.3cm*\LP@scale]%
2681       (\tikzinputsegmentfirst) .. controls%
2682       (\tikzinputsegmentssupporta) and (\tikzinputsegmentssupportb)%
2683       .. (\tikzinputsegmentlast);}]] #1;%
2684 \end{pgfonlayer}%
2685 \egroup%
2686 }%

```

resuko

```

2687 \newenvironment{resuko}[1][]%
2688 {%
2689   \setkeys{resuko}{#1}%
2690   \LP@set@package{resuko}%
2691   \LP@set@env@prefix{LP@RSK}%
2692   \setcounter{LP@rows}{\LP@RSK@rows}%
2693   \setcounter{LP@columns}{\LP@RSK@columns}%
2694   \stepcounter{LP@rows}%
2695   \stepcounter{LP@columns}%
2696   \begin{minipage}[t]{\LP@RSK@width}%
2697     \ifthenelse{\equal{\LP@RSK@title}{}}%
2698     {\par\enspace\par}% empty
2699     {\enspace\par\noindent\hspace{\LP@RSK@titleindent}\parbox{\LP@RSK@titlewidth}{\strut\
2700   \begin{tikzpicture}[LP@preset,scale=\LP@RSK@scale]%
2701     \LP@drawbackground{1}{1}{\LP@RSK@columns}{\LP@RSK@rows}{\LP@RSK@bgcolor}%
2702     \LP@drawgrid{1}{1}{\LP@RSK@columns}{\LP@RSK@rows}{1cm}%
2703   }%
2704   {%
2705     \end{tikzpicture}%
2706     \LP@drawcounter{\LP@RSK@counterstyle}%
2707     \stepcounter{LP@puzzlecounter}%
2708   \end{minipage}%
2709   }%

2710 %
2711 % schatzsuche environment and options
2712 %
2713 \newcommand*\LP@SS@init@prefix{LP@SS}%
2714 \newcommand*\LP@SS@init@package{schatzsuche}%
2715 %
2716 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{rows}{5}%
2717 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{columns}{5}%
2718 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{scale}{1}%

```

```

2719 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{counterstyle}{none}%
2720 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{bgcolor}{}%
2721 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{width}{5.1cm}%
2722 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{cvoffset}{-23pt}%
2723 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{title}{}%
2724 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{titleindent}{0cm}%
2725 \LP@define@key{\LP@SS@init@prefix}{\LP@SS@init@package}{titlewidth}{5.1cm}%
2726 \LP@define@choicekey@fontsize{\LP@SS@init@prefix}{\LP@SS@init@package}{Large}%
2727 %
2728 \let\schatzsuchecell\setcell%
2729 %
2730 \newcommand*\schatzsuchesetup[1]%
2731 {%
2732   \setkeys{schatzsuche.sty}{#1}%
2733 }%

```

`\LP@Diamond` For the schatzsuche environment, we need to draw diamonds.

```

2734 \newcommand*\LP@Diamond%
2735 {%
2736   \begin{tikzpicture}[scale=\LP@scale*.9]%
2737     \clip (.05,.905) rectangle (.95,.1);%
2738     \draw[line width=\LP@normallines,fill=yellow!20]%
2739       (.5,.1) -- (.7,.9) -- (.3,.9) -- cycle;%
2740     \draw[line width=\LP@normallines,fill=orange!20]%
2741       (.5,.1) -- (.7,.9) -- (.9,.8) -- cycle;%
2742     \draw[line width=\LP@normallines,fill=orange!20]%
2743       (.5,.1) -- (.3,.9) -- (.1,.8) -- cycle;%
2744     \draw[line width=\LP@normallines] (.1,.8) -- (.9,.8);%
2745 \end{tikzpicture}%
2746 }%

```

`\Diamond` A user command for drawing diamonds.

```
2747 \let\Diamond\LP@Diamond%
```

`schatzsuche`

```

2748 \newenvironment{schatzsuche}[1][]%
2749 {%
2750   \setkeys{schatzsuche}{#1}%
2751   \LP@set@package{schatzsuche}%
2752   \LP@set@env@prefix{\LP@SS}%
2753   \setcounter{LP@rows}{\LP@SS@rows}%
2754   \setcounter{LP@columns}{\LP@SS@columns}%
2755   \stepcounter{LP@rows}%
2756   \stepcounter{LP@columns}%
2757   \begin{minipage}[t]{\LP@SS@width}%
2758     \ifthenelse{\equal{\LP@SS@title}{}}{%
2759       {\par\enspace\par}% empty
2760       {\enspace\par\noindent\hspace{\LP@SS@titleindent}\parbox{\LP@SS@titlewidth}{\strut\LP

```

```

2761 \begin{tikzpicture}[LPpreset,scale=\LP@SS@scale]%
2762 \LP@drawbackground{1}{1}{\LP@SS@columns}{\LP@SS@rows}{\LP@SS@bgcolor}%
2763 \LP@drawgrid{1}{1}{\LP@SS@columns}{\LP@SS@rows}{1cm}%
2764 }%
2765 {%
2766 \end{tikzpicture}%
2767 \LP@drawcounter{\LP@SS@counterstyle}%
2768 \stepcounter{\LP@puzzlecounter}%
2769 \end{minipage}%
2770 }%

2771 %
2772 % skyline environment and options
2773 %
2774 \newcommand*\LP@SL@init@prefix{\LP@SL}%
2775 \newcommand*\LP@SL@init@package{skyline}%
2776 %
2777 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{rows}{5}%
2778 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{columns}{5}%
2779 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{scale}{1}%
2780 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{counterstyle}{none}%
2781 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{bgcolor}{}%
2782 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{width}{6.7cm}%
2783 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{cwoffset}{-38pt}%
2784 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{title}{}%
2785 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{titleindent}{0.75cm}%
2786 \LP@define@key{\LP@SL@init@prefix}{\LP@SL@init@package}{titlewidth}{5.85cm}%
2787 \LP@define@choicekey@fontsize{\LP@SL@init@prefix}{\LP@SL@init@package}{Large}%
2788 %
2789 \newcommand*\LP@SL@sudoku{false}%
2790 %
2791 \define@choicekey*{skyline.sty}{sudoku}[\LP@SL@sudoku\nr]{true,false}[true]%
2792 {%
2793 \ifcase\nr\relax%
2794 \renewcommand*\LP@SL@sudoku{true}%
2795 \renewcommand*\LP@SL@rows{9}%
2796 \renewcommand*\LP@SL@columns{9}%
2797 \or%
2798 \renewcommand*\LP@SL@sudoku{false}%
2799 \fi%
2800 }%
2801 %
2802 \define@choicekey*{skyline}{sudoku}[\LP@SL@sudoku\nr]{true,false}[true]%
2803 {%
2804 \ifcase\nr\relax%
2805 \renewcommand*\LP@SL@sudoku{true}%
2806 \renewcommand*\LP@SL@rows{9}%
2807 \renewcommand*\LP@SL@columns{9}%
2808 \or%
2809 \renewcommand*\LP@SL@sudoku{false}%
2810 \fi%
2811 }%

```

```

2812 %
2813 \let\skylineB\LP@bottomrow%
2814 \let\skylineT\LP@toprow%
2815 \let\skylineL\LP@leftcolumn%
2816 \let\skylineR\LP@rightcolumn%
2817 \let\skylinecell\setcell%
2818 %
2819 \newcommand*\skylinesetup[1]%
2820 {%
2821   \setkeys{skyline.sty}{#1}%
2822 }%

```

skyline

```

2823 \newenvironment{skyline}[1][1]%
2824 {%
2825   \setkeys{skyline}{#1}%
2826   \LP@set@package{skyline}%
2827   \LP@set@env@prefix{LP@SL}%
2828   \setcounter{LP@rows}{\LP@SL@rows}%
2829   \setcounter{LP@columns}{\LP@SL@columns}%
2830   \stepcounter{LP@rows}%
2831   \stepcounter{LP@columns}%
2832   \begin{minipage}[t]{\LP@SL@width}%
2833     \ifthenelse{\equal{\LP@SL@title}{}}{%
2834       {\par\enspace\par}% empty
2835     }{\enspace\par\noindent\hspace{\LP@SL@titleindent}\parbox{\LP@SL@titlewidth}{\strut\LP@SL@title}%
2836     \begin{tikzpicture}[LP@preset,scale=\LP@SL@scale]%
2837       \LP@drawbackground{1}{1}{\LP@SL@columns}{\LP@SL@rows}{\LP@SL@bgcolor}%
2838       \LP@drawgrid{1}{1}{\LP@SL@columns}{\LP@SL@rows}{1cm}%
2839       \ifthenelse{\equal{\LP@SL@sudoku}{true}}{%
2840         {\LP@drawsudokugrid}%
2841       }{}%
2842     }%
2843   }%
2844   \end{tikzpicture}%
2845   \LP@drawcounter{\LP@SL@counterstyle}%
2846   \stepcounter{LP@puzzlecounter}%
2847 \end{minipage}%
2848 }%

2849 %
2850 % slitherlink environment and options
2851 %
2852 \newcommand*\LP@SK@init@prefix{LP@SK}%
2853 \newcommand*\LP@SK@init@package{slitherlink}%
2854 %
2855 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{rows}{5}%
2856 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{columns}{5}%
2857 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{scale}{1}%
2858 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{counterstyle}{none}%
2859 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{bgcolor}{%

```

```

2860 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{width}{5.2cm}%
2861 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{cvoffset}{-23pt}%
2862 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{title}{}%
2863 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{titleindent}{0cm}%
2864 \LP@define@key{\LP@SK@init@prefix}{\LP@SK@init@package}{titlewidth}{5.2cm}%
2865 \LP@define@choicekey@fontsize{\LP@SK@init@prefix}{\LP@SK@init@package}{Large}%
2866 %
2867 \let\slitherlinkcell\setcell%
2868 %
2869 \newcommand*\slitherlinksetup[1]%
2870 {%
2871   \setkeys{slitherlink.sty}{#1}%
2872 }%

```

`\LP@drawgriddots` For the slitherlink environment, we need little dots in every cell corner. Therefore, we loop through all rows and columns and paint the dots on the LPforeground layer.

```

2873 \newcommand*\LP@drawgriddots%
2874 {%
2875   \pgfmathsetmacro{\START}{1}%

```

The dots are drawn immediately after the grid, so we can recycle LP@counti and LP@countii.

```

2876   \pgfmathsetmacro{\ENDC}{\arabic{LP@counti}}%
2877   \pgfmathsetmacro{\ENDR}{\arabic{LP@countii}}%
2878   \begin{pgfonlayer}{LPforeground}%
2879     \foreach \i in {\START,...,\ENDC}%
2880       \foreach \j in {\START,...,\ENDR}%
2881         \fill[color=LP@cc@griddots] (\i,\j) circle [radius=3pt];%
2882   \end{pgfonlayer}%
2883 }%

```

slitherlink

```

2884 \newenvironment{slitherlink}[1][]%
2885 {%
2886   \setkeys{slitherlink}{#1}%
2887   \LP@set@package{slitherlink}%
2888   \LP@set@env@prefix{\LP@SK}%
2889   \ifthenelse{\equal{\LP@grid@linestyle}{} }%
2890     {\setgridlinestyle{dashed}}{}%
2891   \setcounter{LP@rows}{\LP@SK@rows}%
2892   \setcounter{LP@columns}{\LP@SK@columns}%
2893   \stepcounter{LP@rows}%
2894   \stepcounter{LP@columns}%
2895   \begin{minipage}[t]{\LP@SK@width}%
2896     \ifthenelse{\equal{\LP@SK@title}{} }%
2897       {\par\enspace\par}% empty
2898       {\enspace\par\noindent\hspace{\LP@SK@titleindent}\parbox{\LP@SK@titlewidth}{\strut\LP
2899       \begin{tikzpicture}[LPpreset,scale=\LP@SK@scale]%

```



```

2900      \LP@drawbackground{1}{1}{\LP@SK@columns}{\LP@SK@rows}{\LP@SK@bgcolor}%
2901      \LP@drawgrid{1}{1}{\LP@SK@columns}{\LP@SK@rows}{1cm}%
2902      \LP@drawgriddots%
2903 }%
2904 {%
2905     \end{tikzpicture}%
2906     \LP@drawcounter{\LP@SK@counterstyle}%
2907     \stepcounter{LP@puzzlecounter}%
2908 \end{minipage}%
2909 }%

2910 %
2911 % starbattle environment and options
2912 %
2913 \newcommand*\LP@SB@init@prefix{LP@SB}%
2914 \newcommand*\LP@SB@init@package{starbattle}%
2915 %
2916 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{rows}{5}%
2917 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{columns}{5}%
2918 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{scale}{1}%
2919 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{counterstyle}{none}%
2920 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{bgcolor}{}%
2921 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{width}{5.1cm}%
2922 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{cwoffset}{-23pt}%
2923 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{title}{}%
2924 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{titleindent}{0cm}%
2925 \LP@define@key{\LP@SB@init@prefix}{\LP@SB@init@package}{titlewidth}{5.1cm}%
2926 \LP@define@choicekey@fontsize{\LP@SB@init@prefix}{\LP@SB@init@package}{Large}%
2927 %
2928 \let\starbattlecell\setcell%
2929 %
2930 \newcommand*\starbattlesetup[1]%
2931 {%
2932     \setkeys{starbattle.sty}{#1}%
2933 }%

```

starbattle

