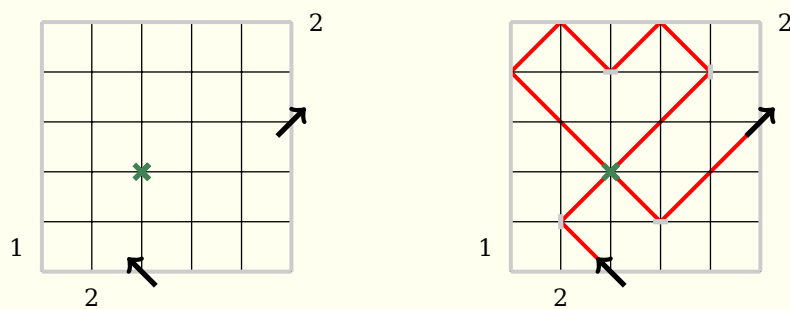


logicpuzzle.sty

v2.4

A style file for typesetting logic puzzles



June 6, 2013

Package author:
Josef Kleber

1	Supported puzzles	9
1.1	2D-Sudoku	9
1.1.1	Example	9
1.1.2	Options	10
1.2	Battleship	10
1.2.1	Example	10
1.2.2	Options	11
1.3	Bokkusu	12
1.3.1	Example	12
1.3.2	Options	13
1.4	Bridges	13
1.4.1	Example	14
1.4.2	Options	14
1.5	Chaos Sudoku	15
1.5.1	Example	15
1.5.2	Options	16
1.6	Four Winds	17
1.6.1	Example	17
1.6.2	Options	18
1.7	Hakyuu	18
1.7.1	Example	19
1.7.2	Options	20
1.8	Hitori	20
1.8.1	Example	21
1.8.2	Options	21
1.9	Kakuro	22
1.9.1	Example	22
1.9.2	Options	23
1.10	Kendoku	24
1.10.1	Example	24
1.10.2	Options	25
1.11	Killer Sudoku	26
1.11.1	Example	26
1.11.2	Options	27
1.12	Laser Beam	28
1.12.1	Example	28
1.12.2	Options	29
1.13	Magic Labyrinth	30
1.13.1	Example	30
1.13.2	Options	31
1.14	Magnets	31
1.14.1	Example	31
1.14.2	Options	32
1.15	Masyu	33
1.15.1	Example	33
1.15.2	Options	34

1.16 Minesweeper	34
1.16.1 Example	34
1.16.2 Options	35
1.17 Number Link	36
1.17.1 Example	36
1.17.2 Options	37
1.18 Resuko	37
1.18.1 Example	38
1.18.2 Options	39
1.19 Schatzsuche	39
1.19.1 Example	39
1.19.2 Options	40
1.20 Skyline	41
1.20.1 Example	41
1.20.1.1 Variants	42
1.20.1.1.1 Skyline Sudoku	42
1.20.1.1.2 Skyline Sudoku (N*N)	43
1.20.2 Options	44
1.21 Slitherlink	45
1.21.1 Example	45
1.21.2 Options	46
1.22 Star Battle	46
1.22.1 Example	46
1.22.2 Options	47
1.23 Stars and Arrows	48
1.23.1 Example	48
1.23.2 Options	49
1.24 Sudoku	50
1.24.1 Example	50
1.24.2 Options	51
1.24.3 Supporting bash scripts	51
1.24.3.1 createlpsudoku	51
1.24.3.2 lpsmag	51
1.25 Sun and Moon	52
1.25.1 Example	52
1.25.2 Options	53
1.26 Tents and Trees	53
1.26.1 Example	54
1.26.2 Options	54
1.27 Tunnel	55
1.27.1 Example	55
1.27.2 Options	56
2 Roll out your own grid-based logic puzzle	57
3 The code	58
3.1 PGF layers	58

3.2	Environments	59
3.2.1	Puzzle environments	59
3.2.1.1	logicpuzzle	59
3.2.1.1.1	Options	59
3.2.2	Supporting environments	60
3.2.2.1	puzzlebackground	60
3.2.2.2	puzzleforeground	60
3.3	Commands	60
3.3.1	Puzzle specific commands	60
3.3.1.1	2D-Sudoku	60
3.3.1.1.1	ddsudokucell	60
3.3.1.1.2	ddsudokusetup	60
3.3.1.2	Battleship	60
3.3.1.2.1	placeship	60
3.3.1.2.2	placesegment	61
3.3.1.2.3	ship	61
3.3.1.2.4	placewater	61
3.3.1.2.5	placeisland	61
3.3.1.2.6	shipH	61
3.3.1.2.7	shipV	61
3.3.1.2.8	shipbox	61
3.3.1.2.9	battleshipsetup	61
3.3.1.2.10	classicgame	61
3.3.1.3	Bokkusu	61
3.3.1.3.1	valueH	61
3.3.1.3.2	valueV	62
3.3.1.3.3	sumH	62
3.3.1.3.4	sumV	62
3.3.1.3.5	bokkususetup	62
3.3.1.4	Bridges	62
3.3.1.4.1	bridgesrow	62
3.3.1.4.2	bridgescolumn	62
3.3.1.4.3	bridge	62
3.3.1.4.4	bridgessetup	62
3.3.1.5	Chaos Sudoku	62
3.3.1.5.1	chaossudokucell	62
3.3.1.5.2	chaossudokusetup	62
3.3.1.6	Four Winds	63
3.3.1.6.1	fourwindscell	63
3.3.1.6.2	fourwindssetup	63
3.3.1.7	Hakyuu	63
3.3.1.7.1	hakyuucell	63
3.3.1.7.2	hakyuusetup	63
3.3.1.8	Hitori	63
3.3.1.8.1	hitorisetup	63
3.3.1.9	Kakuro	63
3.3.1.9.1	kakurorow	63

3.3.1.9.2	kakurocolumn	63
3.3.1.9.3	KKR	63
3.3.1.9.4	Black	63
3.3.1.9.5	kakurosetup	63
3.3.1.10	Kendoku	63
3.3.1.10.1	kendokucell	63
3.3.1.10.2	kendokusetup	64
3.3.1.11	Killer Sudoku	64
3.3.1.11.1	killersudokucell	64
3.3.1.11.2	killersudokusetup	64
3.3.1.12	Laser Beam	64
3.3.1.12.1	laserH	64
3.3.1.12.2	laserV	64
3.3.1.12.3	mirrorH	64
3.3.1.12.4	mirrorV	64
3.3.1.12.5	placearrow	64
3.3.1.12.6	placecross	64
3.3.1.12.7	placemirror	64
3.3.1.12.8	laser	65
3.3.1.12.9	laserbeamsetup	65
3.3.1.13	Magic Labyrinth	65
3.3.1.13.1	magiclabyrinthcell	65
3.3.1.13.2	mline	65
3.3.1.13.3	magiclabyrinthsetup	65
3.3.1.14	Magnets	65
3.3.1.14.1	plusH	65
3.3.1.14.2	minusH	65
3.3.1.14.3	plusV	65
3.3.1.14.4	minusV	65
3.3.1.14.5	magnetsH	65
3.3.1.14.6	magnetsV	66
3.3.1.14.7	Magnetic plates	66
3.3.1.14.8	magnetssetup	66
3.3.1.15	Masyu	66
3.3.1.15.1	masyucell	66
3.3.1.15.2	MasyuW	66
3.3.1.15.3	MasyuB	66
3.3.1.15.4	masyuline	66
3.3.1.15.5	masyusetup	66
3.3.1.16	Minesweeper	66
3.3.1.16.1	Mine	66
3.3.1.16.2	minesweeperssetup	66
3.3.1.17	Number Link	66
3.3.1.17.1	numberlinkcell	66
3.3.1.17.2	link	66
3.3.1.17.3	numberlinksetup	67

3.3.1.18	Resuko	67
3.3.1.18.1	resukocell	67
3.3.1.18.2	Track tiles	67
3.3.1.18.3	pitlane	67
3.3.1.18.4	parkinglot	67
3.3.1.18.5	trackH	67
3.3.1.18.6	trackV	67
3.3.1.18.7	track	67
3.3.1.18.8	resukosetup	68
3.3.1.19	Schatzsuche	68
3.3.1.19.1	Diamond	68
3.3.1.19.2	schatzsuchesetup	68
3.3.1.20	Skyline	68
3.3.1.20.1	skylineT	68
3.3.1.20.2	skylineB	68
3.3.1.20.3	skylineL	68
3.3.1.20.4	skylineR	68
3.3.1.20.5	skylinecell	68
3.3.1.20.6	skylinesetup	68
3.3.1.21	Slitherlink	68
3.3.1.21.1	slitherlinkcell	68
3.3.1.21.2	slitherlinksetup	68
3.3.1.22	Star Battle	69
3.3.1.22.1	starbattlecell	69
3.3.1.22.2	starbattlesetup	69
3.3.1.23	Stars and Arrows	69
3.3.1.23.1	starsH	69
3.3.1.23.2	starsV	69
3.3.1.23.3	Star	69
3.3.1.23.4	Arrows	69
3.3.1.23.5	starsandarrowssetup	69
3.3.1.24	Sudoku	69
3.3.1.24.1	lpsudokucell	69
3.3.1.24.2	lpsudokusetup	69
3.3.1.25	Sun and Moon	69
3.3.1.25.1	Star	69
3.3.1.25.2	Cloud	70
3.3.1.25.3	Howl at the Moon	70
3.3.1.25.4	sunandmoonssetup	70
3.3.1.26	Tents and Trees	70
3.3.1.26.1	tentH	70
3.3.1.26.2	tentV	70
3.3.1.26.3	Tree	70
3.3.1.26.4	Tent	70
3.3.1.26.5	tentsandtreessetup	70
3.3.1.27	Tunnel	70
3.3.1.27.1	tunnelH	70

3.3.1.27.2	tunnelV	70
3.3.1.27.3	portal	71
3.3.1.27.4	tube	71
3.3.1.27.5	tunnelsetup	71
3.3.2	User commands	71
3.3.2.1	In the grid	71
3.3.2.1.1	setcell	71
3.3.2.1.2	setcells	71
3.3.2.1.3	setbigcell	71
3.3.2.1.4	setrow	71
3.3.2.1.5	setcolorrow	71
3.3.2.1.6	setcolumn	71
3.3.2.1.7	setcolorcolumn	72
3.3.2.1.8	setrule	72
3.3.2.1.9	fillcell	72
3.3.2.1.10	fillrow	72
3.3.2.1.11	fillcolumn	72
3.3.2.1.12	filldiagonals	72
3.3.2.1.13	framearea	72
3.3.2.1.14	fillarea	72
3.3.2.1.15	colorarea	73
3.3.2.1.16	framepuzzle	73
3.3.2.1.17	tikzpath	73
3.3.2.1.18	xtikzpath	73
3.3.2.2	Presentation	73
3.3.2.2.1	titleformat	73
3.3.2.2.2	puzzlecounter	73
3.3.2.2.3	setpuzzlecounter	73
3.3.2.2.4	definecounterstyle	74
3.3.2.2.5	setgridlinestyle	74
3.3.2.2.6	setnormallinewidth	74
3.3.2.2.7	setthicklinewidth	74
3.3.3	Internal commands	74
3.3.3.1	Initialization	74
3.3.3.1.1	LP@define@key	74
3.3.3.1.2	LP@define@choicelkey@fontsize	74
3.3.3.2	Drawing grids	75
3.3.3.2.1	LP@drawgrid	75
3.3.3.2.2	LP@drawsudokugrid	75
3.3.3.2.3	LP@drawbackground	75
3.3.3.3	In the grid	75
3.3.3.3.1	LP@setcellcontent	75
3.3.3.3.2	LP@setcellcontentC	75
3.3.3.3.3	LP@setrowcontents	75
3.3.3.3.4	LP@setcolumncontents	75
3.3.3.3.5	LP@ingrid	75
3.3.3.3.6	LP@definecolor	75

3.3.3.4	Around the grid	76
3.3.3.4.1	LP@leftcolumn	76
3.3.3.4.2	LP@rightcolumn	76
3.3.3.4.3	LP@toprow	76
3.3.3.4.4	LP@bottomrow	76
3.3.3.5	Presentation	76
3.3.3.5.1	LP@drawcounter	76
4	Examples	76
	References	79
	Index	80

1 Supported puzzles

1.1 2D-Sudoku

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

1.1.1 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

```

1 \begin{center}
2   \begin{ddsudoku}
3     \framepuzzle
4     \filldiagonals[orange!50]
5     \ddsudokucell{1}{5}{1}
6     \ddsudokucell{1}{4}{3}
7     \ddsudokucell{2}{3}{4}
8     \ddsudokucell{4}{1}{3}
9     \ddsudokucell{4}{3}{2}
10    \ddsudokucell{5}{4}{4}
11  \end{ddsudoku}
12  \hspace{1.5cm}
13  \begin{ddsudoku}
14    \framepuzzle
15    \filldiagonals[orange!50]
16    \setrow{5}{1,3,4,5,2}
17    \setrow{4}{3,2,5,1,4}
18    \setrow{3}{5,4,3,2,1}
19    \setrow{2}{2,5,1,4,3}
20    \setrow{1}{4,1,2,3,5}
21  \end{ddsudoku}
22 \end{center}

```

1.1.2 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.

1.2 Battleship

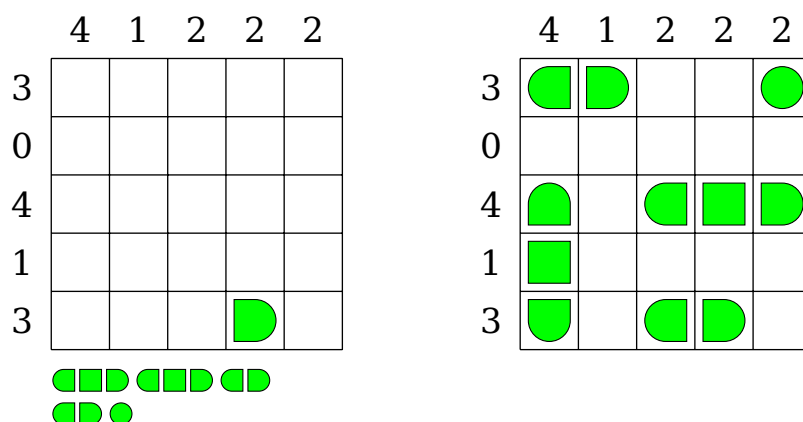
Try to find the positions of the ships listed below the puzzle. The numbers on the side of the puzzle reveals how many ship segments can be found in the rows and columns. All remaining fields indicate 'water'. Consider the following rules: The ships are arranged horizontally and vertically. No ship touches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

1.2.1 Example

```

1 \begin{center}
2   \begin{battleship}
3     \placesegment{4}{1}{\ShipR}
4     \shipH{4,1,2,2,2}
5     \shipV{3,1,4,0,3}
6     \shipbox{3,3,2,2,1}
7   \end{battleship}
8   \hspace{1.5cm}
9   \begin{battleship}
10    \placeship{V}{1}{1}{3}

```



```

11 \placeship{H}{1}{5}{2}
12 \placeship{H}{3}{1}{2}
13 \placeship{H}{3}{3}{3}
14 \placeship{H}{5}{5}{1}
15 \shipH{4,1,2,2,2}
16 \shipV{3,1,4,0,3}
17 \end{battleship}
18 \end{center}

```

1.2.2 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.

