The maze package

Sicheng Du
siddsc@foxmail.com

May 25, 2026 v1.3

1 Changes in this version

1. Added input validation for the `size` argument: it must be an integer in the range 2–99; the package now raises an error for out-of-range values.
2. Added a storage feature: `\mazesave{name}{size}[seed]` saves the rendered maze in a named box register so that the exact same maze can be reproduced later. Stored mazes also keep a solved overlay, and `\mazeuse[view]{name}` can typeset that overlay on demand.

2 User instructions

The maze package can generate random square mazes, which can be played by navigating from the bottom-left corner to the top-right corner. The user can specify the size of the maze or whether to show its solution.

`\maze` `{`⟨*size*⟩`}[`⟨*seed*⟩`]` is the syntax of the command that generates a maze. Thereinto

`{`⟨*size*⟩`}` controls the density of the walls inside the maze and directly influences its complexity. It must be a positive integer in the range [**2,99**].

`[`⟨*seed*⟩`]` is an optional parameter that specifies the seed for random numbers. If it is omitted, the current time (minute) will be used as the seed instead.

For example, the mazes in Figure 1 can be created by `\maze{30}[4]` and `\maze{25}` respectively.

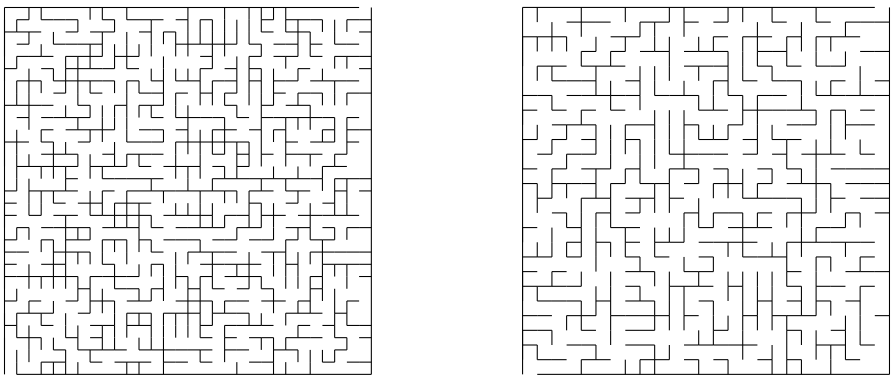

Figure 1: Examples of mazes

`\mazesave` `{`⟨*name*⟩`}{`⟨*size*⟩`}[`⟨*seed*⟩`]` generates a maze once, stores the rendered result and allows that exact maze to be typeset again later.

`{`⟨*name*⟩`}` is the identifier of the stored maze. It is advised to use a string composed of ordinary printable text characters, although special ones like # would probably work.

`{`⟨*size*⟩`}` Same meaning and requirements as in `\maze`.

`[`⟨*seed*⟩`]` Same meaning and requirements as in `\maze`.

`\mazeuse` `[`⟨*view*⟩`]{`⟨*name*⟩`}` typesets a maze that was previously stored with `\mazesave`.

[⟨***view***⟩**]** controls the display of an automatically generated solution to the maze. Put `solution=line` or simply `solution` to generate a line along the solution path, and `solution=cells` to place a circle in each cell along the solution path. If left out, only the maze will be output.

{⟨***name***⟩**}** must match the name set when `\mazesave` is called.

As an example, the mazes in Figure 2 can be created by `\mazesave{demo}{20}` followed by `\mazeuse{demo}`, `\mazeuse[solution]{demo}`, and `\mazeuse[solution=cells]{demo}`.

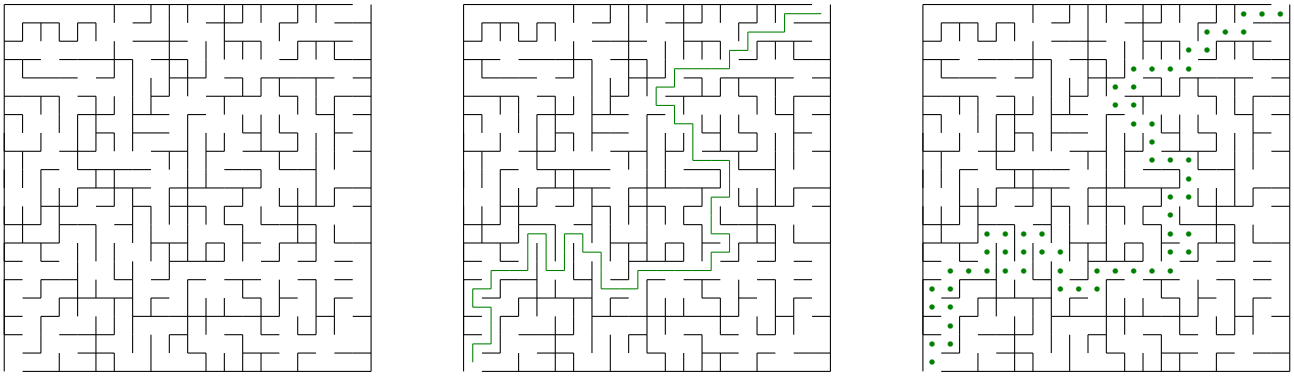

Figure 2: Example of a saved maze and solution display