```

2934 \newenvironment{starbattle}[1][]%
2935 {%
2936     \setkeys{starbattle}{#1}%
2937     \LP@set@package{starbattle}%
2938     \LP@set@env@prefix{LP@SB}%
2939     \setcounter{LP@rows}{\LP@SB@rows}%
2940     \setcounter{LP@columns}{\LP@SB@columns}%
2941     \stepcounter{LP@rows}%
2942     \stepcounter{LP@columns}%
2943     \begin{minipage}[t]{\LP@SB@width}%
2944         \ifthenelse{\equal{\LP@SB@title}{}}{%
2945             {\par\enspace\par}% empty
2946             {\enspace\par\noindent\hspace{\LP@SB@titleindent}\parbox{\LP@SB@titlewidth}{\strut\LP
2947         \begin{tikzpicture}[LPpreset,scale=\LP@SB@scale]%

```

```

2948      \LP@drawbackground{1}{1}{\LP@SB@columns}{\LP@SB@rows}{\LP@SB@bgcolor}%
2949      \LP@drawgrid{1}{1}{\LP@SB@columns}{\LP@SB@rows}{1cm}%
2950 }%
2951 {%
2952   \end{tikzpicture}%
2953   \LP@drawcounter{\LP@SB@counterstyle}%
2954   \stepcounter{\LP@puzzlecounter}%
2955   \end{minipage}%
2956 }%

2957 %
2958 % starsandarrows environment and options
2959 %
2960 \newcommand*\LP@SAA@init@prefix{\LP@SAA}%
2961 \newcommand*\LP@SAA@init@package{starsandarrows}%
2962 %
2963 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{rows}{5}%
2964 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{columns}{5}%
2965 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{scale}{1}%
2966 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{counterstyle}{none}%
2967 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{bgcolor}{}%
2968 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{width}{5.9cm}%
2969 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{cvoffset}{-23pt}%
2970 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{title}{}%
2971 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{titleindent}{0cm}%
2972 \LP@define@key{\LP@SAA@init@prefix}{\LP@SAA@init@package}{titlewidth}{5.9cm}%
2973 \LP@define@choicekey@fontsize{\LP@SAA@init@prefix}{\LP@SAA@init@package}{Large}%
2974 %
2975 \let\starsH\LP@toprow%
2976 \let\starsV\LP@leftcolumn%
2977 %
2978 \newcommand*\starsandarrowssetup[1]%
2979 {%
2980   \setkeys{starsandarrows.sty}{#1}%
2981 }%

```

`\LP@Star` For several environments, we need to draw stars.

```

2982 \newcommand*\LP@Star%
2983 {%
2984   \tikz\node[shape=star,fill=yellow,draw,scale=.8*\LP@scale,%
2985             star point height=.25cm] {};%
2986 }%

```

A user command for drawing stars.

```

2987 \let\Star\LP@Star%

```

`\LP@ArrowA` A generic macro for drawing arrows with different colors and angles.

```

\LP@ArrowA[⟨color⟩]{⟨angle⟩}

```

```

2988 \newcommand*\LP@ArrowA[2][red]%
2989 {%
2990   \LP@set@LP@scale{\LP@env@prefix}%
2991   \pgfmathsetmacro{\LP@linewidth}{3pt*\LP@scale}%
2992   \tikz\draw[->,line width=\LP@linewidth,rotate=#2,%
2993             scale=\LP@scale,color=#1]%
2994             (0.1,.5) -- (0.9,.5);%
2995 }%

```

The we define some arrows with standard angles.

```

2996 \newcommand*\LP@c@stararrow{red}%

```

\Right

```

2997 \newcommand*\Right{\LP@ArrowA[\LP@c@stararrow]{0}}%

```

\RightUp

```

2998 \newcommand*\RightUp{\LP@ArrowA[\LP@c@stararrow]{45}}%

```

\Up

```

2999 \newcommand*\Up{\LP@ArrowA[\LP@c@stararrow]{90}}%

```

\LeftUp

```

3000 \newcommand*\LeftUp{\LP@ArrowA[\LP@c@stararrow]{135}}%

```

\Left

```

3001 \newcommand*\Left{\LP@ArrowA[\LP@c@stararrow]{180}}%

```

\LeftDown

```

3002 \newcommand*\LeftDown{\LP@ArrowA[\LP@c@stararrow]{225}}%

```

\Down

```

3003 \newcommand*\Down{\LP@ArrowA[\LP@c@stararrow]{270}}%

```

\RightDown

```

3004 \newcommand*\RightDown{\LP@ArrowA[\LP@c@stararrow]{315}}%

```

starsandarrows

```

3005 \newenvironment{starsandarrows}[1][]%
3006 {%
3007   \setkeys{starsandarrows}{#1}%
3008   \LP@set@package{starsandarrows}%

```

```

3009 \LP@set@env@prefix{LP@SAA}%
3010 \setcounter{LP@rows}{\LP@SAA@rows}%
3011 \setcounter{LP@columns}{\LP@SAA@columns}%
3012 \stepcounter{LP@rows}%
3013 \stepcounter{LP@columns}%
3014 \begin{minipage}[t]{\LP@SAA@width}%
3015 \ifthenelse{\equal{\LP@SAA@title}{}}{%
3016 \par\enspace\par}% empty
3017 {\enspace\par\noindent\hspace{\LP@SAA@titleindent}\parbox{\LP@SAA@titlewidth}{\strut\
3018 \begin{tikzpicture}[LP@preset,scale=\LP@SAA@scale]%
3019 \LP@drawbackground{1}{1}{\LP@SAA@columns}{\LP@SAA@rows}{\LP@SAA@bgcolor}%
3020 \LP@drawgrid{1}{1}{\LP@SAA@columns}{\LP@SAA@rows}{1cm}%
3021 }%
3022 {%
3023 \end{tikzpicture}%
3024 \LP@drawcounter{\LP@SAA@counterstyle}%
3025 \stepcounter{LP@puzzlecounter}%
3026 \end{minipage}%
3027 }%

3028 %
3029 % sunandmoon environment and options
3030 %
3031 \newcommand*\LP@SAM@init@prefix{LP@SAM}%
3032 \newcommand*\LP@SAM@init@package{sunandmoon}%
3033 %
3034 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{rows}{5}%
3035 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{columns}{5}%
3036 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{scale}{1}%
3037 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{counterstyle}{none}%
3038 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{bgcolor}{}%
3039 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{width}{5.1cm}%
3040 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{cvmoffset}{-23pt}%
3041 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{title}{}%
3042 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{titleindent}{0cm}%
3043 \LP@define@key{\LP@SAM@init@prefix}{\LP@SAM@init@package}{titlewidth}{5.1cm}%
3044 \LP@define@choicekey@fontsize{\LP@SAM@init@prefix}{\LP@SAM@init@package}{Large}%
3045 %
3046 \let\sunandmooncell\setcell%
3047 %
3048 \newcommand*\sunandmoonsetup[1]%
3049 {%
3050 \setkeys{sunandmoon.sty}{#1}%
3051 }%

```

`\LP@DarkCloud` For the sunandmoon environment, we need to draw a dark cloud and differently lighted moons.

```

3052 \newcommand*\LP@DarkCloud{%
3053 \tikz[scale=0.7*\LP@scale]\fill[color=black!60] (0,0)%
3054 rectangle (1,1);%
3055 }%

```

\LP@Moon

```
3056 \newcommand*\LP@Moon{%
3057   \tikz[scale=0.7*\LP@scale]\fill[color=black!60]%
3058     (0,0) circle (0.5cm);%
3059 }%
```

\LP@MoonR

```
3060 \newcommand*\LP@MoonR%
3061 {%
3062   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3063     \fill[color=black!60] (0.5,0.5) circle (0.5cm);%
3064     \fill[color=yellow] (.5,0) arc (270:450:.5);%
3065   \end{tikzpicture}%
3066 }%
```

\LP@MoonL

```
3067 \newcommand*\LP@MoonL%
3068 {%
3069   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3070     \fill[color=black!60] (0.5,0.5) circle (0.5cm);%
3071     \fill[color=yellow] (.5,0) arc (270:90:.5);%
3072   \end{tikzpicture}%
3073 }%
```

\LP@MoonT

```
3074 \newcommand*\LP@MoonT%
3075 {%
3076   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3077     \fill[color=black!60] (0.5,0.5) circle (0.5cm);%
3078     \fill[color=yellow] (0,0.5) arc (180:0:.5);%
3079   \end{tikzpicture}%
3080 }%
```

\LP@MoonB

```
3081 \newcommand*\LP@MoonB%
3082 {%
3083   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3084     \fill[color=black!60] (0.5,0.5) circle (0.5cm);%
3085     \fill[color=yellow] (0,0.5) arc (180:360:.5);%
3086   \end{tikzpicture}%
3087 }%
```

\LP@MoonTL

```
3088 \newcommand*\LP@MoonTL%
3089 {%
```

```

3090 \begin{tikzpicture}[scale=0.7*\LP@scale]%
3091   \fill[color=yellow] (0.5,0.5) circle (0.5cm);%
3092   \fill[color=black!60] (1,.5) -- (.5,.5) -- (.5,0)%
3093                       arc (270:360:.5);%
3094 \end{tikzpicture}%
3095 }%

```

\LP@MoonBL

```

3096 \newcommand*\LP@MoonBL%
3097 {%
3098   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3099     \fill[color=yellow] (0.5,0.5) circle (0.5cm);%
3100     \fill[color=black!60] (1,.5) -- (.5,.5) -- (.5,1)%
3101                       arc (90:0:.5);%
3102   \end{tikzpicture}%
3103 }%

```

\LP@MoonBR

```

3104 \newcommand*\LP@MoonBR%
3105 {%
3106   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3107     \fill[color=yellow] (0.5,0.5) circle (0.5cm);%
3108     \fill[color=black!60] (0,.5) -- (.5,.5) -- (.5,1)%
3109                       arc (90:180:.5);%
3110   \end{tikzpicture}%
3111 }%

```

\LP@MoonTR

```

3112 \newcommand*\LP@MoonTR%
3113 {%
3114   \begin{tikzpicture}[scale=0.7*\LP@scale]%
3115     \fill[color=yellow] (0.5,0.5) circle (0.5cm);%
3116     \fill[color=black!60] (0,.5) -- (.5,.5) -- (.5,0)%
3117                       arc (270:180:.5);%
3118   \end{tikzpicture}%
3119 }%

```

\Cloud

```

3120 \let\Cloud\LP@DarkCloud%

```

\Moon

```

3121 \let\Moon\LP@Moon%

```

\MoonR

```

3122 \let\MoonR\LP@MoonR%

```

\MoonL

```
3123 \let\MoonL\LP@MoonL%
```

\MoonT

```
3124 \let\MoonT\LP@MoonT%
```

\MoonB

```
3125 \let\MoonB\LP@MoonB%
```

\MoonTL

```
3126 \let\MoonTL\LP@MoonTL%
```

\MoonBL

```
3127 \let\MoonBL\LP@MoonBL%
```

\MoonBR

```
3128 \let\MoonBR\LP@MoonBR%
```

\MoonTR

```
3129 \let\MoonTR\LP@MoonTR%
```

sunandmoon

```
3130 \newenvironment{sunandmoon}[1][ ]%
```

```
3131 {%
```

```
3132 \setkeys{sunandmoon}{#1}%
```

```
3133 \LP@set@package{sunandmoon}%
```

```
3134 \LP@set@env@prefix{LP@SAM}%
```

```
3135 \setcounter{LP@rows}{\LP@SAM@rows}%
```

```
3136 \setcounter{LP@columns}{\LP@SAM@columns}%
```

```
3137 \stepcounter{LP@rows}%
```

```
3138 \stepcounter{LP@columns}%
```

```
3139 \begin{minipage}[t]{\LP@SAM@width}%
```

```
3140 \ifthenelse{\equal{\LP@SAM@title}{}}{%
```

```
3141 {\par\enspace\par}% empty
```

```
3142 {\enspace\par\noindent\hspace{\LP@SAM@titleindent}\parbox{\LP@SAM@titlewidth}{\strut\
```

```
3143 \begin{tikzpicture}[LP@preset,scale=\LP@SAM@scale]%
```

```
3144 \LP@drawbackground{1}{1}{\LP@SAM@columns}{\LP@SAM@rows}{\LP@SAM@bgcolor}%
```

```
3145 \LP@drawgrid{1}{1}{\LP@SAM@columns}{\LP@SAM@rows}{1cm}%
```

```
3146 }%
```

```
3147 {%
```

```
3148 \end{tikzpicture}%
```

```
3149 \LP@drawcounter{LP@SAM@counterstyle}%
```

```
3150 \stepcounter{LP@puzzlecounter}%
```

```

3151 \end{minipage}%
3152 }%

3153 %
3154 % tentsandtrees environment and options
3155 %
3156 \newcommand*\LP@TAT@init@prefix{\LP@TAT}%
3157 \newcommand*\LP@TAT@init@package{tentsandtrees}%
3158 %
3159 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{rows}{5}%
3160 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{columns}{5}%
3161 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{scale}{1}%
3162 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{counterstyle}{none}%
3163 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{bgcolor}{}%
3164 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{width}{5.9cm}%
3165 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{cvmoffset}{-23pt}%
3166 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{title}{}%
3167 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{titleindent}{0cm}%
3168 \LP@define@key{\LP@TAT@init@prefix}{\LP@TAT@init@package}{titlewidth}{5.9cm}%
3169 \LP@define@choicekey@fontsize{\LP@TAT@init@prefix}{\LP@TAT@init@package}{Large}%
3170 %
3171 \let\tentH\LP@toprow%
3172 \let\tentV\LP@leftcolumn%
3173 %
3174 \newcommand*\tentsandtreessetup[1]%
3175 {%
3176 \setkeys{tentsandtrees.sty}{#1}%
3177 }%

```

`\LP@Tree` For the tentsandtree environments, we need to draw trees.