1.3 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.

1.3.1 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		

```

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

```

```

14 \fillrow{5}{0,0,1,0,0}
15 \fillrow{4}{1,0,1,1,1}
16 \fillrow{3}{1,0,0,1,0}
17 \fillrow{2}{0,0,1,1,1}
18 \fillrow{1}{0,1,0,0,0}
19 \end{bokkusu}
20 \end{center}

```

1.3.2 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.

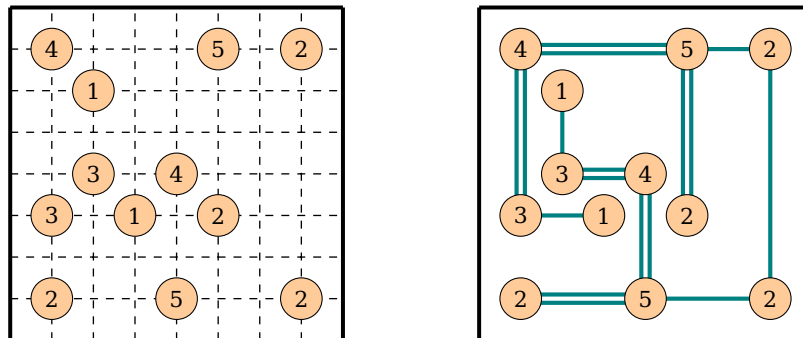
bgcolor [] sets the background color of the grid.

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

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

1.4 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.



1.4.1 Example

```

1 \colorlet{LP@cc@bridge}{Teal}
2 \begin{center}
3   \begin{bridges}
4     \framepuzzle
5     \bridgesrow{8}{{}},4,{{}},{{}},{{}},5,{{}},2}
6     \bridgesrow{7}{{}},{{}},1}
7     \bridgesrow{5}{{}},{{}},3,{{}},4}
8     \bridgesrow{4}{{}},3,{{}},1,{{}},2}
9     \bridgesrow{3}{{}},{{}},{{}},{{}},{{}},{{}},{{}}
10    \bridgesrow{2}{{}},2,{{}},{{}},5,{{}},{{}},2}
11  \end{bridges}
12  \hspace{1.5cm}
13  \begin{bridges}[grid=none]
14    \framepuzzle
15    \bridgesrow{8}{{}},4,{{}},{{}},{{}},5,{{}},2}
16    \bridgesrow{7}{{}},{{}},1}
17    \bridgesrow{5}{{}},{{}},3,{{}},4}
18    \bridgesrow{4}{{}},3,{{}},1,{{}},2}
19    \bridgesrow{3}{{}},{{}},{{}},{{}},{{}},{{}},{{}}
20    \bridgesrow{2}{{}},2,{{}},{{}},5,{{}},{{}},2}
21    \bridge[double]{\tikzpath{2}{4}{8,8,8,8,6,6,6,6,2,2,2,2}}
22    \bridge[double]{\tikzpath{2}{2}{6,6,6,8,8,8,4,4}}
23    \bridge{\tikzpath{2}{4}{6,6}}
24    \bridge{\tikzpath{3}{5}{8,8}}
25    \bridge{\tikzpath{5}{2}{6,6,6,8,8,8,8,8,4,4}}
26  \end{bridges}
27 \end{center}

```

1.4.2 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

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

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

1.5 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.

1.5.1 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

```

1 \begin{center}
2   \begin{chaossudoku}
3     \chaossudokucell{1}{1}{3}
4     \chaossudokucell{1}{5}{4}

```

```

5 \chaossudokucell{3}{2}{4}
6 \chaossudokucell{4}{2}{5}
7 \chaossudokucell{5}{5}{2}
8 \begin{puzzlebackground}
9 \fillarea{Wheat}{(1,1)--(1,2)--(2,2)--(2,3)--(4,3)--(4,1)
10 --(1,1)}
11 \fillarea{HotPink!30}{(1,2)--(1,6)--(3,6)--(3,5)--(2,5)
12 --(2,2)--(1,2)}
13 \fillarea{GreenYellow}{(2,3)--(2,5)--(3,5)--(3,4)--(5,4)
14 --(5,2)--(4,2)--(4,3)--(2,3)}
15 \fillarea{LightBlue}{(3,4)--(3,6)--(6,6)--(6,5)--(5,5)
16 --(5,4)--(3,4)}
17 \fillarea{LightYellow}{(4,1)--(4,2)--(5,2)--(5,5)--(6,5)
18 --(6,1)--(4,1)}
19 \end{puzzlebackground}
20 \end{chaossudoku}
21 \hspace{1.5cm}
22 \begin{chaossudoku}
23 \setrow{5}{4,3,5,1,2}
24 \setrow{4}{2,1,3,4,5}
25 \setrow{3}{5,4,2,3,1}
26 \setrow{2}{1,2,4,5,3}
27 \setrow{1}{3,5,1,2,4}
28 \begin{puzzlebackground}
29 \fillarea{Wheat}{(1,1)--(1,2)--(2,2)--(2,3)--(4,3)--(4,1)
30 --(1,1)}
31 \fillarea{HotPink!30}{(1,2)--(1,6)--(3,6)--(3,5)--(2,5)
32 --(2,2)--(1,2)}
33 \fillarea{GreenYellow}{(2,3)--(2,5)--(3,5)--(3,4)--(5,4)
34 --(5,2)--(4,2)--(4,3)--(2,3)}
35 \fillarea{LightBlue}{(3,4)--(3,6)--(6,6)--(6,5)--(5,5)
36 --(5,4)--(3,4)}
37 \fillarea{LightYellow}{(4,1)--(4,2)--(5,2)--(5,5)--(6,5)
38 --(6,1)--(4,1)}
39 \end{puzzlebackground}
40 \end{chaossudoku}
41 \end{center}

```

1.5.2 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.

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.

1.6 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.

1.6.1 Example

2				5
	3			
		3		
2				
			4	

2	T	┌───		5
┌	3	───┐		┌
┌	┌──	3	──┐	┌
2	──┐	┌	┌	┌
┌	┌──		4	──┐

```

1 \begin{fourwinds}
2   \framepuzzle
3   \fourwindscell{1}{2}{2}{}
4   \fourwindscell{1}{5}{2}{}
5   \fourwindscell{2}{4}{3}{}
6   \fourwindscell{3}{3}{3}{}
7   \fourwindscell{4}{1}{4}{}

```

```

8 \fourwindscell{5}{5}{5}{}
9 \end{fourwinds}
10 \hspace{1.5cm}
11 \begin{fourwinds}
12 \framepuzzle
13 \fourwindscell{1}{2}{2}{2/1,6/1}
14 \fourwindscell{1}{5}{2}{2/2}
15 \fourwindscell{2}{4}{3}{8/1,6/2}
16 \fourwindscell{3}{3}{3}{4/1,6/1,2/1}
17 \fourwindscell{4}{1}{4}{4/2,6/1,8/1}
18 \fourwindscell{5}{5}{5}{4/2,2/3}
19 \end{fourwinds}

```

1.6.2 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.

1.7 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.

1.7.1 Example

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

```

1 \begin{center}
2   \begin{hakyuu}
3     \hakyuucell{1}{5}{2}
4     \hakyuucell{3}{5}{6}
5     \hakyuucell{4}{5}{5}
6     \hakyuucell{4}{4}{4}
7     \hakyuucell{1}{3}{3}
8     \hakyuucell{2}{2}{2}
9     \hakyuucell{5}{2}{5}
10    \hakyuucell{4}{1}{1}
11    \begin{puzzlebackground}
12      \fillarea{Wheat}{(1,1)--(1,4)--(2,4)--(2,1)--(1,1)}
13      \fillarea{HotPink!30}{(1,4)--(1,6)--(6,6)--(6,5)--(3,5)
14        --(3,4)--(1,4)}
15      \fillarea{GreenYellow}{(2,4)--(3,4)--(3,5)--(5,5)--(5,4)
16        --(4,4)--(4,3)--(2,3)--(2,4)}
17      \fillarea{LightBlue}{(5,5)--(6,5)--(6,3)--(4,3)--(4,4)
18        --(5,4)--(5,5)}
19      \fillarea{LightSalmon!50}{(2,2)--(2,3)--(5,3)--(5,2)
20        --(2,2)}
21      \fillarea{LightYellow}{(2,1)--(2,2)--(5,2)--(5,3)--(6,3)
22        --(6,1)--(2,1)}
23    \end{puzzlebackground}
24  \end{hakyuu}
25  \hspace{1.5cm}
26  \begin{hakyuu}
27    \setrow{5}{2,3,6,5,4}
28    \setrow{4}{1,7,3,4,2}
29    \setrow{3}{3,1,2,1,3}
30    \setrow{2}{1,2,1,3,5}
31    \setrow{1}{2,3,4,1,2}
32    \begin{puzzlebackground}

```

```

33 \fillarea{Wheat}{(1,1)--(1,4)--(2,4)--(2,1)--(1,1)}
34 \fillarea{HotPink!30}{(1,4)--(1,6)--(6,6)--(6,5)--(3,5)
35 --(3,4)--(1,4)}
36 \fillarea{GreenYellow}{(2,4)--(3,4)--(3,5)--(5,5)--(5,4)
37 --(4,4)--(4,3)--(2,3)--(2,4)}
38 \fillarea{LightBlue}{(5,5)--(6,5)--(6,3)--(4,3)--(4,4)
39 --(5,4)--(5,5)}
40 \fillarea{LightSalmon!50}{(2,2)--(2,3)--(5,3)--(5,2)
41 --(2,2)}
42 \fillarea{LightYellow}{(2,1)--(2,2)--(5,2)--(5,3)--(6,3)
43 --(6,1)--(2,1)}
44 \end{puzzlebackground}
45 \end{hakyuu}
46 \end{center}

```

1.7.2 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.

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.

1.8 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.

1.8.1 Example

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

```

1 \begin{center}
2   \begin{hitori}
3     \framepuzzle
4     \setcolorrow{5}{2,4,2,1,1}
5     \setcolorrow{4}{1,3,2,4,1}
6     \setcolorrow{3}{1,3,3,3,2}
7     \setcolorrow{2}{4,2,1,3,3}
8     \setcolorrow{1}{4,1,2,2,3}
9   \end{hitori}
10  \hspace{1.5cm}
11  \begin{hitori}
12    \framepuzzle
13    \setcolorrow{5}{2,4,0,1,0}
14    \setcolorrow{4}{0,3,2,4,1}
15    \setcolorrow{3}{1,0,3,0,2}
16    \setcolorrow{2}{4,2,1,3,0}
17    \setcolorrow{1}{0,1,0,2,3}
18  \end{hitori}
19 \end{center}

```

1.8.2 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.

1.9 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.

1.9.1 Example

	23	16	10		
14					3
16					
14					
	8				

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

```

1 \definecolor{kakuro}{RGB}{155,206,167}
2 \kakurosetup{color=kakuro}
3 \begin{center}
4   \begin{kakuro}
5     \framepuzzle
6     \kakurorow{5}{\Black,\KKR{23}{},\KKR{16}{},\KKR{10}{},\Black}
7     \kakurorow{4}{\KKR{}}{14},9,1,4,\KKR{3}{}}
```

```

8      \kakurorow{3}{\KKR{}}{16},6,5,3,2}
9      \kakurorow{2}{\KKR{}}{14},8,3,2,1}
10     \kakurorow{1}{\Black,\KKR{}}{8},7,1,\Black}
11     \end{kakuro}
12     \hspace{1.5cm}
13     \begin{kakuro}[solution]
14         \framepuzzle
15         \kakurorow{5}{\Black,\KKR{23}}{\},\KKR{16}}{\},\KKR{10}}{\},\Black}
16         \kakurorow{4}{\KKR{}}{14},9,1,4,\KKR{3}}{\}}
17         \kakurorow{3}{\KKR{}}{16},6,5,3,2}
18         \kakurorow{2}{\KKR{}}{14},8,3,2,1}
19         \kakurorow{1}{\Black,\KKR{}}{8},7,1,\Black}
20     \end{kakuro}
21 \end{center}

```

1.9.2 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.

1.10 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.