```

3178 \newcommand*\LP@Tree%
3179 {%
3180 \begin{tikzpicture}[scale=.07*\LP@scale]%
3181 \draw[fill=brown!95!black!95,line width=.1pt*\LP@scale]%
3182 (.75,-1) .. controls (.5,.5) and (.5,3) .. (0.5,4) --%
3183 (-0.5,4) .. controls (-.5,3) and (-.5,.5) .. (-.75,-1);%
3184 \shade[top color=green!50!black!60,draw=black,%
3185 line width=.1pt*\LP@scale,bottom color=green!50!black]%
3186 (0,10) .. controls (0,8) and (1,7) .. (1.5,7)%
3187 .. controls (1,7) and (1,7) .. (0.5,7.25)%
3188 .. controls (1.5,5) and (2.5,4) .. (3,4)%
3189 .. controls (2,4) and (1.25,4) .. (1,4.5)%
3190 .. controls (2,2) and (3.5,2) .. (4,2)%
3191 .. controls (1,1) and (-1,1) .. (-4,2)%
3192 .. controls (-3.5,2) and (-2,2) .. (-1,4.5)%
3193 .. controls (-1.25,4) and (-2,4) .. (-3,4)%
3194 .. controls (-2.5,4) and (-1.5,5) .. (-0.5,7.25)%
3195 .. controls (-1,7) and (-1,7) .. (-1.5,7)%
3196 .. controls (-1,7) and (0,8) .. (0,10);%
3197 \end{tikzpicture}%
3198 }%

```


`\Tree` A user command for drawing a tree.

```
3199 \let\Tree\LP@Tree%
```

`\LP@Tent` We also need tents.

```
3200 \newcommand*\LP@Tent%
3201 {%
3202   \begin{tikzpicture}[scale=\LP@scale]%
3203     \draw[fill=yellow!50,line width=.1pt*\LP@scale]%
3204       (.1,.1) -- (.6,.1) -- (.9,.5) -- (.75,.8) -- (.35,.6)%
3205       -- cycle;%
3206     \draw[line width=.1pt*\LP@scale] (.35,.6) -- (.35,.1);%
3207     \draw[line width=.1pt*\LP@scale] (.35,.6) -- (.6,.1);%
3208   \end{tikzpicture}%
3209 }%
```

`\Tent` A user command for drawing tents.

```
3210 \let\Tent\LP@Tent%
```

tentsandtrees

```
3211 \newenvironment{tentsandtrees}[1][]%
3212 {%
3213   \setkeys{tentsandtrees}{#1}%
3214   \LP@set@package{tentsandtrees}%
3215   \LP@set@env@prefix{LP@TAT}%
3216   \setcounter{LP@rows}{\LP@TAT@rows}%
3217   \setcounter{LP@columns}{\LP@TAT@columns}%
3218   \stepcounter{LP@rows}%
3219   \stepcounter{LP@columns}%
3220   \begin{minipage}[t]{\LP@TAT@width}%
3221     \ifthenelse{\equal{\LP@TAT@title}{}}{%
3222       {\par\enspace\par}% empty
3223       {\enspace\par\noindent\hspace{\LP@TAT@titleindent}\parbox{\LP@TAT@titlewidth}{\strut\LP@drawbackground{1}{1}{\LP@TAT@columns}{\LP@TAT@rows}{\LP@TAT@bgcolor}%
3224         \LP@drawgrid{1}{1}{\LP@TAT@columns}{\LP@TAT@rows}{1cm}%
3225       }%
3226     }%
3227   }%
3228   \end{tikzpicture}%
3229   \LP@drawcounter{\LP@TAT@counterstyle}%
3230   \stepcounter{LP@puzzlecounter}%
3231   \end{minipage}%
3232 }%

3234 %
3235 % tunnel environment and options
3236 %
3237 \newcommand*\LP@TN@init@prefix{LP@TN}%
3238 \newcommand*\LP@TN@init@package{tunnel}%
```

```

3239 %
3240 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{rows}{5}%
3241 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{columns}{5}%
3242 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{scale}{1}%
3243 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{counterstyle}{none}%
3244 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{bgcolor}{}%
3245 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{width}{5.9cm}%
3246 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{cvooffset}{-23pt}%
3247 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{title}{}%
3248 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{titleindent}{0cm}%
3249 \LP@define@key{\LP@TN@init@prefix}{\LP@TN@init@package}{titlewidth}{5.9cm}%
3250 \LP@define@choicekey@fontsize{\LP@TN@init@prefix}{\LP@TN@init@package}{Large}%
3251 %
3252 \let\tunnelH\LP@toprow%
3253 \let\tunnelV\LP@leftcolumn%
3254 %
3255 \newcommand*\tunnelsetup[1]%
3256 {%
3257   \setkeys{tunnel.sty}{#1}%
3258 }%

```

`\LP@Portal` We define a TikZ picture that symbolizes a tunnel portal.

```

3259 \newcommand*\LP@Portal%
3260 {%
3261   \LP@set@LP@scale{\LP@env@prefix}%
3262   \begin{tikzpicture}[scale=\LP@scale]%
3263     \fill[color=black]%
3264       (0,0) -- (0,1) -- (.4,.5) -- (.4,0) -- cycle;%
3265     \fill[color=\LP@c@tunnel@ii]%
3266       (0,1) -- (1,1) -- (1,.5) -- (.4,.5) -- cycle;%
3267     \fill[color=\LP@c@tunnel@i]%
3268       (.4,0) rectangle (1,.5);%
3269     \draw[line width=\LP@normallines] (0,0) rectangle (1,1) ;%
3270   \end{tikzpicture}%
3271 }%

```

`\portal` We define a user command to set a tunnel portal into a grid cell.

`\portal{<column>}{<row>}`

```

3272 \newcommand*\portal[2]%
3273 {%

```

To avoid interference with the grid lines, we use the `LPbackground` layer.

```

3274 \begin{puzzlebackground}%
3275   \LP@G@setcellcontent[hcenter,vcenter]{#1}{#2}{\LP@Portal}%
3276 \end{puzzlebackground}%
3277 }%

```

`\tube` For the tunnel environments, we need to draw tubes.

```
\tube{<TikZ path>}
```

```
3278 \newcommand*\tube[1]%
3279 {%
3280   \LP@set@LP@scale{\LP@env@prefix}%
3281   \bgroup%
```

Tubes start in the center of a grid cell, therefore we must redefine `\LP@rel@tikzpath` to `.5` inside a group for `\tikzpath` and `\xtikzpath`.

```
3282   \def\LP@rel@tikzpath{.5}%
3283   \end{macrocode}
3284 % We draw on the \layer{LPbackgroundtwo} layer to get a segmented look behind
3285 % the grid lines.
3286 % \begin{macrocode}
3287   \begin{pgfonlayer}{LPbackgroundtwo}%
3288   \pgfsetcornersarced{\pgfpoint{3mm}{3mm}}%
```

We draw the tube several times with slightly different colors to get a 3D effect.

```
3289   \draw[color=LP@c@tunnel!80!black, line width=.4cm*\LP@scale]%
3290     #1;%
3291   \draw[color=LP@c@tunnel, line width=.38cm*\LP@scale] #1;%
3292   \draw[color=LP@c@tunnel!85, line width=.35cm*\LP@scale] #1;%
3293   \draw[color=LP@c@tunnel!70, line width=.32cm*\LP@scale] #1;%
3294   \draw[color=LP@c@tunnel!55, line width=.29cm*\LP@scale] #1;%
3295   \draw[color=LP@c@tunnel!45, line width=.25cm*\LP@scale] #1;%
3296   \draw[color=LP@c@tunnel!35, line width=.2cm*\LP@scale] #1;%
3297   \end{pgfonlayer}%
3298 \egroup%
3299 }%
```

tunnel

```
3300 \newenvironment{tunnel}[1][]%
3301 {%
3302   \setkeys{tunnel}{#1}%
3303   \LP@set@package{tunnel}%
3304   \LP@set@env@prefix{LP@TN}%
3305   \setcounter{LP@rows}{\LP@TN@rows}%
3306   \setcounter{LP@columns}{\LP@TN@columns}%
3307   \stepcounter{LP@rows}%
3308   \stepcounter{LP@columns}%
3309   \begin{minipage}[t]{\LP@TN@width}%
3310     \ifthenelse{\equal{\LP@TN@title}{}}{%
3311       {\par\enspace\par}% empty
3312     }{\enspace\par\noindent\hspace{\LP@TN@titleindent}\parbox{\LP@TN@titlewidth}{\strut\LP@
3313       \begin{tikzpicture}[LP@preset,scale=\LP@TN@scale]%
3314         \LP@drawbackground{1}{1}{\LP@TN@columns}{\LP@TN@rows}{\LP@TN@bgcolor}%
3315         \LP@drawgrid{1}{1}{\LP@TN@columns}{\LP@TN@rows}{1cm}%
3316       }%
3317     }%
3318   \end{tikzpicture}%
```

```
3319 \LP@drawcounter{\LP@TN@counterstyle}%  
3320 \stepcounter{LP@puzzlecounter}%  
3321 \end{minipage}%  
3322 }%  
  
3323 </lpenv>
```

5 References

- [1] Bernhard Hobiger. HoDoKu, 2012.
(<http://hodoku.sourceforge.net/en/index.php>).
- [2] Josef Kleber. createlopsudoku, 2013.
(<http://logicpuzzle.square7.de/createlopsudoku>).
- [3] Josef Kleber. Examples & Solutions (battleship), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-battleship.pdf>).
- [4] Josef Kleber. Examples & Solutions (bokkusu), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-bokkusu.pdf>).
- [5] Josef Kleber. Examples & Solutions (bridges), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-bridges.pdf>).
- [6] Josef Kleber. Examples & Solutions (chaossudoku), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-chaossudoku.pdf>).
- [7] Josef Kleber. Examples & Solutions (ddsudoku), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-ddsudoku.pdf>).
- [8] Josef Kleber. Examples & Solutions (fourwinds), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-fourwinds.pdf>).
- [9] Josef Kleber. Examples & Solutions (hakyuu), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-hakyuu.pdf>).
- [10] Josef Kleber. Examples & Solutions (hitori), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-hitori.pdf>).
- [11] Josef Kleber. Examples & Solutions (kakuro), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-kakuro.pdf>).
- [12] Josef Kleber. Examples & Solutions (kendoku), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-kendoku.pdf>).
- [13] Josef Kleber. Examples & Solutions (killersudoku), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-killersudoku.pdf>).

- [14] Josef Kleber. Examples & Solutions (laserbeam), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-laserbeam.pdf>).
- [15] Josef Kleber. Examples & Solutions (lpsudoku), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-lpsudoku.pdf>).
- [16] Josef Kleber. Examples & Solutions (magiclabyrinth), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-magiclabyrinth.pdf>).
- [17] Josef Kleber. Examples & Solutions (magnets), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-magnets.pdf>).
- [18] Josef Kleber. Examples & Solutions (masyu), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-masyu.pdf>).
- [19] Josef Kleber. Examples & Solutions (minesweeper), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-minesweeper.pdf>).
- [20] Josef Kleber. Examples & Solutions (numberlink), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-numberlink.pdf>).
- [21] Josef Kleber. Examples & Solutions (schatzsuche), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-schatzsuche.pdf>).
- [22] Josef Kleber. Examples & Solutions (skyline), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-skyline.pdf>).
- [23] Josef Kleber. Examples & Solutions (slitherlink), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-slitherlink.pdf>).
- [24] Josef Kleber. Examples & Solutions (sunandmoon), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-sunandmoon.pdf>).
- [25] Josef Kleber. Examples & Solutions (tentsandtrees), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-tentsandtrees.pdf>).
- [26] Josef Kleber. Examples & Solutions (tunnel), 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-tunnel.pdf>).

- [27] Josef Kleber. How to create a Sudoku magazine, 2013.
(http://logicpuzzle.square7.de/sudoku_magazine/).
- [28] Josef Kleber. logicpuzzle – A LaTeX style file for typesetting logic puzzles, 2013. (<http://logicpuzzle.square7.de/>).
- [29] Josef Kleber. The logicpuzzle bundle, 2013.
(<https://bitbucket.org/kleberj/logicpuzzle/>).
- [30] Josef Kleber. lpsmag, 2013. (<http://logicpuzzle.square7.de/lpsmag>).
- [31] Josef Kleber. Examples & Solutions (nonogram), 2014.
(<https://bitbucket.org/kleberj/logicpuzzle/downloads/example-nonogram.pdf>).
- [32] Stephen Ostermiller. QQwing – Sudoku Generator and Solver, 2011.
(<http://ostermiller.org/qqwing/>).

6 Change History

v1.0	General: first version of battleship.sty on CTAN 72	v1.5	General: added support for hakyuu puzzle 72
v1.1	General: added \placeisland .. 99		added support for skyline and chaossudoku puzzles 72
	added \placewater 99	v1.6	General: added support for lpsudoku puzzle 72
	added \battleshipsetup for resetting global options 100	v1.7	General: added support for ddsudoku puzzle 72
	implemented \placeship; placement of a complete ship 99		added support for hitori puzzle 72
	implemented \BS@ingrid checks if coordinate is in the grid → PackageError 77	v1.8	General: added support for killersudoku and kendokut puzzles 72
	renamed \ship → \placesegment 98		\colorarea: added \colorarea (\framearea without frame) . 86
v1.2	General: added \classicgame for drawing game sheets for classic Battleship 100	v1.9	General: added support for laserbeam puzzle 72
	added option bgcolor 97		added support for slitherlink puzzle 72
	added option counterstyle .. 97	v2.0	General: added support for minesweeper puzzle 72
	added option title 97		added support for tunnel puzzle 72
	moved code base for logic puzzles into logicpuzzle.sty 72		moved code from packages to logicpuzzle.sty 72
v1.3	General: added option cvooffset .. 97	v2.1	General: added support for kakuro puzzle 72
	added support for bokkusu puzzle 72		added support for schatzsuche puzzle 72
	added support for bokkusu puzzle 101		added support for tentsandtrees puzzle 72
	\fillcell: added \LP@ingrid to \fillcell 84	v2.2	General: added support for bridges puzzle 72
	\LP@setcolumncontents: Bugfix: mix up column and row 80		added support for starbattle puzzle 72
	\LP@setrowcontents: Bugfix: mix up column and row 79		added support for sunandmoon and starsandarrows puzzles 72
v1.4	General: added support for skyline puzzle 72		reduced counters to LP@rows and LP@columns instead of LP@XX@... 73
	\fillarea: added \fillarea ... 85		
	\framearea: added \framearea . 85		
	\setcell: added \setcell 81		
	\setcolumn: added \setcolumn . 81		
	\setrow: added \setrow 81		
	puzzlebackground: added puzzlebackground 89		
	puzzleforeground: added puzzleforeground 89		

v2.3		puzzles 72
General: added support for		added support for magnets puzzle 72
fourwinds puzzle 72		
added support for numberlink		\setcells: added \setcells ... 82
puzzle 72	v2.5	
added support for resuko puzzle 72	General: added LPpreset TikZ	style to all puzzle environments 72
\framepuzzle: Bugfix: \framepuzzle	added support for nonogram	puzzle. Feature request by:
only worked correctly for	Theresa Spannbaier 72	
quadratic puzzles 88	\LP@drawgriddots: removed un-	necessary scale update 144
v2.4		
General: added support for		
magiclabyrinth and masyu		

Symbols

\@Ship 852, 861
 \@ShipB 855
 \@ShipC 853, 863, 864, 865, 866,
 867, 868, 869, 870
 \@ShipL 856, 862, 863, 864, 865,
 866, 867, 868, 869, 870
 \@ShipR 857, 862, 863, 864, 865,
 866, 867, 868, 869, 870
 \@ShipT 854

B

battleship (environment) 1001
 \battleshipsetup 13, 968
 bgcolor (Puzzle option) 9, 24, 25,
 27, 28, 30, 32, 34, 35, 37,
 39, 41, 43, 45, 46, 48, 49,
 51, 53, 54, 56, 60, 62, 63,
 65, 66, 69, 70, 72
 \Black 15, 1544
 bokkusu (environment) ... 1054
 bokkusu.sty (Package) 6
 \bokkususetup 13, 1050
 \bridge 14, 1142
 bridges (environment) ... 1146
 \bridgescell 1097
 \bridgescolumn 14, 1122
 \bridgesrow 14, 1103
 \bridgessetup 14, 1099

C

chaossudoku (environment) 1204
 \chaossudokucell 14, 1198
 \chaossudokusetup ... 14, 1200
 \classicgame 13, 973
 \Cloud 21, 3120
 color (Puzzle option) 9, 10, 27, 28,
 32, 37, 48, 51, 53, 54
 \colorarea 10, 449
 columns (Puzzle option) 7–10, 23,
 25, 27, 28, 30, 32, 34, 35,
 37, 39, 41, 43, 44, 46, 47,
 49, 50, 52, 54, 56, 60, 61,
 63, 64, 69, 70, 72
 counterstyle (Puzzle option) . 9,
 24, 25, 27, 29, 30, 32, 34,
 36, 37, 40, 42, 43, 45, 46,

48, 49, 51, 53, 54, 56, 60,
 62, 63, 65, 66, 69, 70, 72
 \Cross 19, 2565
 \CrossH 19, 2563, 2565
 \CrossV 19, 2564
 \CurveBL 19, 2512
 \CurveBR 19, 2513
 \CurveTL 19, 2515
 \CurveTR 19, 2514
 cvoffset (Puzzle option) 9, 12, 24,
 25, 27, 29, 31, 32, 34, 36,
 37, 40, 42, 43, 45, 46, 48,
 49, 51, 53, 54, 56, 60, 62,
 63, 65, 66, 69, 70, 72

D

ddsudoku (environment) .. 1251
 \ddsudokucell 12, 1245
 \ddsudokusetup 12, 1247
 \definecounterstyle .. 11, 619,
 623, 624, 632
 \Diamond 20, 2747
 \Down 21, 3003

E

\ENDC 2876, 2879
 \ENDR 2877, 2880
 environments:
 battleship 1001
 bokkusu 1054
 bridges 1146
 chaossudoku 1204
 ddsudoku 1251
 fourwinds 1319
 hakyuu 1366
 hitori 1413
 kakuro 1552
 kendoku 1599
 killersudoku 1646
 laserbeam 1752
 logicpuzzle 8, 748
 lpsudoku 1845
 magiclabyrinth 1903
 magnets 2058
 masyu 2124
 minesweeper 2182
 nonogram 2379
 numberlink 2445

puzzlebackground .. [9](#), [566](#)
 puzzleforeground .. [9](#), [573](#)
 resuko [2687](#)
 schatzsuche [2748](#)
 skyline [2823](#)
 slitherlink [2884](#)
 starbattle [2934](#)
 starsandarrows [3005](#)
 sunandmoon [3130](#)
 tentsandtrees [3211](#)
 tunnel [3300](#)
 extracells (Puzzle option) ... [51](#)

F

\fillarea [10](#), [445](#)
 \fillcell [10](#), [369](#), [376](#), [393](#), [400](#),
 [407](#), [423](#), [436](#), [2234](#), [2248](#)
 \fillcolumn [10](#), [428](#)
 \filldiagonals [10](#), [515](#)
 \fillrow [10](#), [416](#)
 fontsize (Puzzle option) [8](#), [9](#), [24](#),
 [25](#), [27](#), [30](#), [32](#), [34](#), [35](#), [37](#),
 [39](#), [41](#), [43](#), [44](#), [46](#), [47](#), [49](#),