1.10.1 Example

4+	2÷	75×		2
			2×	
5	60×			1
8×		2-	1-	
			8+	

4+	2÷	75×		2
1	4	3	5	2
3	2	5	2×	4
5	3	4	2	1
8×		2-	1-	
2	5	1	4	3
4	1	2	8+	5

```

1 \begin{center}
2   \begin{kendoku}
3     \framearea{black}{\tikzpath{1}{1}{8,8,6,2,6,2,4,4}}
4     \framearea{black}{\tikzpath{1}{3}{8,6,2,4}}
5     \framearea{black}{\tikzpath{1}{4}{8,8,6,2,2,4}}
6     \framearea{black}{\tikzpath{2}{2}{8,8,6,6,2,4,2,4}}
7     \framearea{black}{\tikzpath{2}{4}{8,8,6,2,2,4}}
8     \framearea{black}{\tikzpath{3}{1}{8,8,6,2,2,4}}
9     \framearea{black}{\tikzpath{3}{4}{8,8,6,6,2,4,2,4}}
10    \framearea{black}{\tikzpath{4}{1}{8,6,6,2,4,4}}
11    \framearea{black}{\tikzpath{4}{2}{8,6,6,2,4,4}}
12    \framearea{black}{\tikzpath{4}{3}{8,8,6,2,2,4}}
13    \framearea{black}{\tikzpath{5}{3}{8,6,2,4}}
14    \framearea{black}{\tikzpath{5}{4}{8,6,2,4}}
15    \framearea{black}{\tikzpath{5}{5}{8,6,2,4}}
16    \setrule{1}{2}{8\times}
17    \setrule{1}{3}{5}
18    \setrule{1}{5}{4+}
19    \setrule{2}{3}{60\times}
20    \setrule{2}{5}{2\div}
21    \setrule{3}{2}{2-}
22    \setrule{3}{5}{75\times}
23    \setrule{4}{1}{8+}

```



```

24 \setrule{4}{2}{1-}
25 \setrule{4}{4}{2\times}
26 \setrule{5}{3}{1}
27 \setrule{5}{5}{2}
28 \end{kendoku}
29 \hspace{1.5cm}
30 \begin{kendoku}
31 \framearea{black}{\tikzpath{1}{1}{8,8,6,2,6,2,4,4}}
32 \framearea{black}{\tikzpath{1}{3}{8,6,2,4}}
33 \framearea{black}{\tikzpath{1}{4}{8,8,6,2,2,4}}
34 \framearea{black}{\tikzpath{2}{2}{8,8,6,6,2,4,2,4}}
35 \framearea{black}{\tikzpath{2}{4}{8,8,6,2,2,4}}
36 \framearea{black}{\tikzpath{3}{1}{8,8,6,2,2,4}}
37 \framearea{black}{\tikzpath{3}{4}{8,8,6,6,2,4,2,4}}
38 \framearea{black}{\tikzpath{4}{1}{8,6,6,2,4,4}}
39 \framearea{black}{\tikzpath{4}{2}{8,6,6,2,4,4}}
40 \framearea{black}{\tikzpath{4}{3}{8,8,6,2,2,4}}
41 \framearea{black}{\tikzpath{5}{3}{8,6,2,4}}
42 \framearea{black}{\tikzpath{5}{4}{8,6,2,4}}
43 \framearea{black}{\tikzpath{5}{5}{8,6,2,4}}
44 \setrule{1}{2}{8\times}
45 \setrule{1}{3}{5}
46 \setrule{1}{5}{4+}
47 \setrule{2}{3}{60\times}
48 \setrule{2}{5}{2\div}
49 \setrule{3}{2}{2-}
50 \setrule{3}{5}{75\times}
51 \setrule{4}{1}{8+}
52 \setrule{4}{2}{1-}
53 \setrule{4}{4}{2\times}
54 \setrule{5}{3}{1}
55 \setrule{5}{5}{2}
56 \setrow{5}{1,4,3,5,2}
57 \setrow{4}{3,2,5,1,4}
58 \setrow{3}{5,3,4,2,1}
59 \setrow{2}{2,5,1,4,3}
60 \setrow{1}{4,1,2,3,5}
61 \end{kendoku}
62 \end{center}

```

1.10.2 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.

1.11 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.

1.11.1 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
			4

```

1 \begin{killersudoku}
2   \framearea{black}{\tikzpath{1}{1}{8,8,6,6,2,4,2,4}}
```

```

3   \framearea{black}{\tikzpath{1}{3}{8,8,6,2,2,4}}
4   \framearea{black}{\tikzpath{2}{1}{8,6,6,6,2,4,4,4}}
5   \framearea{black}{\tikzpath{2}{3}{8,8,6,2,6,2,4,4}}
6   \framearea{black}{\tikzpath{3}{2}{8,6,8,6,2,2,4,4}}
7   \framearea{black}{\tikzpath{3}{4}{8,6,6,2,4,4}}
8   \begin{puzzlebackground}
9       \colorarea{orange!20}{\tikzpath{1}{1}{8,8,6,6,2,2,4,4}}
10      \colorarea{orange!20}{\tikzpath{3}{3}{8,8,6,6,2,2,4,4}}
11  \end{puzzlebackground}
12  \setrule{1}{2}{7}
13  \setrule{1}{4}{7}
14  \setrule{2}{1}{9}
15  \setrule{2}{4}{6}
16  \setrule{3}{4}{5}
17  \setrule{4}{3}{6}
18  \end{killersudoku}
19  \hspace{1.5cm}
20  \begin{killersudoku}
21      \framearea{black}{\tikzpath{1}{1}{8,8,6,6,2,4,2,4}}
22      \framearea{black}{\tikzpath{1}{3}{8,8,6,2,2,4}}
23      \framearea{black}{\tikzpath{2}{1}{8,6,6,6,2,4,4,4}}
24      \framearea{black}{\tikzpath{2}{3}{8,8,6,2,6,2,4,4}}
25      \framearea{black}{\tikzpath{3}{2}{8,6,8,6,2,2,4,4}}
26      \framearea{black}{\tikzpath{3}{4}{8,6,6,2,4,4}}
27      \begin{puzzlebackground}
28          \colorarea{orange!20}{\tikzpath{1}{1}{8,8,6,6,2,2,4,4}}
29          \colorarea{orange!20}{\tikzpath{3}{3}{8,8,6,6,2,2,4,4}}
30      \end{puzzlebackground}
31      \setrule{1}{2}{7}
32      \setrule{1}{4}{7}
33      \setrule{2}{1}{9}
34      \setrule{2}{4}{6}
35      \setrule{3}{4}{5}
36      \setrule{4}{3}{6}
37      \setrow{4}{3,2,4,1}
38      \setrow{3}{4,1,3,2}
39      \setrow{2}{2,4,1,3}
40      \setrow{1}{1,3,2,4}
41  \end{killersudoku}

```

1.11.2 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.

bgcolor [] sets the background color of the grid.

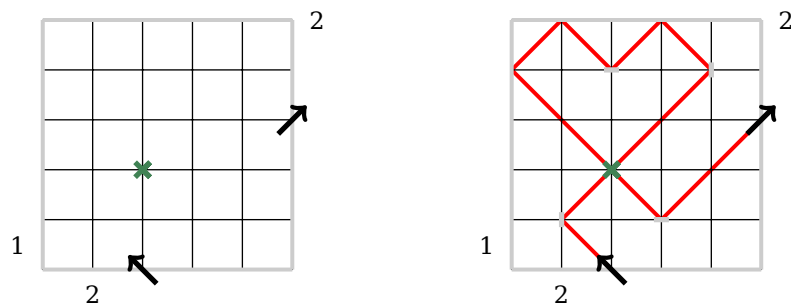
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.12 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.

1.12.1 Example



```

1 \begin{center}
2   \begin{laserbeam}
3     \laserV{1}
4     \laserH{[]}
5     \mirrorH{[],2}
6     \mirrorV{[],[],[],[],[],2}
7     \framepuzzle[LP@@mirror]
8     \placearrow{3}{1}{LeftUp}
9     \placearrow{6}{4}{RightUp}
10    \placecross{3}{3}
11  \end{laserbeam}
12  \hspace{1cm}
13  \begin{laserbeam}
14    \laserV{1}
15    \laserH{[]}
16    \mirrorH{[],2}
17    \mirrorV{[],[],[],[],[],2}
18    \framepuzzle[LP@@mirror]
19    \placearrow{3}{1}{LeftUp}
20    \placearrow{6}{4}{RightUp}
21    \placecross{3}{3}
22    \placemirror{2}{2}{V}
23    \placemirror{4}{2}{H}
24    \placemirror{5}{5}{V}
25    \placemirror{3}{5}{H}
26    \begin{puzzlebackground}
27      \laser{\tikzpath{3}{1}{7,9,9,9,7,1,7,1,3,3,3,9,9}}
28    \end{puzzlebackground}
29  \end{laserbeam}
30 \end{center}

```

1.12.2 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.

1.13 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.

1.13.1 Example

			3	
	3			1
			2	
3				

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

```

1 \begin{magiclabyrinth}
2   \mline{\xtikzpath{1}{6}{6/5,2/5,4/5,8/4,6/4,2/3,4/3,8/2,6/2,
3     2/1,4/1}}
4   \setcells{1/1,2/4,4/5}{3}
5   \magiclabyrinthcell{4}{2}{2}
6   \magiclabyrinthcell{5}{4}{1}
7 \end{magiclabyrinth}
8 \hspace{1.5cm}
9 \begin{magiclabyrinth}
10  \mline{\xtikzpath{1}{6}{6/5,2/5,4/5,8/4,6/4,2/3,4/3,8/2,6/2,
11    2/1,4/1}}
12  \setrow{5}{{},1,2,3}
13  \setrow{4}{2,3,{},{ },1}
14  \setrow{3}{{},{ },3,1,2}

```

```

15 \setrow{2}{1,{},{},2,3}
16 \setrow{1}{3,2,1}
17 \end{magiclabyrinth}

```

1.13.2 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

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

1.14 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.

1.14.1 Example

```

1 \magnetssetup{bgcolor=Teal!50}
2 \begin{magnets}
3   \minusH{2,1,2,2,1,2}
4   \minusV{2,1,3,1,2,1}
5   \plusH{2,1,2,2,2,1}

```

+		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	−			−	+	

```

6  \plusV{1,2,1,3,2,1}
7  \magnetsH{2/1,2/4,2/5,2/6,3/2,3/3,4/1,4/4,5/5,5/6}
8  \magnetsV{1/1,1/3,1/5,2/2,4/5,5/2,6/1,6/3}
9  \MPH{2/6}
10 \PMH{5/5}
11 \MPV{6/3}
12 \end{magnets}
13 \hspace{1.5cm}
14 \begin{magnets}
15   \minusH{2,1,2,2,1,2}
16   \minusV{2,1,3,1,2,1}
17   \plusH{2,1,2,2,2,1}
18   \plusV{1,2,1,3,2,1}
19   \MPH{2/6,3/2,4/1}
20   \PMH{2/5,5/5,4/4,3/3}
21   \MPV{1/1,1/3,6/3}
22 \end{magnets}

```

1.14.2 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.

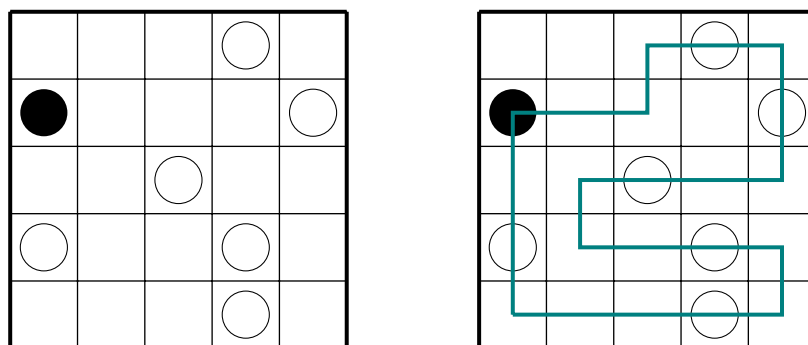
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.15 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.

1.15.1 Example



```

1 \masyusetup{color=Teal}
2 \begin{masyu}
3   \framepuzzle
4   \setcells{1/2,3/3,4/1,4/2,4/5,5/4}{\MasyuW}
5   \masyucell{1}{4}{\MasyuB}
6 \end{masyu}
7 \hspace{1.5cm}
8 \begin{masyu}
9   \framepuzzle
10  \setcells{1/2,3/3,4/1,4/2,4/5,5/4}{\MasyuW}
11  \masyucell{1}{4}{\MasyuB}
12  \masyuline{\xtikzpath{1}{1}{8/3,6/2,8/1,6/2,2/2,4/3,2/1,6/3,
13                2/1,4/4}}

```

```
14 \end{masyu}
```

1.15.2 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.








1.16 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.

1.16.1 Example

```
1 \begin{center}
2   \begin{minesweeper}
3     \framepuzzle
4     \setrow{5}{{}},1
5     \setrow{4}{{}},{ },3,3
6     \setrow{3}{3,{ },4,2}
7     \setrow{2}{{}},{ },{ },{ },0}
```

	1			
		3	3	
3		4	2	
				0
	2			

	1			
		3	3	
3		4	2	
				0
	2			

```

8   \setrow{1}{{}},2}
9   \end{minesweeper}
10  \hspace{1.5cm}
11  \begin{minesweeper}
12    \framepuzzle
13    \setrow{5}{{}},1,{},\Mine,\Mine}
14    \setrow{4}{{}},\Mine,3,3,\Mine}
15    \setrow{3}{3,\Mine,4,2}
16    \setrow{2}{{}},\Mine,\Mine,{},0}
17    \setrow{1}{{}},2}
18  \end{minesweeper}
19 \end{center}

```

1.16.2 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.

1.17 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.

1.17.1 Example

A		E	D	
B				
		A	E	
	C		D	
	B			C

A		E	D	
B				
		A	E	
	C		D	
	B			C

```

1 \begin{numberlink}
2   \framepuzzle
3   \numberlinkcell{2}{1}{B}
4   \numberlinkcell{5}{1}{C}
5   \numberlinkcell{2}{2}{C}
6   \numberlinkcell{4}{2}{D}
7   \numberlinkcell{3}{3}{A}
8   \numberlinkcell{4}{3}{E}
9   \numberlinkcell{1}{4}{B}
10  \numberlinkcell{1}{5}{A}
11  \numberlinkcell{3}{5}{E}
12  \numberlinkcell{4}{5}{D}
13 \end{numberlink}
14 \hspace{1.5cm}
15 \begin{numberlink}
16   \framepuzzle
17   \numberlinkcell{2}{1}{B}
18   \numberlinkcell{5}{1}{C}
19   \numberlinkcell{2}{2}{C}
20   \numberlinkcell{4}{2}{D}

```

```

21 \numberlinkcell{3}{3}{A}
22 \numberlinkcell{4}{3}{E}
23 \numberlinkcell{1}{4}{B}
24 \numberlinkcell{1}{5}{A}
25 \numberlinkcell{3}{5}{E}
26 \numberlinkcell{4}{5}{D}
27 \link{\tikzpath{4}{3}{8,4,8}}
28 \link{\tikzpath{2}{1}{4,8,8,8}}
29 \link{\tikzpath{2}{2}{6,2,6,6}}
30 \link{\tikzpath{1}{5}{6,2,2,6}}
31 \link{\tikzpath{4}{2}{6,8,8,8,4}}
32 \end{numberlink}

```

1.17.2 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

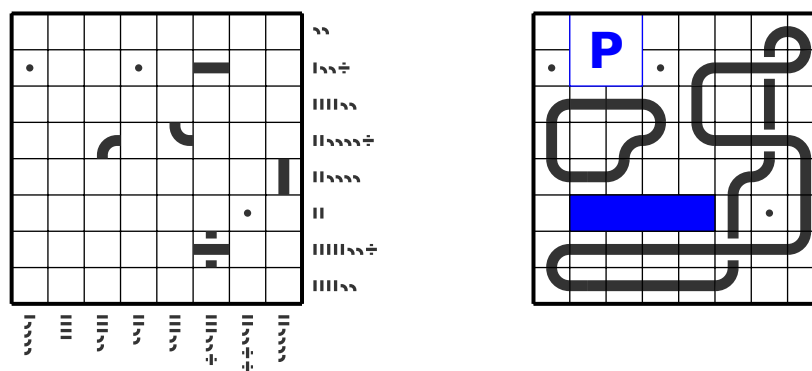
cvoffset [-23pt] sets the vertical offset of the counters in the margin.