 [50](#), [52](#), [54](#), [56](#), [60](#), [61](#), [63](#),
 [65](#), [66](#), [69](#), [70](#), [72](#)
 fourwinds (environment) . [1319](#)
 \fourwindscell . [14](#), [1293](#), [2429](#)
 \fourwindssetup [14](#), [1315](#)
 \framearea [10](#), [441](#), [1696](#)
 \framepuzzle [11](#), [547](#), [2075](#), [2395](#)

G

\Graveltrap [19](#), [2575](#)
 grid (Puzzle option) [29](#)

H

hakyuu (environment) [1366](#)
 \hakyuucell [14](#), [1360](#)
 \hakyuusetup [14](#), [1362](#)
 helplines (Puzzle option) [51](#)
 hitori (environment) [1413](#)
 \hitoricell [1407](#)
 \hitorisetup [15](#), [1409](#)

I

\ifLP@KKR@solution [1455](#), [1488](#)
 \Island [849](#), [901](#)

K

kakuro (environment) [1552](#)
 \kakurocell [1479](#)
 \kakurocolumn [15](#), [1506](#)
 \kakurorow [15](#), [1496](#)
 \kakurosetup [15](#), [1481](#)
 kendoku (environment) ... [1599](#)
 \kendokucell [15](#), [1593](#)
 \kendokusetup [15](#), [1595](#)
 killersudoku (environment) [1646](#)
 \killersudokucell ... [15](#), [1640](#)
 \killersudokusetup .. [15](#), [1642](#)
 \KKR [15](#), [1516](#)

L

\laser [16](#), [1692](#)
 laserbeam (environment) . [1752](#)
 \laserbeamsetup [16](#), [1699](#)
 \laserH [16](#), [1687](#)
 \laserV [16](#), [1688](#)
 \layer [3284](#)
 \Left [21](#), [3001](#)
 \LeftDown [21](#), [3002](#)
 \LeftUp [21](#), [3000](#)
 \link [19](#), [2437](#)
 logicpuzzle (environment) [8](#), [748](#)
 logicpuzzle.sty (Package) [6](#), [7](#),
 [72](#)
 \logicpuzzlecell [781](#)
 \logicpuzzlesetup [783](#)
 \LP@Arrow [1713](#), [1715](#)
 \LP@ArrowA .. [2988](#), [2997](#), [2998](#),
 [2999](#), [3000](#), [3001](#), [3002](#),
 [3003](#), [3004](#)
 \LP@BG@bgcolor [1160](#)
 \LP@BG@columns [1152](#), [1160](#),
 [1167](#), [1170](#)
 \LP@BG@counterstyle [1176](#)
 \LP@BG@grid [1161](#), [1164](#)
 \LP@BG@init@package [1081](#), [1083](#),
 [1084](#), [1085](#), [1086](#), [1087](#),
 [1088](#), [1089](#), [1090](#), [1091](#),
 [1092](#), [1093](#), [1094](#), [1095](#)
 \LP@BG@init@prefix [1080](#), [1083](#),
 [1084](#), [1085](#), [1086](#), [1087](#),
 [1088](#), [1089](#), [1090](#), [1091](#),
 [1092](#), [1093](#), [1094](#), [1095](#)

`\LP@BG@rows` . 1151, 1160, 1167, 1170
`\LP@BG@scale` 1159
`\LP@BG@title` 1156, 1158
`\LP@BG@titleindent` 1158
`\LP@BG@titlewidth` 1158
`\LP@BG@width` 1155
`\LP@bgcolor` 41, 206, 208, 341, 343, 696, 1299, 1301, 1521, 1523, 2585, 2587, 2667, 2669
`\LP@BK@bgcolor` 1068
`\LP@BK@columns` 1060, 1068, 1069
`\LP@BK@counterstyle` 1073
`\LP@BK@init@package` 1030, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040, 1041, 1042, 1043
`\LP@BK@init@prefix` 1029, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040, 1041, 1042, 1043
`\LP@BK@rows` .. 1059, 1068, 1069
`\LP@BK@scale` 1067
`\LP@BK@title` 1064, 1066
`\LP@BK@titleindent` 1066
`\LP@BK@titlewidth` 1066
`\LP@BK@width` 1063
`\LP@Block` ... 414, 532, 534, [639](#)
`\LP@bottomrow` . [580](#), 1045, 2813
`\LP@bottomrow@edge` . [584](#), 1689
`\LP@BS@bgcolor` 1016
`\LP@BS@columns` .. 893, 900, 905, 1008, 1016, 1017
`\LP@BS@counterstyle` 1021
`\LP@BS@init@package` . 822, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838
`\LP@BS@init@prefix` .. 821, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838
`\LP@BS@printship` 859, 878
`\LP@BS@printshipbox` 873, 1022
`\LP@BS@rows` 893, 900, 905, 1007, 1016, 1017
`\LP@BS@sbindent` 1022
`\LP@BS@sbshipscale` .. 861, 862, 863, 864, 865, 866, 867, 868, 869, 870
`\LP@BS@sbwidth` 1022
`\LP@BS@scale` 843, 844, 845, 846, 847, 848, 849, 850, 1015
`\LP@BS@shipbox` . 840, 912, 1022
`\LP@BS@shipcolor` 843, 844, 845, 846, 847, 848, 852, 853, 854, 855, 856, 857
`\LP@BS@title` 1012, 1014
`\LP@BS@titleindent` 1014
`\LP@BS@titlewidth` 1014
`\LP@BS@width` 1011
`\LP@c@bg@fw` .. 1300, 1301, 1311
`\LP@c@bg@parkinglot` 2586, 2587, 2593
`\LP@c@bg@track` 2668, 2669, 2676
`\LP@c@cross` . 2644, 2647, 2655, 2659
`\LP@c@curve` . 2644, 2647, 2655, 2659
`\LP@c@parkinglot` .. 2583, 2584, 2592, 2594
`\LP@c@pitlane` 2602, 2603, 2605, 2607
`\LP@c@stararrow` ... 2996, 2997, 2998, 2999, 3000, 3001, 3002, 3003, 3004
`\LP@c@straight` 2644, 2647, 2655, 2659
`\LP@color` 40, 526, 641, 680, 1114, 1133, 1306, 1532, 2120, 2442, 2582, 2584, 2601, 2603
`\LP@columns` . 38, 320, 332, 413, 520, 551, 688, 1823, 1830, 1836, 2304, 2641
`\LP@counterstyle` 45
`\LP@Cross` 1705, [1727](#)
`\LP@CS@bgcolor` 1218
`\LP@CS@columns` 1210, 1218, 1219
`\LP@CS@counterstyle` 1223
`\LP@CS@init@package` 1184, 1186, 1187, 1188, 1189, 1190, 1191, 1192, 1193, 1194,

1195, 1196
 \LP@CS@init@prefix 1183, 1186,
 1187, 1188, 1189, 1190,
 1191, 1192, 1193, 1194,
 1195, 1196
 \LP@CS@rows .. 1209, 1218, 1219
 \LP@CS@scale 1217
 \LP@CS@title 1214, 1216
 \LP@CS@titleindent 1216
 \LP@CS@titlewidth 1216
 \LP@CS@width 1213
 \LP@cwoffset . 43, 629, 637, 692
 \LP@DarkCloud 3052, 3120
 \LP@dck@fontsize 72, 98
 \LP@DDS@bgcolor 1265
 \LP@DDS@columns ... 1257, 1265,
 1266
 \LP@DDS@counterstyle ... 1270
 \LP@DDS@init@package ... 1231,
 1233, 1234, 1235, 1236,
 1237, 1238, 1239, 1240,
 1241, 1242, 1243
 \LP@DDS@init@prefix 1230, 1233,
 1234, 1235, 1236, 1237,
 1238, 1239, 1240, 1241,
 1242, 1243
 \LP@DDS@rows . 1256, 1265, 1266
 \LP@DDS@scale 1264
 \LP@DDS@title 1261, 1263
 \LP@DDS@titleindent 1263
 \LP@DDS@titlewidth 1263
 \LP@DDS@width 1260
 \LP@define@choicetype@fontsize
 69, 773, 838, 1043, 1095,
 1196, 1243, 1291, 1358,
 1405, 1453, 1591, 1638,
 1685, 1791, 1885, 1942,
 2100, 2163, 2224, 2425,
 2485, 2726, 2787, 2865,
 2926, 2973, 3044, 3169,
 3250
 \LP@define@key ... 57, 751, 753,
 755, 757, 759, 761, 763,
 765, 767, 769, 771, 824,
 825, 826, 827, 828, 829,
 830, 831, 832, 833, 834,
 835, 836, 837, 1032, 1033,
 1034, 1035, 1036, 1037,
 1038, 1039, 1040, 1041,
 1042, 1083, 1084, 1085,
 1086, 1087, 1088, 1089,
 1090, 1091, 1092, 1093,
 1094, 1186, 1187, 1188,
 1189, 1190, 1191, 1192,
 1193, 1194, 1195, 1233,
 1234, 1235, 1236, 1237,
 1238, 1239, 1240, 1241,
 1242, 1280, 1281, 1282,
 1283, 1284, 1285, 1286,
 1287, 1288, 1289, 1290,
 1348, 1349, 1350, 1351,
 1352, 1353, 1354, 1355,
 1356, 1357, 1395, 1396,
 1397, 1398, 1399, 1400,
 1401, 1402, 1403, 1404,
 1442, 1443, 1444, 1445,
 1446, 1447, 1448, 1449,
 1450, 1451, 1452, 1581,
 1582, 1583, 1584, 1585,
 1586, 1587, 1588, 1589,
 1590, 1628, 1629, 1630,
 1631, 1632, 1633, 1634,
 1635, 1636, 1637, 1675,
 1676, 1677, 1678, 1679,
 1680, 1681, 1682, 1683,
 1684, 1781, 1782, 1783,
 1784, 1785, 1786, 1787,
 1788, 1789, 1790, 1875,
 1876, 1877, 1878, 1879,
 1880, 1881, 1882, 1883,
 1884, 1932, 1933, 1934,
 1935, 1936, 1937, 1938,
 1939, 1940, 1941, 2089,
 2090, 2091, 2092, 2093,
 2094, 2095, 2096, 2097,
 2098, 2099, 2153, 2154,
 2155, 2156, 2157, 2158,
 2159, 2160, 2161, 2162,
 2210, 2211, 2212, 2213,
 2214, 2215, 2216, 2217,
 2218, 2219, 2220, 2221,
 2222, 2223, 2414, 2415,
 2416, 2417, 2418, 2419,
 2420, 2421, 2422, 2423,

2424, 2474, 2475, 2476,
 2477, 2478, 2479, 2480,
 2481, 2482, 2483, 2484,
 2716, 2717, 2718, 2719,
 2720, 2721, 2722, 2723,
 2724, 2725, 2777, 2778,
 2779, 2780, 2781, 2782,
 2783, 2784, 2785, 2786,
 2855, 2856, 2857, 2858,
 2859, 2860, 2861, 2862,
 2863, 2864, 2916, 2917,
 2918, 2919, 2920, 2921,
 2922, 2923, 2924, 2925,
 2963, 2964, 2965, 2966,
 2967, 2968, 2969, 2970,
 2971, 2972, 3034, 3035,
 3036, 3037, 3038, 3039,
 3040, 3041, 3042, 3043,
 3159, 3160, 3161, 3162,
 3163, 3164, 3165, 3166,
 3167, 3168, 3240, 3241,
 3242, 3243, 3244, 3245,
 3246, 3247, 3248, 3249
 \LP@definecolor 125
 \LP@Diamond 2734, 2747
 \LP@dir 487, 489, 492, 495, 498,
 501, 504, 507, 510
 \LP@direction 456, 458, 461, 464,
 467, 470, 473, 476, 479
 \LP@draw@opacity 52, 153, 1801,
 1803, 1805, 1807, 1809,
 1811, 1813, 1815
 \LP@drawbackground .. 156, 804,
 1016, 1068, 1160, 1218,
 1265, 1333, 1380, 1427,
 1566, 1613, 1660, 1766,
 1859, 1917, 2072, 2138,
 2196, 2393, 2459, 2701,
 2762, 2837, 2900, 2948,
 3019, 3144, 3225, 3314
 \LP@drawcounter 614, 810, 1021,
 1073, 1176, 1223, 1270,
 1338, 1385, 1432, 1571,
 1618, 1665, 1771, 1865,
 1922, 2079, 2143, 2201,
 2404, 2464, 2706, 2767,
 2845, 2906, 2953, 3024,
 3149, 3230, 3319
 \LP@drawgrid ... 146, 806, 1017,
 1069, 1167, 1170, 1219,
 1266, 1334, 1381, 1428,
 1567, 1614, 1661, 1767,
 1860, 1918, 2073, 2139,
 2197, 2394, 2460, 2702,
 2763, 2838, 2901, 2949,
 3020, 3145, 3226, 3315
 \LP@drawgriddots .. 2873, 2902
 \LP@drawsudokugrid 1799, 1861,
 2840
 \LP@element 257, 261, 270, 274,
 283, 287, 296, 300, 363,
 365, 374, 378, 387, 389,
 398, 402, 420, 422, 433,
 435, 876, 878, 1109, 1111,
 1117, 1128, 1130, 1137,
 1500, 1502, 1510, 1512,
 1953, 1955, 1967, 1969,
 2262, 2266, 2271, 2276,
 2292, 2295
 \LP@env@prefix ... 47, 204, 205,
 254, 267, 280, 293, 306,
 311, 316, 317, 318, 319,
 325, 326, 327, 328, 339,
 340, 409, 410, 411, 412,
 517, 518, 519, 549, 550,
 616, 664, 709, 729, 732,
 1107, 1108, 1126, 1127,
 1295, 1296, 1297, 1298,
 1518, 1519, 1520, 1546,
 1717, 1729, 1742, 1820,
 1821, 1822, 1951, 1962,
 1963, 1980, 1996, 2012,
 2028, 2116, 2256, 2302,
 2303, 2439, 2495, 2504,
 2518, 2527, 2539, 2551,
 2568, 2578, 2579, 2580,
 2581, 2599, 2600, 2612,
 2640, 2665, 2666, 2990,
 3261, 3280
 \LP@extracells 700
 \LP@fe@column 1981, 1983, 1997,
 1999, 2013, 2015, 2029,
 2031, 2044, 2047, 2052,
 2055, 2228, 2231

`\LP@fe@length` 2228, 2230, 2242, 2244