1.18 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.

1.18.1 Example



```

1 \resukosetup{rows=8,columns=8,width=5.8cm,fontsize=Huge,
2             scale=.708}
3 \begin{resuko}[width=7.4cm]
4   \resukocell{1}{7}{\Graveltrap}
5   \resukocell{4}{7}{\Graveltrap}
6   \resukocell{7}{3}{\Graveltrap}
7   \resukocell{6}{2}{\Cross}
8   \resukocell{8}{4}{\StraightV}
9   \resukocell{6}{7}{\StraightH}
10  \resukocell{3}{5}{\CurveBR}
11  \resukocell{5}{5}{\CurveTR}
12  \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}
13  \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}
14  \framepuzzle
15 \end{resuko}
16 \hspace{1.5cm}
17 \begin{resuko}
18   \resukocell{1}{7}{\Graveltrap}
19   \resukocell{4}{7}{\Graveltrap}
20   \resukocell{7}{3}{\Graveltrap}
21   \parkinglot{2}{7}
22   \pitlane{2}{3}{H}
23   \track{\tikzpath{2}{4}{6,8,6,8,4,4,4,2,2,6}}
24   \track{\xtikzpath{2}{1}{6/4,8/3,6/1,8/4,6/1,2/1,4/3,2/2,6/3,
25             2/3,4/7,2/1,6/1}}
26   \framepuzzle

```

27 `\end{resuko}`

1.18.2 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 [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.

1.19 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.








1.19.1 Example

```

1 \begin{center}
2   \begin{schatzsuche}
3     \framepuzzle
4     \setrow{5}{{}},1}
5     \setrow{4}{{}},{ },3,3}
6     \setrow{3}{3,{ },4,2}

```

	1			
		3	3	
3		4	2	
				0
	2			

	1			
		3	3	
3		4	2	
				0
	2			

```

7   \setrow{2}{{},{},{},{},0}
8   \setrow{1}{{},{},2}
9   \end{schatzsuche}
10  \hspace{1.5cm}
11  \begin{schatzsuche}
12    \framepuzzle
13    \setrow{5}{{},{},1,{},{},\Diamond,\Diamond}
14    \setrow{4}{{},{},{},\Diamond,3,3,\Diamond}
15    \setrow{3}{3,\Diamond,4,2}
16    \setrow{2}{{},{},{},\Diamond,\Diamond,{},{},0}
17    \setrow{1}{{},{},{},2}
18  \end{schatzsuche}
19  \end{center}

```

1.19.2 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.

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.

1.20 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.

1.20.1 Example

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

```

1 \begin{center}
2   \begin{skyline}
3     \skylineB{3,{},3,1,{}}
4     \skylineL{{},4,3,{},{}}
5     \skylineT{{},{},2,{},3}
6     \skylineR{{},1,3,{},{}}
7     \skylinecell{1}{3}{2}
8     \skylinecell{4}{2}{3}
9   \end{skyline}
10  \hspace{1cm}
11  \begin{skyline}
12    \skylineB{3,{},3,1,{}}
13    \skylineL{{},4,3,{},{}}
14    \skylineT{{},{},2,{},3}
15    \skylineR{{},1,3,{},{}}
16    \setrow{5}{5,4,3,1,2}

```

```

17 \setrow{4}{4,5,1,2,3}
18 \setrow{3}{2,3,5,4,1}
19 \setrow{2}{1,2,4,3,5}
20 \setrow{1}{3,1,2,5,4}
21 \end{skyline}
22 \end{center}

```

1.20.1.1 Variants

1.20.1.1.1 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					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	

```

1 \begin{center}
2 \begin{skyline}[sudoku,scale=.4]
3 \skylineB{4,5,2,5,2,1,2,4,3}
4 \skylineL{4,2,2,2,1,3,3,3,2}
5 \skylineT{4,1,3,2,3,5,3,2,3}
6 \skylineR{3,3,1,3,4,2,3,2,4}
7 \setrow{9}{{},{},{},8,{},{},{},7}
8 \setrow{8}{{},{},{},4,{},{},6,{},{},8}
9 \setrow{7}{{},{},2,{},7,{},{},{},1}
10 \setrow{6}{{},{},{},{},8,2}
11 \setrow{5}{{},{},{},2,{},4,{},7}
12 \setrow{4}{{},{},{},{},3,{},{},4}
13 \setrow{3}{{},{},{},{},1}
14 \setrow{2}{{},{},3,{},{},{},{},1,2}
15 \setrow{1}{{},{},{},5}
16 \end{skyline}
17 \hspace{1cm}
18 \begin{skyline}[sudoku,scale=.4]
19 \skylineB{4,5,2,5,2,1,2,4,3}
20 \skylineL{4,2,2,2,1,3,3,3,2}
21 \skylineT{4,1,3,2,3,5,3,2,3}
22 \skylineR{3,3,1,3,4,2,3,2,4}
23 \setrow{9}{3,9,6,8,5,1,2,7,4}

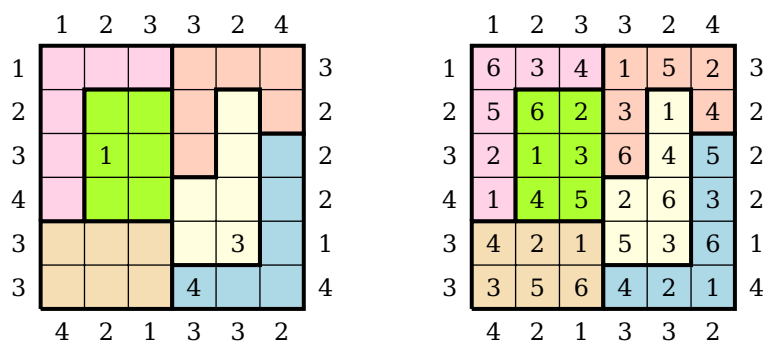
```

```

24 \setrow{8}{1,7,4,9,2,6,3,5,8}
25 \setrow{7}{5,2,8,7,3,4,9,6,1}
26 \setrow{6}{7,4,3,1,8,2,6,9,5}
27 \setrow{5}{9,8,2,6,4,5,7,1,3}
28 \setrow{4}{6,5,1,3,9,7,4,8,2}
29 \setrow{3}{8,6,7,2,1,3,5,4,9}
30 \setrow{2}{4,3,9,5,6,8,1,2,7}
31 \setrow{1}{2,1,5,4,7,9,8,3,6}
32 \end{skyline}
33 \end{center}

```

1.20.1.1.2 Skyline Sudoku (N*N)



```

1 \begin{center}
2 \begin{skyline}[rows=6,columns=6,scale=.58]
3 \skylineB{4,2,1,3,3,2}
4 \skylineL{3,3,4,3,2,1}
5 \skylineT{1,2,3,3,2,4}
6 \skylineR{4,1,2,2,2,3}
7 \skylinecell{2}{4}{1}
8 \skylinecell{4}{1}{4}
9 \skylinecell{5}{2}{3}
10 \begin{puzzlebackground}
11 \fillarea{Wheat}{(1,1)--(1,3)--(4,3)--(4,1)--(1,1)}
12 \fillarea{HotPink!30}{(1,3)--(1,7)--(4,7)--(4,6)--(2,6)
13 --(2,3)--(1,3)}
14 \fillarea{GreenYellow}{(2,3)--(2,6)--(4,6)--(4,3)--(2,3)}
15 \fillarea{LightBlue}{(4,1)--(7,1)--(7,5)--(6,5)--(6,2)
16 --(4,2)--(4,1)}
17 \fillarea{LightSalmon!50}{(4,7)--(4,4)--(5,4)--(5,6)--(6,6)
18 --(6,5)--(7,5)--(7,7)--(4,7)}
19 \fillarea{LightYellow}{(4,2)--(4,4)--(5,4)--(5,6)--(6,6)
20 --(6,2)--(4,2)}
21 \end{puzzlebackground}
\end{center}

```

```

22 \end{skyline}
23 \hspace{1cm}
24 \begin{skyline}[rows=6,columns=6,scale=.58]
25   \skylineB{4,2,1,3,3,2}
26   \skylineL{3,3,4,3,2,1}
27   \skylineT{1,2,3,3,2,4}
28   \skylineR{4,1,2,2,2,3}
29   \setrow{6}{6,3,4,1,5,2}
30   \setrow{5}{5,6,2,3,1,4}
31   \setrow{4}{2,1,3,6,4,5}
32   \setrow{3}{1,4,5,2,6,3}
33   \setrow{2}{4,2,1,5,3,6}
34   \setrow{1}{3,5,6,4,2,1}
35   \begin{puzzlebackground}
36     \fillarea{Wheat}{(1,1)--(1,3)--(4,3)--(4,1)--(1,1)}
37     \fillarea{HotPink!30}{(1,3)--(1,7)--(4,7)--(4,6)--(2,6)
38                          --(2,3)--(1,3)}
39     \fillarea{GreenYellow}{(2,3)--(2,6)--(4,6)--(4,3)--(2,3)}
40     \fillarea{LightBlue}{(4,1)--(7,1)--(7,5)--(6,5)--(6,2)
41                        --(4,2)--(4,1)}
42     \fillarea{LightSalmon!50}{(4,7)--(4,4)--(5,4)--(5,6)--(6,6)
43                          --(6,5)--(7,5)--(7,7)--(4,7)}
44     \fillarea{LightYellow}{(4,2)--(4,4)--(5,4)--(5,6)--(6,6)
45                          --(6,2)--(4,2)}
46   \end{puzzlebackground}
47 \end{skyline}
48 \end{center}

```

1.20.2 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.

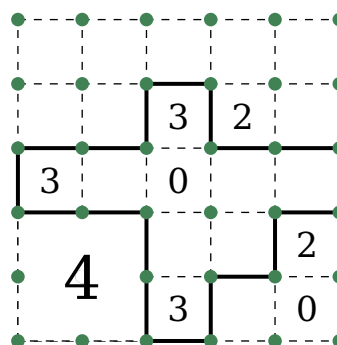
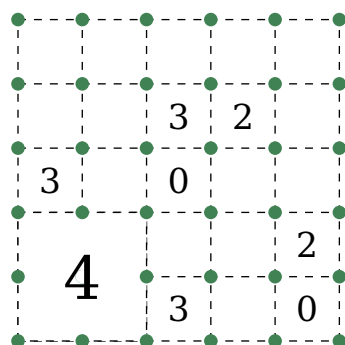
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.21 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.

1.21.1 Example



```

1 \begin{center}
2   \begin{slitherlink}
3     \setbigcell{1}{1}{4}
4     \slitherlinkcell{1}{3}{3}
5     \slitherlinkcell{3}{1}{3}
6     \slitherlinkcell{3}{3}{0}
7     \slitherlinkcell{3}{4}{3}
8     \slitherlinkcell{4}{4}{2}
9     \slitherlinkcell{5}{1}{0}
10    \slitherlinkcell{5}{2}{2}
11  \end{slitherlink}
12  \hspace{1.5cm}
13  \begin{slitherlink}
14    \setbigcell{1}{1}{4}
15    \slitherlinkcell{1}{3}{3}

```

```

16 \slitherlinkcell{3}{1}{3}
17 \slitherlinkcell{3}{3}{0}
18 \slitherlinkcell{3}{4}{3}
19 \slitherlinkcell{4}{4}{2}
20 \slitherlinkcell{5}{1}{0}
21 \slitherlinkcell{5}{2}{2}
22 \framearea{black}{\tikzpath{3}{1}{8,8,4,4,8,6,6,8,6,2,
23                                     6,6,2,4,2,4,2,4}}
24 \end{slitherlink}
25 \end{center}

```

1.21.2 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.

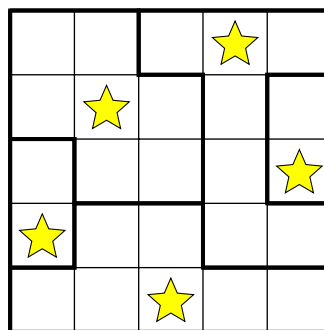
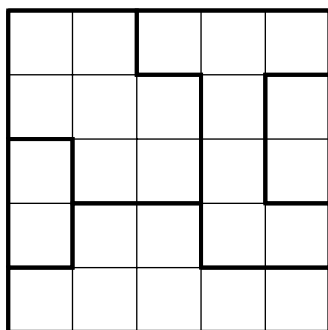
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.22 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.

1.22.1 Example



```

1 \begin{center}
2   \begin{starbattle}
3     \framepuzzle
4     \framearea{black}{\tikzpath{1}{1}{8,6,8,6,6,2,6,6,2,4,4,4,
5                                   4,4}}
6     \framearea{black}{\tikzpath{1}{2}{8,8,6,2,2,4}}
7     \framearea{black}{\tikzpath{1}{4}{8,8,6,6,2,6,2,2,4,4,8,4}}
8     \framearea{black}{\tikzpath{4}{2}{8,8,8,4,8,6,6,6,2,4,2,2,6,
9                                   2,4,4}}
10    \framearea{black}{\tikzpath{5}{3}{8,8,6,2,2,4}}
11  \end{starbattle}
12  \hspace{1.5cm}
13  \begin{starbattle}
14    \framepuzzle
15    \framearea{black}{\tikzpath{1}{1}{8,6,8,6,6,2,6,6,2,4,4,4,
16                                  4,4}}
17    \framearea{black}{\tikzpath{1}{2}{8,8,6,2,2,4}}
18    \framearea{black}{\tikzpath{1}{4}{8,8,6,6,2,6,2,2,4,4,8,4}}
19    \framearea{black}{\tikzpath{4}{2}{8,8,8,4,8,6,6,6,2,4,2,2,6,
20                                  2,4,4}}
21    \framearea{black}{\tikzpath{5}{3}{8,8,6,2,2,4}}
22    \starbattlecell{1}{2}{\Star}
23    \starbattlecell{2}{4}{\Star}
24    \starbattlecell{3}{1}{\Star}
25    \starbattlecell{4}{5}{\Star}
26    \starbattlecell{5}{3}{\Star}
27  \end{starbattle}
28 \end{center}

```

1.22.2 Options

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

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

width [5.1cm] 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.

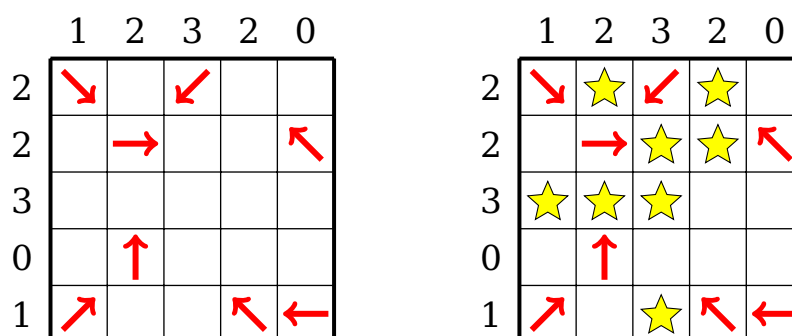
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.23 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.

1.23.1 Example



```

1 \begin{center}
2 \begin{starsandarrows}

```



```

3   \framepuzzle
4   \starsH{1,2,3,2,0}
5   \starsV{1,0,3,2,2}
6   \setrow{5}{\RightDown,{},{},\LeftDown}
7   \setrow{4}{{},{},\Right,{},{},\LeftUp}
8   \setrow{2}{{},{},\Up,{},{},{}}
9   \setrow{1}{\RightUp,{},{},\LeftUp,\Left}
10  \end{starsandarrows}
11  \hspace{1.5cm}
12  \begin{starsandarrows}
13    \framepuzzle
14    \starsH{1,2,3,2,0}
15    \starsV{1,0,3,2,2}
16    \setrow{5}{\RightDown,\Star,\LeftDown,\Star}
17    \setrow{4}{{},{},\Right,\Star,\Star,\LeftUp}
18    \setrow{3}{\Star,\Star,\Star}
19    \setrow{2}{{},{},\Up,{},{},{}}
20    \setrow{1}{\RightUp,{},{},\Star,\LeftUp,\Left}
21  \end{starsandarrows}
22 \end{center}

```

1.23.2 Options

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

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

width [5.9cm] 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.

1.24 Sudoku

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

1.24.1 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

```

1 \begin{center}
2   \begin{lsudoku}
3     \setrow{9}{{}},2,6,{{}},{{}},{{}},{{}},{{}},{{}}
4     \setrow{8}{{}},{{}},{{}},{{}},{{}},{{}},1,7,{{}}
5     \setrow{7}{{}},{{}},3,1,{{}},6,{{}},{{}},{{}}
6     \setrow{6}{{}},6,{{}},{{}},5,{{}},8,{{}},3
7     \setrow{5}{{}},{{}},9,2,6,1,7,{{}},{{}}
8     \setrow{4}{5,{{}},4,{{}},8,{{}},{{}},6,{{}}
9     \setrow{3}{{}},{{}},{{}},8,{{}},4,3,{{}},{{}}
10    \setrow{2}{{}},4,8,{{}},{{}},{{}},{{}},{{}},{{}}
11    \setrow{1}{{}},{{}},{{}},{{}},{{}},{{}},{{}},9,4,{{}}
12    \end{lsudoku}
13    \hspace{1.5cm}
14    \begin{lsudoku}
15      \setrow{9}{1,2,6,5,7,8,4,3,9}
16      \setrow{8}{4,8,5,9,3,2,1,7,6}
17      \setrow{7}{7,9,3,1,4,6,5,8,2}
18      \setrow{6}{2,6,1,4,5,7,8,9,3}
19      \setrow{5}{8,3,9,2,6,1,7,5,4}
20      \setrow{4}{5,7,4,3,8,9,2,6,1}
21      \setrow{3}{6,5,2,8,9,4,3,1,7}
22      \setrow{2}{9,4,8,7,1,3,6,2,5}
23      \setrow{1}{3,1,7,6,2,5,9,4,8}
24    \end{lsudoku}

```

1.24.2 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.

1.24.3 Supporting bash scripts

1.24.3.1 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 format¹ (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

1.24.3.2 lpsmag

With the lpsmag [30] bash script you can half automatically produce a Sudoku magazine using the L^AT_EX package lpsudoku.sty and the createlpsudoku bash script.

¹processing of several sudokus in 81 format (one in each line) is possible

Usage: `lpsmag configfile`

The script needs an installed [QQwing](#) [31] and a config file for defining the magazine's contents:

```

1 page p1 easy
2 page p2 easy
3 startpuzzles
4 typesetpage p1
5 typesetpage p2
6 startsolutions
7 typesetsolpage p1 p2 last

```

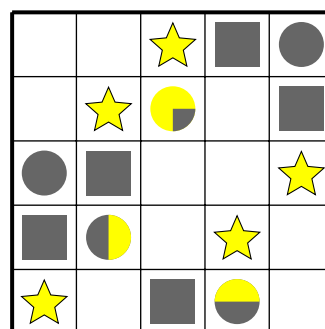
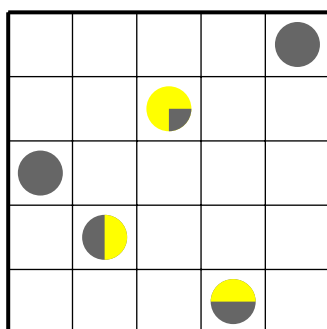
This config file will be sourced into the `lpsmag` bash script and contains calls of `lpsmag` functions. Make sure, that the config file has UNIX line endings (LF). For a detailed documentation I refer to the following [wiki](#) [30] entry.

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](#).

1.25 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.

1.25.1 Example



```

1 \begin{center}
2   \begin{sunandmoon}
3     \framepuzzle

```

```

4      \setrow{5}{{},{},{},{},\Moon}
5      \setrow{4}{{},{},\MoonTL}
6      \setrow{3}{\Moon}
7      \setrow{2}{{},{},\MoonR}
8      \setrow{1}{{},{},{},\MoonT}
9      \end{sunandmoon}
10     \hspace{1.5cm}
11     \begin{sunandmoon}
12       \framepuzzle
13       \setrow{5}{{},{},\Star,\Cloud,\Moon}
14       \setrow{4}{{},{},\Star,\MoonTL,{},\Cloud}
15       \setrow{3}{\Moon,\Cloud,{},{},\Star}
16       \setrow{2}{\Cloud,\MoonR,{},\Star}
17       \setrow{1}{\Star,{},\Cloud,\MoonT}
18     \end{sunandmoon}
19 \end{center}

```

1.25.2 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

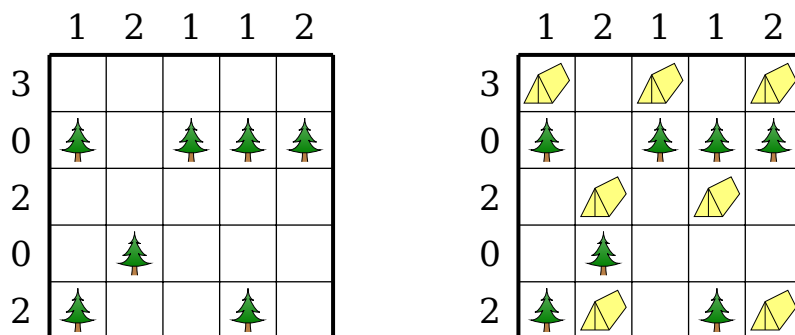
cvoffset [-23pt] sets the vertical offset of the counters in the margin.

1.26 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.

1.26.1 Example



```

1 \begin{center}
2   \begin{tentsandtrees}
3     \framepuzzle
4     \tentH{1,2,1,1,2}
5     \tentV{2,0,2,0,3}
6     \setrow{4}{\Tree, {}, \Tree, \Tree, \Tree}
7     \setrow{2}{{}}, \Tree}
8     \setrow{1}{\Tree, {}, {}, \Tree}
9   \end{tentsandtrees}
10  \hspace{1.5cm}
11  \begin{tentsandtrees}
12    \framepuzzle
13    \tentH{1,2,1,1,2}
14    \tentV{2,0,2,0,3}
15    \setrow{5}{\Tent, {}, \Tent, {}, \Tent,}
16    \setrow{4}{\Tree, {}, \Tree, \Tree, \Tree}
17    \setrow{3}{{}}, \Tent, {}, \Tent}
18    \setrow{2}{{}}, \Tree}
19    \setrow{1}{\Tree, \Tent, {}, \Tree, \Tent}
20  \end{tentsandtrees}
21 \end{center}

```

1.26.2 Options

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

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

width [5.9cm] 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.

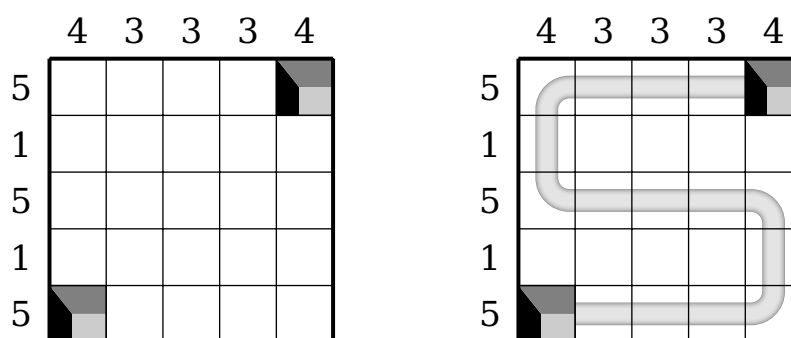
counterstyle [none] defines the counter style. Predefined styles: none, left, right

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

1.27 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!

1.27.1 Example



```

1 \begin{center}
2   \begin{tunnel}
3     \framepuzzle
4     \tunnelH{4,3,3,3,4}

```

```

5      \tunnelV{5,1,5,1,5}
6      \portal{1}{1}
7      \portal{5}{5}
8      \end{tunnel}
9      \hspace{1.5cm}
10     \begin{tunnel}
11       \framepuzzle
12       \tunnelH{4,3,3,3,4}
13       \tunnelV{5,1,5,1,5}
14       \portal{1}{1}
15       \portal{5}{5}
16       \tube{\tikzpath{1}{1}{6,6,6,6,8,8,4,4,4,4,8,8,6,6,6,6}}
17     \end{tunnel}
18 \end{center}

```

1.27.2 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.

2 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.

3 The code

3.1 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

[see: 3.2.2.1] and puzzleforeground [see: 3.2.2.2] 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.