`\LP@fe@row` .. 1981, 1984, 1997, 2000, 2013, 2016, 2029, 2032, 2044, 2047, 2052, 2055, 2242, 2245
`\LP@fontsize` 42, 261, 274, 287, 300, 321, 334, 705, 1116, 1135, 1311, 1839, 1842, 1955, 1969, 1987, 1990, 2003, 2006, 2019, 2022, 2035, 2038, 2266, 2271, 2275, 2294, 2594
`\LP@FW@bgcolor` 1333
`\LP@FW@columns` 1325, 1333, 1334
`\LP@FW@counterstyle` 1338
`\LP@fw@dir` 1302, 1307
`\LP@FW@init@package` 1278, 1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287, 1288, 1289, 1290, 1291
`\LP@FW@init@prefix` 1277, 1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287, 1288, 1289, 1290, 1291
`\LP@fw@length` 1302, 1307
`\LP@fw@linestyle` 55, 1306
`\LP@FW@rows` .. 1324, 1333, 1334
`\LP@FW@scale` 1332
`\LP@FW@title` 1329, 1331
`\LP@FW@titleindent` 1331
`\LP@FW@titlewidth` 1331
`\LP@FW@width` 1328
`\LP@G@setcellcontent` 220, 238, 242, 246, 250, 259, 272, 285, 298, 321, 333, 414, 531, 533, 894, 901, 906, 1705, 1709, 1713, 3275
`\LP@G@trackC` [2502](#), 2512, 2513, 2514, 2515
`\LP@G@trackCR` [2549](#)
`\LP@G@trackS` . [2525](#), 2534, 2535
`\LP@graveltrap` [2566](#)
`\LP@grid@linestyle` 51, 152, 349, 717, 2889
`\LP@HT@bgcolor` 1427
`\LP@HT@color` 367, 373, 391, 397
`\LP@HT@columns` 1419, 1427, 1428
`\LP@HT@counterstyle` 1432
`\LP@HT@init@package` 1393, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 1405
`\LP@HT@init@prefix` 1392, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 1405
`\LP@HT@rows` .. 1418, 1427, 1428
`\LP@HT@scale` 1426
`\LP@HT@title` 1423, 1425
`\LP@HT@titleindent` 1425
`\LP@HT@titlewidth` 1425
`\LP@HT@width` 1422
`\LP@HY@bgcolor` 1380
`\LP@HY@columns` 1372, 1380, 1381
`\LP@HY@counterstyle` 1385
`\LP@HY@init@package` 1346, 1348, 1349, 1350, 1351, 1352, 1353, 1354, 1355, 1356, 1357, 1358
`\LP@HY@init@prefix` 1345, 1348, 1349, 1350, 1351, 1352, 1353, 1354, 1355, 1356, 1357, 1358
`\LP@HY@rows` .. 1371, 1380, 1381
`\LP@HY@scale` 1379
`\LP@HY@title` 1376, 1378
`\LP@HY@titleindent` 1378
`\LP@HY@titlewidth` 1378
`\LP@HY@width` 1375
`\LP@ingrid` . [171](#), 320, 331, 413, 893, 900, 905
`\LP@KD@bgcolor` 1613
`\LP@KD@columns` 1605, 1613, 1614
`\LP@KD@counterstyle` 1618
`\LP@KD@init@package` 1579, 1581, 1582, 1583, 1584, 1585, 1586, 1587, 1588, 1589, 1590, 1591
`\LP@KD@init@prefix` 1578, 1581, 1582, 1583, 1584, 1585, 1586, 1587, 1588, 1589, 1590, 1591
`\LP@KD@rows` .. 1604, 1613, 1614

\LP@KD@scale 1612
\LP@KD@title 1609, 1611
\LP@KD@titleindent 1611
\LP@KD@titlewidth 1611
\LP@KD@width 1608
\LP@KKR@bgcolor 1566
\LP@KKR@columns ... 1558, 1566,
1567
\LP@KKR@counterstyle ... 1571
\LP@KKR@init@package ... 1440,
1442, 1443, 1444, 1445,
1446, 1447, 1448, 1449,
1450, 1451, 1452, 1453
\LP@KKR@init@prefix 1439, 1442,
1443, 1444, 1445, 1446,
1447, 1448, 1449, 1450,
1451, 1452, 1453
\LP@KKR@print@element .. 1485,
1502, 1512
\LP@KKR@rows . 1557, 1566, 1567
\LP@KKR@scale 1565
\LP@KKR@resolution .. 1457, 1460,
1463, 1468, 1471, 1474
\LP@KKR@resolutionfalse .. 1455,
1464, 1475
\LP@KKR@resolutiontrue ... 1461,
1472
\LP@KKR@title 1562, 1564
\LP@KKR@titleindent 1564
\LP@KKR@titlewidth 1564
\LP@KKR@width 1561
\LP@KSDK@bgcolor 1660
\LP@KSDK@columns .. 1652, 1660,
1661
\LP@KSDK@counterstyle .. 1665
\LP@KSDK@init@package .. 1626,
1628, 1629, 1630, 1631,
1632, 1633, 1634, 1635,
1636, 1637, 1638
\LP@KSDK@init@prefix ... 1625,
1628, 1629, 1630, 1631,
1632, 1633, 1634, 1635,
1636, 1637, 1638
\LP@KSDK@rows 1651, 1660, 1661
\LP@KSDK@scale 1659
\LP@KSDK@title 1656, 1658
\LP@KSDK@titleindent ... 1658
\LP@KSDK@titlewidth 1658
\LP@KSDK@width 1655
\LP@LB@bgcolor 1766
\LP@LB@columns 1758, 1766, 1767
\LP@LB@counterstyle 1771
\LP@LB@init@package 1673, 1675,
1676, 1677, 1678, 1679,
1680, 1681, 1682, 1683,
1684, 1685
\LP@LB@init@prefix 1672, 1675,
1676, 1677, 1678, 1679,
1680, 1681, 1682, 1683,
1684, 1685
\LP@LB@rows .. 1757, 1766, 1767
\LP@LB@scale 1765
\LP@LB@title 1762, 1764
\LP@LB@titleindent 1764
\LP@LB@titlewidth 1764
\LP@LB@width 1761
\LP@leftcolumn
.. 588, 1046, 1688, 1947,
2815, 2976, 3172, 3253
\LP@length 36, 2613, 2616, 2629
\LP@Line 643, 1144, 2442
\LP@line 2258, 2262, 2287, 2292
\LP@Line@double . 645, 660, 666
\LP@Line@linecolor .. 650, 661,
668, 670
\LP@Line@linewidth .. 655, 662,
667, 669, 671
\LP@LP@bgcolor 805
\LP@LP@columns .. 794, 804, 806
\LP@LP@counterstyle 810
\LP@LP@init@package . 749, 751,
753, 755, 757, 759, 761,
763, 765, 767, 769, 771,
774
\LP@LP@init@prefix .. 748, 751,
753, 755, 757, 759, 761,
763, 765, 767, 769, 771,
773
\LP@LP@rows 793, 804, 806
\LP@LP@scale 803
\LP@LP@title 798, 801
\LP@LP@titleindent 800
\LP@LP@titlewidth 801
\LP@LP@width 797

\LP@magnetsgrid ... [1818](#), [2074](#)
\LP@Mine [2171](#), [2181](#)
\LP@Mirror [1709](#), [1738](#)
\LP@ML@bgcolor [1917](#)
\LP@ML@columns [1909](#), [1917](#), [1918](#)
\LP@ML@counterstyle [1922](#)
\LP@ML@init@package [1873](#), [1875](#),
[1876](#), [1877](#), [1878](#), [1879](#),
[1880](#), [1881](#), [1882](#), [1883](#),
[1884](#), [1885](#)
\LP@ML@init@prefix [1872](#), [1875](#),
[1876](#), [1877](#), [1878](#), [1879](#),
[1880](#), [1881](#), [1882](#), [1883](#),
[1884](#), [1885](#)
\LP@ML@rows .. [1908](#), [1917](#), [1918](#)
\LP@ML@scale [1916](#)
\LP@ML@title [1913](#), [1915](#)
\LP@ML@titleindent [1915](#)
\LP@ML@titlewidth [1915](#)
\LP@ML@width [1912](#)
\LP@MN@bgcolor [2072](#)
\LP@MN@columns [2064](#), [2072](#), [2073](#)
\LP@MN@counterstyle [2079](#)
\LP@MN@init@package [1930](#), [1932](#),
[1933](#), [1934](#), [1935](#), [1936](#),
[1937](#), [1938](#), [1939](#), [1940](#),
[1941](#), [1942](#)
\LP@MN@init@prefix [1929](#), [1932](#),
[1933](#), [1934](#), [1935](#), [1936](#),
[1937](#), [1938](#), [1939](#), [1940](#),
[1941](#), [1942](#)
\LP@MN@rows .. [2063](#), [2072](#), [2073](#)
\LP@MN@scale [2071](#)
\LP@MN@title [2068](#), [2070](#)
\LP@MN@titleindent [2070](#)
\LP@MN@titlewidth [2070](#)
\LP@MN@width [2067](#)
\LP@Moon [3056](#), [3121](#)
\LP@MoonB [3081](#), [3125](#)
\LP@MoonBL [3096](#), [3127](#)
\LP@MoonBR [3104](#), [3128](#)
\LP@MoonL [3067](#), [3123](#)
\LP@MoonR [3060](#), [3122](#)
\LP@MoonT [3074](#), [3124](#)
\LP@MoonTL [3088](#), [3126](#)
\LP@MoonTR [3112](#), [3129](#)
\LP@MS@bgcolor [2196](#)
\LP@MS@columns [2188](#), [2196](#), [2197](#)
\LP@MS@counterstyle [2201](#)
\LP@MS@init@package [2151](#), [2153](#),
[2154](#), [2155](#), [2156](#), [2157](#),
[2158](#), [2159](#), [2160](#), [2161](#),
[2162](#), [2163](#)
\LP@MS@init@prefix [2150](#), [2153](#),
[2154](#), [2155](#), [2156](#), [2157](#),
[2158](#), [2159](#), [2160](#), [2161](#),
[2162](#), [2163](#)
\LP@MS@rows .. [2187](#), [2196](#), [2197](#)
\LP@MS@scale [2195](#)
\LP@MS@title [2192](#), [2194](#)
\LP@MS@titleindent [2194](#)
\LP@MS@titlewidth [2194](#)
\LP@MS@width [2191](#)
\LP@MY@bgcolor [2138](#)
\LP@MY@columns [2130](#), [2138](#), [2139](#)
\LP@MY@counterstyle [2143](#)
\LP@MY@init@package [2087](#), [2089](#),
[2090](#), [2091](#), [2092](#), [2093](#),
[2094](#), [2095](#), [2096](#), [2097](#),
[2098](#), [2099](#), [2100](#)
\LP@MY@init@prefix [2086](#), [2089](#),
[2090](#), [2091](#), [2092](#), [2093](#),
[2094](#), [2095](#), [2096](#), [2097](#),
[2098](#), [2099](#), [2100](#)
\LP@MY@rows .. [2129](#), [2138](#), [2139](#)
\LP@MY@scale [2137](#)
\LP@MY@title [2134](#), [2136](#)
\LP@MY@titleindent [2136](#)
\LP@MY@titlewidth [2136](#)
\LP@MY@width [2133](#)
\LP@NG@bgcolor [2393](#)
\LP@NG@columns [2385](#), [2393](#), [2394](#)
\LP@NG@counterstyle [2404](#)
\LP@NG@extracells
.. [735](#), [2306](#), [2311](#), [2314](#),
[2325](#), [2341](#), [2344](#), [2358](#)
\LP@NG@helplines .. [2336](#), [2340](#),
[2350](#), [2357](#), [2365](#)
\LP@NG@init@package [2209](#), [2210](#),
[2211](#), [2212](#), [2213](#), [2214](#),
[2215](#), [2216](#), [2217](#), [2218](#),
[2219](#), [2220](#), [2221](#), [2222](#),
[2223](#), [2224](#)

\LP@NG@init@prefix 2208, 2210,
 2211, 2212, 2213, 2214,
 2215, 2216, 2217, 2218,
 2219, 2220, 2221, 2222,
 2223, 2224
 \LP@NG@rows .. 2384, 2393, 2394
 \LP@NG@scale 2392
 \LP@NG@solution 2396
 \LP@NG@title 2389, 2391
 \LP@NG@titleindent 2391
 \LP@NG@titlewidth 2391
 \LP@NG@width 2388
 \LP@NL@bgcolor 2459
 \LP@NL@columns 2451, 2459, 2460
 \LP@NL@counterstyle 2464
 \LP@NL@init@package 2412, 2414,
 2415, 2416, 2417, 2418,
 2419, 2420, 2421, 2422,
 2423, 2424, 2425
 \LP@NL@init@prefix 2411, 2414,
 2415, 2416, 2417, 2418,
 2419, 2420, 2421, 2422,
 2423, 2424, 2425
 \LP@NL@rows .. 2450, 2459, 2460
 \LP@NL@scale 2458
 \LP@NL@title 2455, 2457
 \LP@NL@titleindent 2457
 \LP@NL@titlewidth 2457
 \LP@NL@width 2454
 \LP@nonogramgrid .. 2300, 2398
 \LP@normallines 49, 152,
 348, 641, 721, 1534, 1827,
 1829, 1831, 2318, 2330,
 2592, 2605, 2607, 2738,
 2740, 2742, 2744, 3269
 \LP@package . 48, 320, 332, 413,
 540, 713, 730
 \LP@Pfive 56, 212, 220, 222, 224,
 226, 1527, 1955, 1969
 \LP@plength 487, 490, 493, 496,
 499, 502, 505, 508, 511
 \LP@Portal 3259, 3275
 \LP@rel@tikzpath
 53, 455, 486, 1305, 2119,
 2441, 2671, 3282
 \LP@rightcolumn 596, 1048, 2816
 \LP@rightcolumn@edge 600, 1690
 \LP@rotate
 . 1718, 1719, 1720, 1721,
 1724, 1740, 1741, 1747
 \LP@rows 37, 320, 332, 413, 521,
 528, 552, 684, 733, 1824,
 1965, 2289, 2305
 \LP@RSK@bgcolor 2701
 \LP@RSK@columns ... 2693, 2701,
 2702
 \LP@RSK@counterstyle ... 2706
 \LP@RSK@init@package ... 2472,
 2474, 2475, 2476, 2477,