3.2 Environments

3.2.1 Puzzle environments

3.2.1.1 logicpuzzle

```
\begin{logicpuzzle}{options}
...
\end{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 puzzle environments described in section 1. 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.

3.2.1.1.1 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.

3.2.2 Supporting environments

3.2.2.1 puzzlebackground

`\begin{puzzlebackground}`
`...`
`\end{puzzlebackground}` The puzzlebackground environment allows you to place elements behind the main layer on the LPbackground layer [see: 3.1]. This is for example usefull for the `\fillarea` [see: 3.3.2.1.14] command.

3.2.2.2 puzzleforeground

`\begin{puzzleforeground}`
`...`
`\end{puzzleforeground}` The puzzleforeground environment allows you to place elements in front of the main layer on the LPforeground layer [see: 3.1]. This is for example usefull for the `\framearea` [see: 3.3.2.1.13] command.

3.3 Commands

3.3.1 Puzzle specific commands

3.3.1.1 2D-Sudoku







`\ddsudokucell{<column>}{<row>}`
`{<number>}` **3.3.1.1.1 ddsudokucell** The command `\ddsudokucell` sets the `<number>` of the grid cell `<column><row>`.

`\ddsudokusetup{<options>}` **3.3.1.1.2 ddsudokusetup** With the command `\ddsudokusetup` you can reset the options with global scope.

3.3.1.2 Battleship





`\placeship{<direction>}`
`{<column>}{<row>}{<length>}` **3.3.1.2.1 placeship** With the command `\placeship` you can place 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.

`\placesegment{⟨column⟩}{⟨row⟩}`
`{⟨ship segment⟩}` **3.3.1.2.2 placesegment** The command `\placesegment` is used for the placement of ship segments in the grid. In the mandatory argument *⟨ship segment⟩*, you can use the following commands:

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

3.3.1.2.3 ship The command `\ship` was replaced by the `\placesegment` command. The command `\ship` is deprecated and should not be used longer. It may still be used, but it is not recommended.

`\placewater{⟨column⟩}{⟨row⟩}` **3.3.1.2.4 placewater** With the command `\placewater` you can place water markers (•) in the grid.

`\placeisland{⟨column⟩}{⟨row⟩}` **3.3.1.2.5 placeisland** With the command `\placeisland` you can place islands () in the grid. The island outlines are created randomly:    ...

`\shipH{⟨csv list⟩}` **3.3.1.2.6 shipH** The command `\shipH` typesets the horizontal numbers above the grid. It expects a comma-separated list as an argument.

`\shipV{⟨csv list⟩}` **3.3.1.2.7 shipV** The command `\shipV` typesets the vertical numbers beside the grid. It also expects a comma separated list.

`\shipbox{⟨csv list⟩}` **3.3.1.2.8 shipbox** The command `\shipbox` defines the number and size of the ships, which are typeset under the grid.

`\battleshipsetup{⟨options⟩}` **3.3.1.2.9 battleshipsetup** With the command `\battleshipsetup` you can reset the options with global scope.

`\classicgame{⟨csv list⟩}` **3.3.1.2.10 classicgame** The command `\classicgame` typesets a game sheet for playing classic Battleship. It expects a comma separated list with the number and sizes of the ships.

3.3.1.3 Bokkusu

`\valueH{⟨csv list⟩}` **3.3.1.3.1 valueH** The command `\valueH` typesets the numbers left to the grid indicating the values of the cells. It expects a comma-separated list as an argument.

`\valueV{<csv list>}` **3.3.1.3.2 valueV** The command `\valueV` typesets the numbers below the grid specifying the values of the cells. It also expects a comma separated list.

`\sumH{<csv list>}` **3.3.1.3.3 sumH** The command `\sumH` typesets the numbers right to the grid indicating the sums of the values of the colored cells. It expects a comma-separated list.

`\sumV{<csv list>}` **3.3.1.3.4 sumV** The command `\sumV` typesets the numbers above the grid specifying the sums of the values of the colored cells. It expects a comma separated list.

`\bokkususetup{<options>}` **3.3.1.3.5 bokkususetup** With the command `\bokkususetup` you can reset the options with global scope.

3.3.1.4 Bridges

`\bridgesrow{<row>}{<csv list>}` **3.3.1.4.1 bridgesrow** With the `\bridgesrow` command, you can set the contents of a bridges `<row>`. These are the numbers indicating how many bridges originate from this specific island.

`\bridgescolumn{<column>}{<csv list>}` **3.3.1.4.2 bridgescolumn** With the `\bridgescolumn` command, you can set the contents of a bridges `<column>`.

`\bridge{<optional arguments>}`
`{<TikZ path>}` **3.3.1.4.3 bridge** With the `\bridge` command, you can draw the bridges between islands. With the optional argument `<double>` you can draw a double bridge. Furthermore, you can set the color of the bridge with the option `<color>`.

`\bridgessetup{<options>}` **3.3.1.4.4 bridgessetup** With the command `\bridgessetup` you can reset the options with global scope.

3.3.1.5 Chaos Sudoku

`\chaossudokucell{<column>}`
`{<row>}{<number>}` **3.3.1.5.1 chaossudokucell** With the command `\chaossudokucell`, you can set the `<number>` of the grid cell `<column>` `<row>`.

`\chaossudokusetup{<options>}` **3.3.1.5.2 chaossudokusetup** With the command `\chaossudokusetup` you can reset the options with global scope.

3.3.1.6 Four Winds

`\fourwindscell{⟨column⟩}{⟨row⟩}`
`{⟨number⟩}{⟨csv list⟩}` **3.3.1.6.1 fourwindscell** The command `\fourwindscell` sets the *⟨number⟩* into grid cell *⟨column⟩⟨row⟩*. Furthermore, it draws lines specified in *⟨csv list⟩* in the direction/length format.

`\fourwindssetup{⟨options⟩}` **3.3.1.6.2 fourwindssetup** With the command `\fourwindssetup` you can reset the options with global scope.

3.3.1.7 Hakyuu

`\hakyuucell{⟨column⟩}{⟨row⟩}`
`{⟨number⟩}` **3.3.1.7.1 hakyuucell** The command `\hakyuucell` sets the *⟨number⟩* of the grid cell *⟨column⟩⟨row⟩*.

`\hakyuusetup{⟨options⟩}` **3.3.1.7.2 hakyuusetup** With the command `\hakyuusetup` you can reset the options with global scope.

3.3.1.8 Hitori

`\hitorisetup{⟨options⟩}` **3.3.1.8.1 hitorisetup** With the command `\hitorisetup` you can reset the options with global scope.

3.3.1.9 Kakuro

`\kakurorow{⟨row⟩}{⟨csv list⟩}` **3.3.1.9.1 kakurorow** With the `\kakurorow` command, you can set the contents of a kakuro *⟨row⟩*. These may be numbers and the commands `\KKR` or `\Black`.

`\kakurocolumn{⟨column⟩}{⟨csv list⟩}` **3.3.1.9.2 kakurocolumn** With the `\kakurocolumn` command, you can set the contents of a kakuro *⟨column⟩*.

`\KKR{⟨sumV⟩}{⟨sumH⟩}` **3.3.1.9.3 KKR** With the `\KKR` command, you can set the contents of a kakuro cell.

`\Black` **3.3.1.9.4 Black** The command `\Black` blacks out a cell.

`\kakurosetup{⟨options⟩}` **3.3.1.9.5 kakurosetup** With the command `\kakurosetup` you can reset the options with global scope.

3.3.1.10 Kendoku

`\kendokucell{⟨column⟩}{⟨row⟩}`
`{⟨number⟩}` **3.3.1.10.1 kendokucell** The command `\kendokucell` sets the *⟨number⟩* of the grid cell *⟨column⟩⟨row⟩*.

`\kendokusetup{<options>}` **3.3.1.10.2 kendokusetup** With the command `\kendokusetup` you can reset the options with global scope.

3.3.1.11 Killer Sudoku

`\killersudokucell{<column>}{<row>}{<number>}` **3.3.1.11.1 killersudokucell** The command `\killersudokucell` sets the `<number>` of the grid cell `<column>` `<row>`.

`\killersudokusetup{<options>}` **3.3.1.11.2 killersudokusetup** With the command `\killersudokusetup` you can reset the options with global scope.

3.3.1.12 Laser Beam

`\laserH{<csv list>}` **3.3.1.12.1 laserH** The command `\laserH` 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{<csv list>}` **3.3.1.12.2 laserV** The command `\laserV` typesets the numbers left to the grid.

`\mirrorH{<csv list>}` **3.3.1.12.3 mirrorH** The command `\mirrorH` typesets the numbers below the grid indicating how many mirrors are placed in the intersections of this column.

`\mirrorV{<csv list>}` **3.3.1.12.4 mirrorV** The command `\mirrorV` typesets the numbers right to the grid.

`\placearrow{<column>}{<row>}{<direction>}` **3.3.1.12.5 placearrow** The command `\placearrow` 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{<column>}{<row>}` **3.3.1.12.6 placecross** With the command `\placecross` you can place a cross in the intersections of the grid.

`\placemirror{<column>}{<row>}{<direction>}` **3.3.1.12.7 placemirror** With the command `\placemirror` you can place mirrors in the intersections of the grid. In the mandatory argument `<direction>`, you can use the following indicators: H, V

`\laser[color]{TikZ path}` **3.3.1.12.8 laser** The command `\laser` 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{options}` **3.3.1.12.9 laserbeamsetup** With the command `\laserbeamsetup` you can reset the options with global scope.

3.3.1.13 Magic Labyrinth

`\magiclabyrinthcell{column}{row}{number}` **3.3.1.13.1 magiclabyrinthcell** The command `\magiclabyrinthcell` sets a number into grid cell *column* *row*.

`\mline{TikZ path}` **3.3.1.13.2 mlline** The command `\mline` draws a line given by *TikZ path*.

`\magiclabyrinthsetup{options}` **3.3.1.13.3 magiclabyrinthsetup** The `\magiclabyrinthsetup` command resets the options with global scope.

3.3.1.14 Magnets

`\plusH{csv list}` **3.3.1.14.1 plusH** The command `\plusH` 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{csv list}` **3.3.1.14.2 minusH** The command `\minusH` typesets the numbers above the grid indicating how many negative poles are in the respective column.

`\plusV{csv list}` **3.3.1.14.3 plusV** The command `\plusV` typesets the numbers left to the grid indicating how many positive poles are in the respective row.

`\minusV{csv list}` **3.3.1.14.4 minusV** The command `\minusV` typesets the numbers left to the grid indicating how many negative pole ares in the respective row.

`\magnetSH{csv list}` **3.3.1.14.5 magnetSH** The command `\magnetSH` typesets non-magnetic horizontal plates by using the column/row format in *csv list*.

`\magnetsV{⟨csv list⟩}` **3.3.1.14.6 magnetsV** The command `\magnetsV` typesets non-magnetic vertical plates by using the column/row format in `⟨csv list⟩`.

`\PMH{⟨csv list⟩}`, `\MPH{⟨csv list⟩}`,
`\PMV{⟨csv list⟩}`, `\MPV{⟨csv list⟩}` **3.3.1.14.7 Magnetic plates** The command `\PMH` 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{⟨options⟩}` **3.3.1.14.8 magnetssetup** With the command `\magnetssetup` you can reset the options with global scope.

3.3.1.15 Masyu

`\masyucell`
`{⟨column⟩}{⟨row⟩}{⟨element⟩}` **3.3.1.15.1 masyucell** The command `\masyucell` sets an element into grid cell `⟨column⟩` `⟨row⟩`.

`\MasyuW` **3.3.1.15.2 MasyuW** The command `\MasyuW` draws an empty (white) circle.

`\MasyuB` **3.3.1.15.3 MasyuB** The command `\MasyuB` draws a black circle.

`\masyuline{⟨TikZ path⟩}` **3.3.1.15.4 masyuline** The command `\masyuline` draws a line given by `⟨TikZ path⟩`.

`\masyusetup{⟨options⟩}` **3.3.1.15.5 masyusetup** The `\masyusetup` command resets the options with global scope.

3.3.1.16 Minesweeper

`\Mine` **3.3.1.16.1 Mine** The command `\Mine` draws a mine. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

`\minesweeperssetup{⟨options⟩}` **3.3.1.16.2 minesweeperssetup** With the command `\minesweeperssetup` you can reset the options with global scope.

3.3.1.17 Number Link

`\numberlinkcell`
`{⟨column⟩}{⟨row⟩}{⟨element⟩}` **3.3.1.17.1 numberlinkcell** The command `\numberlinkcell` sets a number or letter into grid cell `⟨column⟩` `⟨row⟩`.

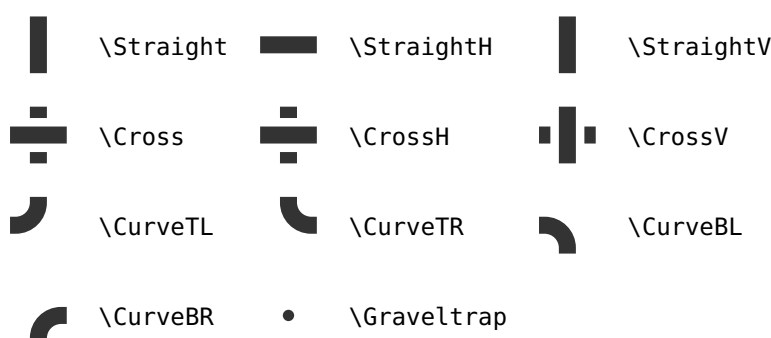
`\link{⟨TikZ path⟩}` **3.3.1.17.2 link** The command `\link` draws a line given by `⟨TikZ path⟩`.

`\numberlinksetup{<options>}` **3.3.1.17.3 numberlinksetup** With the command `\numberlinksetup` you can reset the options with global scope.

3.3.1.18 Resuko

`\resukocell {<column>}{<row>}{<element>}` **3.3.1.18.1 resukocell** The command `\resukocell` sets the *<element>* into grid cell *<column>* *<row>*.

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



`\pitlane{<column>}{<row>}{<direction>}` **3.3.1.18.3 pitlane** The command `\pitlane` draws the pit lane in grid cell *<column>* *<row>* with *<direction>* V or H.

`\parkinglot{<column>}{<row>}` **3.3.1.18.4 parkinglot** The command `\parkinglot` draws the parking lot in grid cell *<column>* *<row>*.

`\trackH{<csv list>}` **3.3.1.18.5 trackH** The command `\trackH` 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{<csv list>}` **3.3.1.18.6 trackV** The command `\trackV` typesets the track tiles left to the grid.

`\track{<TikZ path>}` **3.3.1.18.7 track** The command `\track` 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{⟨options⟩}` **3.3.1.18.8 resukosetup** With the command `\resukosetup` you can reset the options with global scope.

3.3.1.19 Schatzsuche

`\Diamond` **3.3.1.19.1 Diamond** The command `\Diamond` draws a diamond. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

`\schatzsuchesetup{⟨options⟩}` **3.3.1.19.2 schatzsuchesetup** With the command `\schatzsuchesetup` you can reset the options with global scope.

3.3.1.20 Skyline

`\skylineT{⟨csv list⟩}` **3.3.1.20.1 skylineT** The command `\skylineT` typesets the numbers above the grid indicating how many skycrapers are visible. It expects a comma-separated list as an argument.

`\skylineB{⟨csv list⟩}` **3.3.1.20.2 skylineB** The command `\skylineB` typesets the numbers below the grid.

`\skylineL{⟨csv list⟩}` **3.3.1.20.3 skylineL** The command `\skylineL` typesets the numbers left to the grid.

`\skylineR{⟨csv list⟩}` **3.3.1.20.4 skylineR** The command `\skylineR` typesets the numbers right to the grid.

`\skylinecell{⟨column⟩}{⟨row⟩}{⟨height⟩}` **3.3.1.20.5 skylinecell** The command `\skylinecell` sets the *⟨height⟩* of the grid cell *⟨column⟩⟨row⟩*.

`\skylinesetup{⟨options⟩}` **3.3.1.20.6 skylinsetup** With the command `\skylinsetup` you can reset the options with global scope.

3.3.1.21 Slitherlink

`\slitherlinkcell{⟨column⟩}{⟨row⟩}{⟨number⟩}` **3.3.1.21.1 slitherlinkcell** The command `\slitherlinkcell` sets the *⟨number⟩* of the grid cell *⟨column⟩⟨row⟩*.

`\slitherlinksetup{⟨options⟩}` **3.3.1.21.2 slitherlinksetup** With the command `\slitherlinksetup` you can reset the options with global scope.

3.3.1.22 Star Battle

`\starbattlecell{⟨column⟩}{⟨row⟩}`
`{⟨element⟩}` **3.3.1.22.1 starbattlecell** With the `\starbattlecell` command, you can set an `⟨element⟩` in the grid cell `⟨column⟩⟨row⟩`, e.g. the `\Star` command.

`\starbattlesetup{⟨options⟩}` **3.3.1.22.2 starbattlesetup** With the command `\starbattlesetup` you can reset the options with global scope.




3.3.1.23 Stars and Arrows

`\starsH{⟨csv list⟩}` **3.3.1.23.1 starsH** The command `\starsH` 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{⟨csv list⟩}` **3.3.1.23.2 starsV** The command `\starsV` typesets the numbers left to the grid.

`\Star` **3.3.1.23.3 Star** The command `\Star` draws a star. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

3.3.1.23.4 Arrows You can use the following commands to draw different arrows:

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

`\starsandarrowssetup{⟨options⟩}` **3.3.1.23.5 starsandarrowssetup** The command `\starsandarrowssetup` resets the options with global scope.

3.3.1.24 Sudoku

`\lpsudokucell{⟨column⟩}{⟨row⟩}`
`{⟨number⟩}` **3.3.1.24.1 lpsudokucell** The command `\lpsudokucell` sets the `⟨number⟩` of the grid cell `⟨column⟩⟨row⟩`.

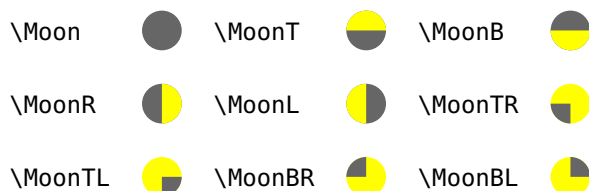
`\lpsudokusetup{⟨options⟩}` **3.3.1.24.2 lpsudokusetup** With the command `\lpsudokusetup` you can reset the options with global scope.

3.3.1.25 Sun and Moon

`\Star` **3.3.1.25.1 Star** The command `\Star` draws a star. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

`\Cloud` **3.3.1.25.2 Cloud** The command `\Cloud` draws a dark cloud. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

3.3.1.25.3 Howl at the Moon You can use the following commands to draw different illuminated moons:



`\sunandmoonssetup{<options>}` **3.3.1.25.4 sunandmoonssetup** With the command `\sunandmoonssetup` you can reset the options with global scope.

3.3.1.26 Tents and Trees

`\tentH{<csv list>}` **3.3.1.26.1 tentH** The command `\tentH` typesets the numbers above the grid indicating how many tents are in the respective column. It expects a comma-separated list as an argument.

`\tentV{<csv list>}` **3.3.1.26.2 tentV** The command `\tentV` typesets the numbers left to the grid.

`\Tree` **3.3.1.26.3 Tree** The command `\Tree` draws a tree. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]! The design of the tree is based on [Alain Matthes](#)' answer to this [question](#) on T_EX.sx.

`\Tent` **3.3.1.26.4 Tent** The command `\Tent` draws a tent. It can be used in commands like `\setcell` [see: 3.3.2.1.1] or `\setrow` [see: 3.3.2.1.4]!

`\tentsandtreessetup{<options>}` **3.3.1.26.5 tentsandtreessetup** With the command `\tentsandtreessetup` you can reset the options with global scope.

3.3.1.27 Tunnel

`\tunnelH{<csv list>}` **3.3.1.27.1 tunnelH** The command `\tunnelH` 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.

`\tunnelV{<csv list>}` **3.3.1.27.2 tunnelV** The command `\tunnelV` typesets the numbers left to the grid.

`\portal{<column>}{<row>}` **3.3.1.27.3 portal** The command `\portal` is used for the placement of tunnel portals in the grid.

`\tube{<TikZ path>}` **3.3.1.27.4 tube** The command `\tube` draws the tunnel tube given by `<TikZ path>`. 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_EX.sx.

```
\tube{(1.5,2.5)--(3.5,2.5)--(3.5,4.5)}
```

`\tunnelsetup{<options>}` **3.3.1.27.5 tunnelsetup** With the command `\tunnelsetup` you can reset the options with global scope.

3.3.2 User commands

3.3.2.1 In the grid

`\setcell{<column>}{<row>}{<element>}` **3.3.2.1.1 setcell** With the `\setcell` command, you can set `<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{<csv list>}{<element>}` **3.3.2.1.2 setcells** With the `\setcells` command, you can set `<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[<fontsize>]{<column>}{<row>}{<element>}` **3.3.2.1.3 setbigcell** The `\setbigcell` command sets `<element>` into a big (2×2) cell `<column><row>` as central node. The optional argument `<fontsize>` is set to 'Huge' by default.

`\setrow{<row>}{<csv list>}` **3.3.2.1.4 setrow** With the `\setrow` command, you can set the contents of a `<row>`. These may be numbers or letters.

`\setcolorrow{<row>}{<csv list>}` **3.3.2.1.5 setcolorrow** With the `\setcolorrow` command, you can set the contents of a `<row>`. Furthermore, the background of the cell is filled with color LP@c@romannumber [see: [3.3.3.3.6](#)]. With the number 0, you can black out the grid cell.

`\setcolumn{<column>}{<csv list>}` **3.3.2.1.6 setcolumn** With the `\setcolumn` command, you can set the contents of a `<column>`. These may be numbers or letters.

`\setcolorcolumn`
`{\langle column \rangle}{\langle csv list \rangle}` **3.3.2.1.7 setcolorcolumn** With the `\setcolorcolumn` command, you can set the contents of a `\langle column \rangle`. Furthermore, the background of the cell is filled with color LP@c@romannumber [see: 3.3.3.3.6].

`\setrule{\langle column \rangle}{\langle row \rangle}`
`{\langle rule \rangle}` **3.3.2.1.8 setrule** With the `\setrule` command, you can set a calculation rule `\langle rule \rangle` into the top left corner of cell `\langle column \rangle \langle row \rangle`. The rule is typeset in inline math mode. You might consider using the `\times` and `\div` commands.

`\fillcell{\langle column \rangle}{\langle row \rangle}` **3.3.2.1.9 fillcell** With the `\fillcell` command, you can fill cell `\langle column \rangle \langle row \rangle` with the color defined with environment option `color`². It is aware of the current values of the surrounding environment options `rows`, `columns`, `scale` and `color`. Furthermore, a check if the cell is within the grid is applied.

`\fillrow{\langle row \rangle}{\langle csv list \rangle}` **3.3.2.1.10 fillrow** With the `\fillrow` command, you can fill a `\langle row \rangle`. In `\langle csv list \rangle` '1' means 'fill' and '0' means 'don't fill'. Internally, `\fillrow` uses `\fillcell` [see: 3.3.2.1.9].

`\fillcolumn{\langle column \rangle}{\langle csv list \rangle}` **3.3.2.1.11 fillcolumn** With the `\fillcolumn` command, you can fill a `\langle column \rangle`. In `\langle csv list \rangle` '1' means 'fill' and '0' means 'don't fill'. Internally, `\fillcolumn` uses `\fillcell` [see: 3.3.2.1.9].

`\filldiagonals[\langle color \rangle]` **3.3.2.1.12 filldiagonals** With the `\filldiagonals` command, you can fill the diagonals with the color specified with the optional argument `\langle color \rangle` (default: yellow!20). Furthermore, it checks for a quadratic grid, otherwise an error message is issued.

`\framearea{\langle color \rangle}{\langle TikZ path \rangle}` **3.3.2.1.13 framearea** The command `\framearea` frames the area given by `\langle TikZ path \rangle` with color `\langle color \rangle`. The reference for coordinates is the bottom left corner of the cell.

```
\framearea{green}{(2,2)--(2,3)--(3,3)--(3,2)--(2,2)}
```

This command will color the frame of the grid cell (2,2) green. You should consider using this command in the `puzzleforeground` [see: 3.2.2.2] environment.

`\fillarea{\langle color \rangle}{\langle TikZ path \rangle}` **3.3.2.1.14 fillarea** The command `\fillarea` fills the area given by `\langle TikZ path \rangle` with color `\langle color \rangle`. The reference for coordinates is the bottom left corner of the cell. You should consider using this command in the `puzzlebackground` [see: 3.2.2.1] environment.

²Therefore, you must define an option `color` in the style file you want to use fill commands

`\colorarea{<color>}{<TikZ path>}` **3.3.2.1.15 colorarea** The command `\colorarea` fills the area given by `<TikZ path>` with color `<color>` – just like `\framearea` without frame.

`\framepuzzle[<color>]` **3.3.2.1.16 framepuzzle** With the `\framepuzzle` command, you can frame the grid (thicker line) with the color specified with the optional argument `<color>` (default: black).

`\tikzpath{<column>}{<row>}`
`{<csv list>}` **3.3.2.1.17 tikzpath** With the `\tikzpath` command, you can easily construct a TikZ path. You just need to define a starting point `<column><row>` (bottom left corner) and a `<csv list>` 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{<column>}{<row>}`
`{<csv list>}` **3.3.2.1.18 xtikzpath** The `\xtikzpath` command is an evolution of the `\tikzpath` command with slightly different input syntax. In the `<csv list>` 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.

3.3.2.2 Presentation

`\titleformat{<format>}` **3.3.2.2.1 titleformat** With the `\titleformat` command, you can define the `<format>` of the title. By default, the definition is as follows:

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

`\puzzlecounter` **3.3.2.2.2 puzzlecounter** The command `\puzzlecounter` provides the general puzzle counter in textual form to use it in `\definecounterstyle`.

`\setpuzzlecounter{<number>}` **3.3.2.2.3 setpuzzlecounter** With the command `\setpuzzlecounter`, you can reset the puzzle counter, for example before the solutions.

`\definecounterstyle{<name>}`
`{<definition>}` **3.3.2.2.4 definecounterstyle** The command `\definecounterstyle` allows you to define your own styles. For example, the style `left` is defined as follows:

```
\definecounterstyle{left}{
  \begingroup\reversemarginpar\marginnote{
    \tikz\node[shape=rectangle,fill=yellow!40,inner sep=7pt,
      draw,rounded corners=3pt,thick]
    {\Huge\puzzlecounter};}\LP@cvmoffset\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 `cvoffset`.

`\setgridlinestyle{<style>}` **3.3.2.2.5 setgridlinestyle** The command `\setgridlinestyle` sets the style of lines used in the grid. By default, the style is set to solid, whereas `slitherlink.sty` uses dashed.

`\setnormallinewidth{<dimension>}` **3.3.2.2.6 setnormallinewidth** With the command `\setnormallinewidth`, you can set the width of the standard lines (default: 0.5pt)

`\setthicklinewidth{<dimension>}` **3.3.2.2.7 setthicklinewidth** With the command `\setthicklinewidth`, you can set the width of the ‘thicker’ lines (default: 1.5pt)

3.3.3 Internal commands

3.3.3.1 Initialization

`\LP@define@key{<prefix>}`
`{<package>}{<option>}{<default>}` **3.3.3.1.1 LP@define@key** With the `\LP@define@key` command, you can define the options of the environment `<package>`. A `<prefix>` is needed for creating different name spaces.

```
\LP@define@key{LP@BS}{battleship}{rows}{5}
```

This code snippet defines the option `rows` for environment `battleship` with the default value 5. This value is stored in `\LP@BS@rows`.

`\LP@define@choicekey@fontsize`
`{<prefix>}{<package>}{<default>}` **3.3.3.1.2 LP@define@choicekey@fontsize** With this command, you can define the choice key option `fontsize` of the environment `<package>`. Possible keys are: `tiny`, `scriptsize`, `footnotesize`, `small`, `normalsize`, `large`, `Large`, `LARGE`, `huge`, `Huge`

3.3.3.2 Drawing grids

`\LP@drawgrid{<xmin>}{<ymin>}{<xmax>}{<ymax>}{<step>}` **3.3.3.2.1 LP@drawgrid** With the `\LP@drawgrid` command, you can draw the grid $(\langle xmin \rangle, \langle ymin \rangle)$ to $(\langle xmax \rangle, \langle ymax \rangle)$ with step $\langle step \rangle$. For drawing the standard puzzle grid the step must be 1cm.

`\LP@drawsudokugrid` **3.3.3.2.2 LP@drawsudokugrid** The command `\LP@drawsudokugrid` draws the standard Sudoku grid, but just the thicker lines. You will have to overlay the standard grid to get a full Sudoku grid.

`\LP@drawbackground{<xmin>}{<ymin>}{<xmax>}{<ymax>}{<color>}` **3.3.3.2.3 LP@drawbackground** The command `\LP@drawbackground` draws the background color of the grid.

3.3.3.3 In the grid

`\LP@LP@setcellcontent{<column>}{<row>}{<element>}` **3.3.3.3.1 LP@setcellcontent** The command `\LP@setcellcontent` is the generic command to set an arbitrary $\langle element \rangle$.

`\LP@LP@setcellcontentC{<column>}{<row>}{<element>}` **3.3.3.3.2 LP@setcellcontentC** The command `\LP@setcellcontentC` is the generic command to set an arbitrary $\langle element \rangle$ in a centered node in the bottom left corner.

`\LP@setrowcontents{<csv list>}{<column>}{<row>}` **3.3.3.3.3 LP@setrowcontents** The command `\LP@setrowcontents` is the generic command to set row contents. It does not necessarily start with $\langle column \rangle$ 1!

`\LP@setcolumncontents{<csv list>}{<column>}{<row>}` **3.3.3.3.4 LP@setcolumncontents** The command `\LP@setcolumncontents` is the generic command to set column contents. It does not necessarily start with $\langle row \rangle$ 1!

`\LP@ingrid{<column>}{<row>}{<max column>}{<max row>}{<package>}` **3.3.3.3.5 LP@ingrid** With the `\LP@ingrid` command, you can check if an element – that should be placed – is within the grid. Otherwise an error message is issued.

`\LP@definecolor{<name>}{<rgb color>}` **3.3.3.3.6 LP@definecolor** With the `\LP@definecolor` command, you can define named rgb colors, especially for defining background colors of numbers used in `\setcolorrow` [see: 3.3.2.1.5] and `\setcolorcolumn` [see: 3.3.2.1.7].

The background color names follow the pattern: `LP@c@romannumber`

```
\LP@definecolor{LP@c@iv}{.55,1,.88}
```

This command will define the new background color of number 4!

3.3.3.4 Around the grid

`\LP@leftcolumn{{\langle csv list \rangle}}` **3.3.3.4.1 LP@leftcolumn** With the `\LP@leftcolumn` command, you can set the contents of the column left to the grid. The `\skylineL` command for the skyline environment is for example defined as follows:

```
\let\skylineL\LP@leftcolumn
```

`\LP@rightcolumn{{\langle csv list \rangle}}` **3.3.3.4.2 LP@rightcolumn** With the `\LP@rightcolumn` command, you can set the contents of the column right to the grid.

`\LP@toprow{{\langle csv list \rangle}}` **3.3.3.4.3 LP@toprow** With the `\LP@toprow` command, you can set the contents of the row above the grid.

`\LP@bottomrow{{\langle csv list \rangle}}` **3.3.3.4.4 LP@bottomrow** With the `\LP@bottomrow` command, you can set the contents of the row below the grid.

3.3.3.5 Presentation

`\LP@drawcounter{\langle name \rangle}` **3.3.3.5.1 LP@drawcounter** The command `\LP@drawcounter` draws the puzzle counter with counter style `\langle name \rangle`.



4 Examples

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

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] Stephen Ostermiller. QQwing – Sudoku Generator and Solver, 2011.
(<http://ostermiller.org/qqwing/>).