 2478, 2479, 2480, 2481,
 2482, 2483, 2484, 2485
 \LP@RSK@init@prefix 2471, 2474,
 2475, 2476, 2477, 2478,
 2479, 2480, 2481, 2482,
 2483, 2484, 2485
 \LP@RSK@rows . 2692, 2701, 2702
 \LP@RSK@scale 2700
 \LP@RSK@title 2697, 2699
 \LP@RSK@titleindent 2699
 \LP@RSK@titlewidth 2699
 \LP@RSK@width 2696
 \LP@SAA@bgcolor 3019
 \LP@SAA@columns ... 3011, 3019,
 3020
 \LP@SAA@counterstyle ... 3024
 \LP@SAA@init@package ... 2961,
 2963, 2964, 2965, 2966,
 2967, 2968, 2969, 2970,
 2971, 2972, 2973
 \LP@SAA@init@prefix 2960, 2963,
 2964, 2965, 2966, 2967,
 2968, 2969, 2970, 2971,
 2972, 2973
 \LP@SAA@rows . 3010, 3019, 3020
 \LP@SAA@scale 3018
 \LP@SAA@title 3015, 3017
 \LP@SAA@titleindent 3017
 \LP@SAA@titlewidth 3017
 \LP@SAA@width 3014
 \LP@SAM@bgcolor 3144
 \LP@SAM@columns ... 3136, 3144,
 3145
 \LP@SAM@counterstyle ... 3149

\LP@SAM@init@package ... 3032,
 3034, 3035, 3036, 3037,
 3038, 3039, 3040, 3041,
 3042, 3043, 3044
 \LP@SAM@init@prefix 3031, 3034,
 3035, 3036, 3037, 3038,
 3039, 3040, 3041, 3042,
 3043, 3044
 \LP@SAM@rows . 3135, 3144, 3145
 \LP@SAM@scale 3143
 \LP@SAM@title 3140, 3142
 \LP@SAM@titleindent 3142
 \LP@SAM@titlewidth 3142
 \LP@SAM@width 3139
 \LP@SB@bgcolor 2948
 \LP@SB@columns 2940, 2948, 2949
 \LP@SB@counterstyle 2953
 \LP@SB@init@package 2914, 2916,
 2917, 2918, 2919, 2920,
 2921, 2922, 2923, 2924,
 2925, 2926
 \LP@SB@init@prefix 2913, 2916,
 2917, 2918, 2919, 2920,
 2921, 2922, 2923, 2924,
 2925, 2926
 \LP@SB@rows .. 2939, 2948, 2949
 \LP@SB@scale 2947
 \LP@SB@title 2944, 2946
 \LP@SB@titleindent 2946
 \LP@SB@titlewidth 2946
 \LP@SB@width 2943
 \LP@sbcbgcolor . 342, 343, 348,
 352
 \LP@sc@column ... 329, 331, 334
 \LP@sc@row 329, 331, 334
 \LP@scale 39, 213, 217, 354, 640,
 667, 669, 671, 676, 1116,
 1135, 1306, 1528, 1538,
 1541, 1722, 1724, 1732,
 1733, 1745, 1746, 1899,
 2110, 2173, 2496, 2497,
 2498, 2505, 2507, 2508,
 2519, 2521, 2528, 2530,
 2540, 2541, 2543, 2545,
 2552, 2554, 2556, 2558,
 2569, 2613, 2673, 2676,
 2677, 2680, 2736, 2984,
 2991, 2993, 3053, 3057,
 3062, 3069, 3076, 3083,
 3090, 3098, 3106, 3114,
 3180, 3181, 3185, 3202,
 3203, 3206, 3207, 3262,
 3289, 3291, 3292, 3293,
 3294, 3295, 3296
 \LP@scc@h 226, 231, 234
 \LP@scc@v 222, 232, 234
 \LP@SDK@bgcolor 1859
 \LP@SDK@columns ... 1851, 1859,
 1860
 \LP@SDK@counterstyle ... 1865
 \LP@SDK@init@package ... 1779,
 1781, 1782, 1783, 1784,
 1785, 1786, 1787, 1788,
 1789, 1790, 1791
 \LP@SDK@init@prefix 1778, 1781,
 1782, 1783, 1784, 1785,
 1786, 1787, 1788, 1789,
 1790, 1791
 \LP@SDK@rows . 1850, 1859, 1860
 \LP@SDK@scale 1858
 \LP@SDK@title 1855, 1857
 \LP@SDK@titleindent 1857
 \LP@SDK@titlewidth 1857
 \LP@SDK@width 1854
 \LP@set@env@prefix .. 707, 792,
 1005, 1058, 1150, 1208,
 1255, 1323, 1370, 1417,
 1556, 1603, 1650, 1756,
 1849, 1907, 2062, 2128,
 2186, 2383, 2449, 2691,
 2752, 2827, 2888, 2938,
 3009, 3134, 3215, 3304
 \LP@set@LP@bgcolor
 205, 340, 694, 1297,
 1519, 2580, 2666
 \LP@set@LP@color 410, 678,
 1108, 1127, 1296, 1520,
 2116, 2439, 2579, 2600
 \LP@set@LP@columns .. 319, 328,
 412, 517, 549, 686, 1821,
 2302, 2640
 \LP@set@LP@cwoffset . 616, 690
 \LP@set@LP@extracells 698, 732

`\LP@set@LP@fontsize` . 254, 267, 280, 293, 317, 326, [702](#), 1107, 1126, 1295, 1820, 1951, 1962, 1980, 1996, 2012, 2028, 2256, 2581
`\LP@set@LP@rows` . 318, 327, 411, 518, 550, [682](#), 729, 1822, 1963, 2303
`\LP@set@LP@scale` 204, 306, 311, 316, 325, 339, 409, 519, 664, [674](#), 1298, 1518, 1546, 1717, 1729, 1742, 2495, 2504, 2518, 2527, 2539, 2551, 2568, 2578, 2599, 2612, 2665, 2990, 3261, 3280
`\LP@set@package` [711](#), 791, 1004, 1057, 1149, 1207, 1254, 1322, 1369, 1416, 1555, 1602, 1649, 1755, 1848, 1906, 2061, 2127, 2185, 2382, 2448, 2690, 2751, 2826, 2887, 2937, 3008, 3133, 3214, 3303
`\LP@setcellcontent` [236](#)
`\LP@setcellcontentC` [240](#)
`\LP@setcellcontentHE` [248](#)
`\LP@setcellcontentVE` [244](#)
`\LP@setcolumncontents` [265](#), 312, 590, 598, 889
`\LP@setcolumncontents@edge` [291](#), 602
`\LP@setrowcontents` .. [252](#), 307, 582, 594, 884
`\LP@setrowcontents@edge` . [278](#), 586
`\LP@SK@bgcolor` 2900
`\LP@SK@columns` 2892, 2900, 2901
`\LP@SK@counterstyle` 2906
`\LP@SK@init@package` 2853, 2855, 2856, 2857, 2858, 2859, 2860, 2861, 2862, 2863, 2864, 2865
`\LP@SK@init@prefix` 2852, 2855, 2856, 2857, 2858, 2859, 2860, 2861, 2862, 2863, 2864, 2865
`\LP@SK@rows` .. 2891, 2900, 2901
`\LP@SK@scale` 2899
`\LP@SK@title` 2896, 2898
`\LP@SK@titleindent` 2898
`\LP@SK@titlewidth` 2898
`\LP@SK@width` 2895
`\LP@SL@bgcolor` 2837
`\LP@SL@columns` 2796, 2807, 2829, 2837, 2838
`\LP@SL@counterstyle` 2845
`\LP@SL@init@package` 2775, 2777, 2778, 2779, 2780, 2781, 2782, 2783, 2784, 2785, 2786, 2787
`\LP@SL@init@prefix` 2774, 2777, 2778, 2779, 2780, 2781, 2782, 2783, 2784, 2785, 2786, 2787
`\LP@SL@rows` . 2795, 2806, 2828, 2837, 2838
`\LP@SL@scale` 2836
`\LP@SL@sudoku`
. 2789, 2791, 2794, 2798, 2802, 2805, 2809, 2839
`\LP@SL@title` 2833, 2835
`\LP@SL@titleindent` 2835
`\LP@SL@titlewidth` 2835
`\LP@SL@width` 2832
`\LP@sr@bgcolor` .. 207, 208, 213, 1522, 1523, 1528
`\LP@SS@bgcolor` 2762
`\LP@SS@columns` 2754, 2762, 2763
`\LP@SS@counterstyle` 2767
`\LP@SS@init@package` 2714, 2716, 2717, 2718, 2719, 2720, 2721, 2722, 2723, 2724, 2725, 2726
`\LP@SS@init@prefix` 2713, 2716, 2717, 2718, 2719, 2720, 2721, 2722, 2723, 2724, 2725, 2726
`\LP@SS@rows` .. 2753, 2762, 2763
`\LP@SS@scale` 2761
`\LP@SS@title` 2758, 2760
`\LP@SS@titleindent` 2760
`\LP@SS@titlewidth` 2760
`\LP@SS@width` 2757

[\LP@Star](#) [2982](#)
[\LP@TAT@bgcolor](#) [3225](#)
[\LP@TAT@columns](#) ... [3217](#), [3225](#),
[3226](#)
[\LP@TAT@counterstyle](#) ... [3230](#)
[\LP@TAT@init@package](#) ... [3157](#),
[3159](#), [3160](#), [3161](#), [3162](#),
[3163](#), [3164](#), [3165](#), [3166](#),
[3167](#), [3168](#), [3169](#)
[\LP@TAT@init@prefix](#) [3156](#), [3159](#),
[3160](#), [3161](#), [3162](#), [3163](#),
[3164](#), [3165](#), [3166](#), [3167](#),
[3168](#), [3169](#)
[\LP@TAT@rows](#) . [3216](#), [3225](#), [3226](#)
[\LP@TAT@scale](#) [3224](#)
[\LP@TAT@title](#) [3221](#), [3223](#)
[\LP@TAT@titleindent](#) [3223](#)
[\LP@TAT@titlewidth](#) [3223](#)
[\LP@TAT@width](#) [3220](#)
[\LP@Tent](#) [3200](#), [3210](#)
[\LP@thicklines](#) [50](#),
[443](#), [447](#), [555](#), [557](#), [560](#),
[563](#), [725](#), [736](#), [1801](#), [1803](#),
[1805](#), [1807](#), [1809](#), [1811](#),
[1813](#), [1815](#), [1833](#), [1835](#),
[1837](#), [1895](#), [1985](#), [2001](#),
[2017](#), [2033](#), [2046](#), [2054](#),
[2120](#), [2347](#), [2352](#), [2353](#),
[2362](#), [2367](#), [2369](#)
[\LP@titleformat](#) .. [46](#), [606](#), [801](#),
[1014](#), [1066](#), [1158](#), [1216](#),
[1263](#), [1331](#), [1378](#), [1425](#),
[1564](#), [1611](#), [1658](#), [1764](#),
[1857](#), [1915](#), [2070](#), [2136](#),
[2194](#), [2391](#), [2457](#), [2699](#),
[2760](#), [2835](#), [2898](#), [2946](#),
[3017](#), [3142](#), [3223](#), [3312](#)
[\LP@TN@bgcolor](#) [3314](#)
[\LP@TN@columns](#) [3306](#), [3314](#), [3315](#)
[\LP@TN@counterstyle](#) [3319](#)
[\LP@TN@init@package](#) [3238](#), [3240](#),
[3241](#), [3242](#), [3243](#), [3244](#),
[3245](#), [3246](#), [3247](#), [3248](#),
[3249](#), [3250](#)
[\LP@TN@init@prefix](#) [3237](#), [3240](#),
[3241](#), [3242](#), [3243](#), [3244](#),
[3245](#), [3246](#), [3247](#), [3248](#),
[3249](#), [3250](#)
[\LP@TN@rows](#) .. [3305](#), [3314](#), [3315](#)
[\LP@TN@scale](#) [3313](#)
[\LP@TN@title](#) [3310](#), [3312](#)
[\LP@TN@titleindent](#) [3312](#)
[\LP@TN@titlewidth](#) [3312](#)
[\LP@TN@width](#) [3309](#)
[\LP@toprow](#) [592](#), [1047](#), [1687](#), [1946](#),
[2814](#), [2975](#), [3171](#), [3252](#)
[\LP@trackC](#) [2493](#), [2626](#)
[\LP@trackCR](#) [2537](#), [2633](#)
[\LP@trackline](#) [2610](#), [2647](#), [2659](#)
[\LP@trackS](#) [2516](#), [2620](#)
[\LP@tracks@scale](#) [54](#), [2614](#)
[\LP@Tree](#) [3178](#), [3199](#)
LPbackground (PGF layer) 7–9, [83](#),
[88](#), [89](#), [154](#)
LPbackgroundtwo (PGF layer) 7, 8
LPbgcolor (PGF layer) 7, 8, [76](#), [78](#)
LPdump (PGF layer) . 7, 8, [78](#), [112](#)
lpenv.sty (Package) [97](#)
LPforeground (PGF layer) 7–9, [83](#),
[89](#), [144](#)
LPforegroundtwo (PGF layer) 7, 8,
[83](#), [118](#)
\LPlinewidth
. [1722](#), [1723](#), [1732](#), [1733](#),
[1745](#), [1746](#), [2991](#), [2992](#)
lpsudoku (environment) .. [1845](#)
\lpsudokucell [21](#), [1793](#)
\lpsudokusetup [21](#), [1795](#)

M

magiclabyrinth (environment) .
..... [1903](#)
\magiclabyrinthcell . [16](#), [1887](#)
\magiclabyrinthsetup [17](#), [1889](#)
magnets (environment) ... [2058](#)
\magnetscell [1944](#)
\magnetsH [17](#), [2042](#)
\magnetssetup [17](#), [1974](#)
\magnetsV [17](#), [2050](#)
main (PGF layer) ... 7–9, [78](#), [117](#)
masyu (environment) [2124](#)
\MasyuB [18](#), [2108](#)
\masyucell [17](#), [2102](#)
\masyuline [18](#), [2114](#)
\masyusetup [18](#), [2104](#)

- \MasyuW [17](#), [1897](#)
- \Mine [18](#), [2181](#)
- minesweeper (environment) [2182](#)
- \minesweepercell [2165](#)
- \minesweepersetup ... [18](#), [2167](#)
- \minusH [17](#), [1946](#)
- \minusV [17](#), [1947](#)
- \mirrorH [16](#), [1689](#)
- \mirrorV [16](#), [1690](#)
- \mlline [17](#), [1893](#)
- \Moon [21](#), [3121](#)
- \MoonB [21](#), [3125](#)
- \MoonBL [21](#), [3127](#)
- \MoonBR [21](#), [3128](#)
- \MoonL [21](#), [3123](#)
- \MoonR [21](#), [3122](#)
- \MoonT [21](#), [3124](#)
- \MoonTL [21](#), [3126](#)
- \MoonTR [21](#), [3129](#)
- \MPH [17](#), [1994](#)
- \MPV [17](#), [2026](#)
- N**
- \NeedsTeXFormat [2](#)
- nonogram (environment) .. [2379](#)
- \nonogramcolumn [18](#), [2240](#)
- \nonogramH [18](#), [2284](#)
- \nonogramrow [18](#), [2226](#)
- \nonogramsetup [18](#), [2375](#)
- \nonogramV [18](#), [2254](#)
- \nr [72](#),
[76](#), [98](#), [102](#), [1457](#), [1459](#),
[1468](#), [1470](#), [2791](#), [2793](#),
[2802](#), [2804](#)
- numberlink (environment) [2445](#)
- \numberlinkcell [18](#), [2427](#)
- \numberlinksetup [19](#), [2433](#)
- P**
- Package
- bokkusu.sty [6](#)
- logicpuzzle.sty .. [6](#), [7](#), [72](#)
- lpenv.sty [97](#)
- tikz.sty [7](#)
- \parkinglot [19](#), [2576](#)
- PGF layer
- LPbackground [7–9](#), [83](#), [88](#), [89](#),
[154](#)
- LPbackgroundtwo [7](#), [8](#)
- LPbgcolor [7](#), [8](#), [76](#), [78](#)
- LPdump [7](#), [8](#), [78](#), [112](#)
- LPforeground .. [7–9](#), [83](#), [89](#),
[144](#)
- LPforegroundtwo .. [7](#), [8](#), [83](#),
[118](#)
- main [7–9](#), [78](#), [117](#)
- \pgfdeclarelayer [12](#), [13](#), [14](#), [15](#),
[16](#), [17](#)
- \pgfmathsetlength [2613](#)
- \pgfmathsetmacro
 . [1722](#), [1732](#), [1745](#), [2875](#),
[2876](#), [2877](#), [2991](#)
- \pgfpoint [2497](#), [2507](#), [2673](#), [3288](#)
- \pgfsetcornersarced [2497](#), [2507](#),
[2673](#), [3288](#)
- \pgfsetlayers [18](#)
- \pitlane [19](#), [2597](#)
- \placearrow [16](#), [1711](#)
- \placecross [16](#), [1703](#)
- \placeisland [13](#), [898](#)
- \placemirror [16](#), [1707](#)
- \placesegment [12](#), [891](#), [897](#), [928](#),
[934](#), [940](#), [943](#), [948](#), [954](#),
[957](#)
- \placeship [12](#), [914](#)
- \placewater [13](#), [903](#)
- \plusH [17](#), [1960](#)
- \plusV [17](#), [1949](#)
- \PMH [17](#), [1978](#)
- \PMV [17](#), [2010](#)
- \portal [22](#), [3272](#)
- \ProvidesPackage [3](#)
- Puzzle option
- bgcolor [9](#), [24](#), [25](#),
[27](#), [28](#), [30](#), [32](#), [34](#), [35](#), [37](#),
[39](#), [41](#), [43](#), [45](#), [46](#), [48](#), [49](#),
[51](#), [53](#), [54](#), [56](#), [60](#), [62](#), [63](#),
[65](#), [66](#), [69](#), [70](#), [72](#)
- color . [9](#), [10](#), [27](#), [28](#), [32](#), [37](#),
[48](#), [51](#), [53](#), [54](#)
- columns [7–10](#), [23](#), [25](#),
[27](#), [28](#), [30](#), [32](#), [34](#), [35](#), [37](#),
[39](#), [41](#), [43](#), [44](#), [46](#), [47](#), [49](#),
[50](#), [52](#), [54](#), [56](#), [60](#), [61](#), [63](#),
[64](#), [69](#), [70](#), [72](#)

- counterstyle . 9, 24, 25, 27,
29, 30, 32, 34, 36, 37, 40,
42, 43, 45, 46, 48, 49, 51,
53, 54, 56, 60, 62, 63, 65,
66, 69, 70, 72
- cvoffset 9, 12, 24, 25, 27, 29,
31, 32, 34, 36, 37, 40, 42,
43, 45, 46, 48, 49, 51, 53,
54, 56, 60, 62, 63, 65, 66,
69, 70, 72
- extracells 51
- fontsize 8, 9, 24,
25, 27, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 56, 60, 61, 63,
65, 66, 69, 70, 72
- grid 29
- helplines 51
- rows 7–10, 23, 25,
26, 28, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 55, 60, 61, 63,
64, 68, 70, 71
- sbindent 25
- sbshipscale 25
- sbwidth 25
- scale 8–10, 23, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 56, 60, 61, 63,
65, 66, 69, 70, 72
- solution 37, 51
- sudoku 60
- title 8, 24, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 45–47, 49, 51,
52, 54, 56, 60, 61, 63, 65,
66, 69, 70, 72
- titleindent 8, 24, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 45–47, 49, 51,
52, 54, 56, 60, 61, 63, 65,
66, 69, 70, 72
- titlewidth 8, 24, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 45, 46, 48, 49,
51, 53, 54, 56, 60, 62, 63,
65, 66, 69, 70, 72
- width 8, 23, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 56, 60, 61, 63,
64, 66, 69, 70, 72
- puzzlebackground (environment)
..... 9, [566](#)
- \puzzlecounter 11, [609](#), [629](#), [636](#)
- puzzleforeground (environment)
..... 9, [573](#)
- \puzzlestrut 18, [727](#)
- R**
- resuko (environment) [2687](#)
- \resukocell 19, [2487](#)
- \resukosetup 19, [2489](#)
- \reversemarginpar 625
- \Right 21, [2997](#)
- \RightDown 21, [3004](#)
- \RightUp 21, [2998](#)
- \romannumeral 374, [398](#)
- rows (Puzzle option) 7–10, 23, 25,
26, 28, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 55, 60, 61, 63,
64, 68, 70, 71
- S**
- sbindent (Puzzle option) 25
- sbshipscale (Puzzle option) . 25
- sbwidth (Puzzle option) 25
- scale (Puzzle option) 8–10, 23, 25,
27, 28, 30, 32, 34, 35, 37,
39, 41, 43, 44, 46, 47, 49,
50, 52, 54, 56, 60, 61, 63,
65, 66, 69, 70, 72
- schatzsuche (environment) [2748](#)
- \schatzsuchecell [2728](#)
- \schatzsuchesetup ... 20, [2730](#)
- \setbigcell 9, [337](#)
- \setcell . 9, [314](#), [378](#), [402](#), [781](#),
[1097](#), [1198](#), [1245](#), [1360](#),
[1407](#), [1479](#), [1489](#), [1593](#),
[1640](#), [1793](#), [1887](#), [1944](#),
[2102](#), [2165](#), [2487](#), [2728](#),
[2817](#), [2867](#), [2928](#), [3046](#)
- \setcells 9, [323](#)

- `\setcolorcolumn` [10](#), [383](#)
- `\setcolorrow` [9](#), [359](#)
- `\setcolumn` [9](#), [309](#)
- `\setgridlinestyle` [12](#), [715](#),
[1166](#), [2890](#)
- `\setnormallinewidth` .. [12](#), [719](#)
- `\setpuzzlecounter` [11](#), [610](#)
- `\setrow` [9](#), [304](#)
- `\setrule` [10](#), [202](#)
- `\setthicklinewidth` ... [12](#), [723](#)
- `\setTikZpreset` [740](#)
- `\Ship` [843](#), [928](#)
- `\ship` [12](#), [897](#)
- `\ShipB` [846](#), [934](#)
- `\shipbox` [13](#), [851](#), [909](#), [979](#), [992](#),
[1006](#)
- `\ShipC` [844](#), [940](#), [954](#)
- `\shipH` [13](#), [882](#), [978](#), [984](#), [991](#), [997](#)
- `\ShipL` [847](#), [948](#)
- `\ShipR` [848](#), [957](#)
- `\ShipT` [845](#), [943](#)
- `\shipV` [13](#), [887](#), [977](#), [983](#), [990](#), [996](#)
- `skyline (environment)` ... [2823](#)
- `\skylineB` [20](#), [2813](#)
- `\skylinecell` [20](#), [2817](#)
- `\skylineL` [20](#), [2815](#)
- `\skylineR` [20](#), [2816](#)
- `\skylinesetup` [20](#), [2819](#)
- `\skylineT` [20](#), [2814](#)
- `slitherlink (environment)` [2884](#)
- `\slitherlinkcell` [20](#), [2867](#)
- `\slitherlinksetup` ... [20](#), [2869](#)
- `solution (Puzzle option)` . [37](#), [51](#)
- `\Star` [21](#), [21](#), [2987](#)
- `starbattle (environment)` [2934](#)
- `\starbattlecell` [20](#), [2928](#)
- `\starbattlesetup` [20](#), [2930](#)
- `starsandarrows (environment)` .
..... [3005](#)
- `\starsandarrowssetup` [21](#), [2978](#)
- `\starsH` [21](#), [2975](#)
- `\starsV` [21](#), [2976](#)
- `\START` [2875](#), [2879](#), [2880](#)
- `\Straight` [19](#), [2536](#)
- `\StraightH` [19](#), [2535](#)
- `\StraightV` [19](#), [2534](#), [2536](#)
- `sudoku (Puzzle option)` [60](#)
- `\sumH` [13](#), [1047](#)
- `\sumV` [13](#), [1048](#)
- `sunandmoon (environment)` [3130](#)
- `\sunandmooncell` [3046](#)
- `\sunandmoonsetup` [3048](#)
- `\sunandmoonssetup` [21](#)
- T**
- `\Tent` [22](#), [3210](#)
- `\tentH` [22](#), [3171](#)
- `tentsandtrees (environment)` ..
..... [3211](#)
- `\tentsandtreessetup` . [22](#), [3174](#)
- `\tentV` [22](#), [3172](#)
- `\theLP@columns` [598](#), [602](#)
- `\theLP@counter@unique` [211](#), [218](#),
[1526](#), [1539](#), [1542](#)
- `\theLP@counti` [954](#), [957](#)
- `\theLP@countii` [940](#), [943](#)
- `\theLP@puzzlecounter` [609](#)
- `\theLP@rows` [594](#)
- `\tikz` .. [213](#), [627](#), [634](#), [640](#), [843](#),
[844](#), [845](#), [846](#), [847](#), [848](#),
[849](#), [850](#), [852](#), [853](#), [854](#),
[855](#), [856](#), [857](#), [1528](#), [1723](#),
[2984](#), [2992](#), [3053](#), [3057](#)
- `tikz.sty (Package)` [7](#)
- `\tikzinputsegmentfirst` . [2678](#),
[2681](#)
- `\tikzinputsegmentlast` .. [2678](#),
[2683](#)
- `\tikzinputsegmentsupporta` ...
..... [2682](#)
- `\tikzinputsegmentupportb` ...
..... [2682](#)
- `\tikzpath` [453](#)
- `\tikzset` [742](#)
- `\tikzstyle` [841](#)
- `\tikzzpath` [11](#)
- `title (Puzzle option)` .. [8](#), [24](#), [25](#),
[27](#), [28](#), [30](#), [32](#), [34](#), [35](#), [37](#),
[39](#), [41](#), [43](#), [45–47](#), [49](#), [51](#),
[52](#), [54](#), [56](#), [60](#), [61](#), [63](#), [65](#),
[66](#), [69](#), [70](#), [72](#)
- `\titleformat` [11](#), [604](#), [608](#)
- `titleindent (Puzzle option)` [8](#), [24](#),
[25](#), [27](#), [28](#), [30](#), [32](#), [34](#), [35](#),
[37](#), [39](#), [41](#), [43](#), [45–47](#), [49](#),

51, 52, 54, 56, 60, 61, 63,
 65, 66, 69, 70, 72
 titlewidth (Puzzle option) 8, 24,
 25, 27, 28, 30, 32, 34, 35,
 37, 39, 41, 43, 45, 46, 48,
 49, 51, 53, 54, 56, 60, 62,
 63, 65, 66, 69, 70, 72
 \track 19, [2663](#)
 \trackH 19, [2651](#)
 \trackV 19, [2638](#)
 \Tree 22, [3199](#)
 \tube 22, [3278](#)
 tunnel (environment) 3300
 \tunnelH 22, [3252](#)
 \tunnelsetup 22, [3255](#)
 \tunnelV 22, [3253](#)

U

\Up 21, [2999](#)

V

\valueH 13, 1045
 \valueV 13, 1046

W

\Water 850, 906
 width (Puzzle option) .. 8, 23, 25,
 27, 28, 30, 32, 34, 35, 37,
 39, 41, 43, 44, 46, 47, 49,
 50, 52, 54, 56, 60, 61, 63,
 64, 66, 69, 70, 72

X

\xtikzpath 11, [484](#), 1307