B

battleship environment 74
 \battleshipsetup 61
 bgcolor environment option . 10,
 12, 13, 15, 17, 18, 20, 22,
 23, 26, 28, 30, 31, 33–35,
 37, 39, 40, 45, 46, 48, 49,
 51, 53, 55, 56, 60
 \Black 63
 bokkusu environment 57, 58
 bokkusu.sty 57
 \bokkususetup 57, 62
 \bridge 62
 \bridgescolumn 62
 \bridgesrow 62
 \bridgessetup 62

C

\chaossudokucell 62
 \chaossudokusetup 62
 \classicgame 61
 \Cloud 70
 <color> mandatory argument . 72,
 73, 75
 <color> optional argument 62, 65,
 72, 73
 color environment option 13, 15,
 18, 23, 34, 37, 39, 60, 72
 \colorarea 73
 <column> mandatory argument 60–
 69, 71–73, 75
 columns environment option . 10,
 11, 13, 15, 16, 18, 20, 21,
 23, 25, 27, 29, 31, 32, 34,
 35, 37, 39, 40, 44, 46, 47,
 49, 53, 54, 56, 58, 59, 71,
 72
 counterstyle environment option
 10, 12, 13, 15, 17, 18, 20,
 22, 23, 26, 28, 30, 31, 33,
 34, 36, 37, 39, 41, 45, 46,
 48, 49, 51, 53, 55, 56, 60
 \Cross 67
 \CrossH 67
 \CrossV 67
 <csv list> mandatory argument 61–
 73, 75, 76
 \CurveBL 67

\CurveBR 67
 \CurveTL 67
 \CurveTR 67
 cvoffset environment option 10,
 12, 13, 15, 17, 18, 20, 22,
 23, 26, 28, 30, 31, 33, 34,
 36, 37, 39, 41, 45, 46, 48,
 49, 51, 53, 55, 56, 60, 74

D

\ddsudokucell 60
 \ddsudukusetup 60
 <default> mandatory argument 74
 \definecounterstyle ... 73, 74
 <definition> mandatory argument
 74
 \Diamond 68
 <dimension> mandatory argument
 74
 <direction> mandatory argument
 60, 64, 67
 \div 72
 <double> optional argument . 62
 \Down 69

E

<element> mandatory argument 66,
 67, 69, 71, 75
 environment
 battleship 74
 bokkusu 57, 58
 logicpuzzle 59
 lpsudoku 51
 minipage 58
 puzzlebackground 58, 60, 65,
 72
 puzzleforeground 59, 60, 72
 skyline 76
 environment option
 bgcolor 10, 12, 13, 15, 17, 18,
 20, 22, 23, 26, 28, 30, 31,
 33–35, 37, 39, 40, 45, 46,
 48, 49, 51, 53, 55, 56, 60
 color 13, 15, 18, 23, 34, 37,
 39, 60, 72
 columns 10, 11, 13, 15, 16, 18,
 20, 21, 23, 25, 27, 29, 31,
 32, 34, 35, 37, 39, 40, 44,

- 46, 47, 49, 53, 54, 56, 58,
59, 71, 72
 - counterstyle ... 10, 12, 13,
15, 17, 18, 20, 22, 23, 26,
28, 30, 31, 33, 34, 36, 37,
39, 41, 45, 46, 48, 49, 51,
53, 55, 56, 60
 - cvoffset 10, 12, 13,
15, 17, 18, 20, 22, 23, 26,
28, 30, 31, 33, 34, 36, 37,
39, 41, 45, 46, 48, 49, 51,
53, 55, 56, 60, 74
 - fontsize 10, 11, 13, 17, 18,
20, 22, 23, 26, 28, 29, 31,
32, 34, 35, 37, 39, 40, 44,
46, 48, 49, 51, 53, 55, 56,
59, 71, 74
 - grid 15
 - rows . 10, 11, 13, 14, 16, 18,
20, 21, 23, 25, 27, 29, 31,
32, 34, 35, 37, 39, 40, 44,
46, 47, 49, 53, 54, 56, 58,
59, 71, 72, 74
 - sbindent 11
 - sbshipscale 12
 - sbwidth 12
 - scale 10, 11, 13, 15, 16, 18,
20, 22, 23, 26, 28, 29, 31,
32, 34, 35, 37, 39, 40, 44,
46, 48, 49, 51, 53, 55, 56,
59, 71, 72
 - solution 23
 - sudoku 44
 - title 10, 11, 13, 15, 17, 18,
20, 22, 23, 26, 28, 29, 31,
32, 34, 35, 37, 39, 40, 44,
46, 48, 49, 51, 53, 55, 56,
59
 - titleindent 10, 11,
13, 15, 17, 18, 20, 22, 23,
26, 28, 30–32, 34, 35, 37,
39, 40, 44, 46, 48, 49, 51,
53, 55, 56, 59
 - titlewidth 10, 11,
13, 15, 17, 18, 20, 22, 23,
26, 28, 30, 31, 33–35, 37,
39, 40, 44, 46, 48, 49, 51,
53, 55, 56, 60
 - width 10, 11, 13, 15, 16, 18,
20, 21, 23, 26, 28, 29, 31,
32, 34, 35, 37, 39, 40, 44,
46, 47, 49, 51, 53, 55, 56,
59
- ## F
- \fillarea 60, 72
 - \fillcell 72
 - \fillcolumn 72
 - \filldiagonals 72
 - \fillrow 72
 - <fontsize>* optional argument .. 71
 - fontsize environment option 10,
11, 13, 17, 18, 20, 22, 23,
26, 28, 29, 31, 32, 34, 35,
37, 39, 40, 44, 46, 48, 49,
51, 53, 55, 56, 59, 71, 74
 - <format>* mandatory argument 73
 - \fourwindscell 63
 - \fourwindssetup 63
 - \framearea 60, 72, 73
 - \framepuzzle 73
- ## G
- \Graveltrap 67
 - grid environment option 15
- ## H
- \hakyuucell 63
 - \hakyuusetup 63
 - <height>* mandatory argument 68
 - \hitorisetup 63
- ## K
- \kakurocolumn 63
 - \kakurorow 63
 - \kakurosetup 63
 - \kendokucell 63
 - \kendokusetup 64
 - \killersudokucell 64
 - \killersudokusetup 64
 - \KKR 63, 71
- ## L
- \laser 65
 - \laserbeamsetup 65
 - \laserH 64

<code>\laserV</code>	64	<code><default></code>	74
<code>\Left</code>	69	<code><definition></code>	74
<code>\LeftDown</code>	69	<code><dimension></code>	74
<code>\LeftUp</code>	69	<code><direction></code>	60, 64, 67
<code><length></code> mandatory argument	60	<code><element></code> .	66, 67, 69, 71, 75
<code>\link</code>	66	<code><format></code>	73
logicpuzzle environment ...	59	<code><height></code>	68
logicpuzzle.sty	57, 58	<code><length></code>	60
<code>\LP</code>	74–76	<code><max column></code>	75
<code>\LP@bottomrow</code>	76	<code><max row></code>	75
<code>\LP@BS@rows</code>	74	<code><name></code>	74–76
<code>\LP@define@key</code>	74	<code><number></code> .	60, 62–65, 68, 69,
<code>\LP@definecolor</code>	75		73
<code>\LP@drawbackground</code>	75	<code><option></code>	74
<code>\LP@drawgrid</code>	75	<code><options></code>	60–71
<code>\LP@drawsudokugrid</code>	75	<code><package></code>	74, 75
<code>\LP@ingrid</code>	75	<code><prefix></code>	74
<code>\LP@leftcolumn</code>	76	<code><rgb color></code>	75
<code>\LP@rightcolumn</code>	76	<code><row></code>	60–69, 71–73, 75
<code>\LP@setcellcontent</code>	75	<code><rule></code>	72
<code>\LP@setcellcontentC</code>	75	<code><ship segment></code>	61
<code>\LP@setcolumncontents</code>	75	<code><step></code>	75
<code>\LP@setrowcontents</code>	75	<code><style></code>	74
<code>\LP@toprow</code>	76	<code><sumH></code>	63
LPbackground PGF layer ..	58–60	<code><sumV></code>	63
LPbackgroundtwo PGF layer .	58,	<code><TikZ path></code> 62, 65–67, 71–73	
	59	<code><width></code>	58
LPbgcolor PGF layer	58, 59	<code><xmax></code>	75
LPdump PGF layer	58, 59	<code><xmin></code>	75
LPforeground PGF layer ..	58–60	<code><ymax></code>	75
LPforegroundtwo PGF layer .	58,	<code><ymin></code>	75
	59	<code>\marginnote</code>	74
lpsudoku environment	51	<code>\MasyuB</code>	66
lpsudoku.sty	51	<code>\masyucell</code>	66
<code>\lpsudokucell</code>	69	<code>\masyuline</code>	66
<code>\lpsudokusetup</code>	69	<code>\masyusetup</code>	66
		<code>\MasyuW</code>	66
M		<code><max column></code> mandatory argument	
<code>\magiclabyrinthcell</code>	65		75
<code>\magiclabyrinthsetup</code>	65	<code><max row></code> mandatory argument	75
<code>\magnetsH</code>	65	<code>\Mine</code>	66
<code>\magnetssetup</code>	66	<code>\minesweeperssetup</code>	66
<code>\magnetsV</code>	66	minipage environment	58
main PGF layer	58–60	<code>\minusH</code>	65
mandatory argument		<code>\minusV</code>	65
<code><color></code>	72, 73, 75	<code>\mirrorH</code>	64
<code><column></code> .	60–69, 71–73, 75	<code>\mirrorV</code>	64
<code><csv list></code>	61–73, 75, 76		

- \mline 65
- \Moon 70
- \MoonB 70
- \MoonBL 70
- \MoonBR 70
- \MoonL 70
- \MoonR 70
- \MoonT 70
- \MoonTL 70
- \MoonTR 70
- \MPH 66
- \MPV 66
- N**
- $\langle name \rangle$ mandatory argument 74–76
- $\langle number \rangle$ mandatory argument 60, 62–65, 68, 69, 73
- \numberlinkcell 66
- \numberlinksetup 67
- O**
- $\langle option \rangle$ mandatory argument . 74
- optional argument
 - $\langle color \rangle$ 62, 65, 72, 73
 - $\langle double \rangle$ 62
 - $\langle fontsize \rangle$ 71
 - $\langle optional arguments \rangle$... 62
 - $\langle options \rangle$ 59
- $\langle optional arguments \rangle$ optional argument 62
- $\langle options \rangle$ mandatory argument 60–71
- $\langle options \rangle$ optional argument . 59
- P**
- $\langle package \rangle$ mandatory argument 74, 75
- \parkinglot 67
- PGF layer
 - LPbackground 58–60
 - LPbackgroundtwo ... 58, 59
 - LPbgcolor 58, 59
 - LPdump 58, 59
 - LPforeground 58–60
 - LPforegroundtwo ... 58, 59
 - main 58–60
- \pitlane 67
- \placearrow 64
- \placecross 64
- \placeisland 61
- \placemirror 64
- \placesegment 61
- \placeship 60
- \placewater 61
- \plusH 65
- \plusV 65
- \PMH 66
- \PMV 66
- \portal 71
- $\langle prefix \rangle$ mandatory argument . 74
- puzzlebackground environment .
 - 58, 60, 65, 72
- \puzzlecounter 73, 74
- puzzleforeground environment .
 - 59, 60, 72
- R**
- \resukocell 67
- \resukosetup 68
- \reversemarginpar 74
- $\langle rgb color \rangle$ mandatory argument
 - 75
- \Right 69
- \RightDown 69
- \RightUp 69
- $\langle row \rangle$ mandatory argument 60–69, 71–73, 75
- rows environment option . 10, 11, 13, 14, 16, 18, 20, 21, 23, 25, 27, 29, 31, 32, 34, 35, 37, 39, 40, 44, 46, 47, 49, 53, 54, 56, 58, 59, 71, 72, 74
- $\langle rule \rangle$ mandatory argument .. 72
- S**
- sbindent environment option . 11
- sbshipscale environment option
 - 12
- sbwidth environment option . 12
- scale environment option 10, 11, 13, 15, 16, 18, 20, 22, 23, 26, 28, 29, 31, 32, 34, 35, 37, 39, 40, 44, 46, 48, 49, 51, 53, 55, 56, 59, 71, 72

- \schatzsuchesetup 68
- \setbigcell 71
- \setcell 66, 68–71
- \setcells 71
- \setcolorcolumn 72, 75
- \setcolorrow 71, 75
- \setcolumn 71
- \setgridlinestyle 74
- \setnormallinewidth 74
- \setpuzzlecounter 73
- \setrow 66, 68–71
- \setrule 72
- \setthicklinewidth 74
- ⟨*ship segment*⟩ mandatory argument
..... 61
- \shipbox 61
- \shipH 61
- \shipV 61
- skyline environment 76
- \skylineB 68
- \skylinecell 68
- \skylineL 68, 76
- \skylineR 68
- \skylinesetup 68
- \skylineT 68
- slitherlink.sty 74
- \slitherlinkcell 68
- \slitherlinksetup 68
- solution environment option 23
- \Star 69
- \starbattlecell 69
- \starbattlesetup 69
- \starsandarrowssetup 69
- \starsH 69
- \starsV 69
- ⟨*step*⟩ mandatory argument . 75
- \Straight 67
- \StraightH 67
- \StraightV 67
- ⟨*style*⟩ mandatory argument .. 74
- sudoku environment option ... 44
- \sumH 62
- ⟨*sumH*⟩ mandatory argument 63
- \sumV 62
- ⟨*sumV*⟩ mandatory argument 63
- \sunandmoonssetup 70
- T**
- \Tent 70
- \tentH 70
- \tentsandtreessetup 70
- \tentV 70
- ⟨*TikZ path*⟩ mandatory argument
... 62, 65–67, 71–73
- \tikz 74
- tikz.sty 58
- \tikzpath 73
- \times 72
- title environment option 10, 11,
13, 15, 17, 18, 20, 22, 23,
26, 28, 29, 31, 32, 34, 35,
37, 39, 40, 44, 46, 48, 49,
51, 53, 55, 56, 59
- \titleformat 73
- titleindent environment option
10, 11, 13, 15, 17, 18, 20,
22, 23, 26, 28, 30–32, 34,
35, 37, 39, 40, 44, 46, 48,
49, 51, 53, 55, 56, 59
- titlewidth environment option .
... 10, 11, 13, 15, 17, 18,
20, 22, 23, 26, 28, 30, 31,
33–35, 37, 39, 40, 44, 46,
48, 49, 51, 53, 55, 56, 60
- \track 67
- \trackH 67
- \trackV 67
- \Tree 70
- \tube 71
- \tunnelH 70
- \tunnelsetup 71
- \tunnelV 70
- U**
- \Up 69
- V**
- \valueH 61
- \valueV 62
- W**
- ⟨*width*⟩ mandatory argument 58
- width environment option 10, 11,
13, 15, 16, 18, 20, 21, 23,
26, 28, 29, 31, 32, 34, 35,

37, 39, 40, 44, 46, 47, 49,
51, 53, 55, 56, 59

X

$\langle x_{max} \rangle$ mandatory argument . 75
 $\langle x_{min} \rangle$ mandatory argument . 75
 $\backslash x_{tikzpath}$ 67, 73

Y

$\langle y_{max} \rangle$ mandatory argument . 75
 $\langle y_{min} \rangle$ mandatory argument . 75