

mazer_solver_documentation

July 27, 2023

1 MAZE SOLVER DOCUMENTATION

Code is developed to solve the maze with the given conditions.

1.1 Code

The following scripts are available:

- **src modules:** required modules contained in the **src** folder
 - **test:** unit tests applied to the code
 - **main:** main execution script
 - **dockerfile:** to generate docker container with dependencies
-

1.2 Way of Operation

The code will ask us to enter the maze (by file or by console directly) or even that the code itself generate it according to some conditions that the user is asked to do, always complying with the given restrictions.

It is possible to modify the symbols that make up the obstacles or walls of the maze, we will be asked when starting the code.

When the maze is available, we will proceed to solve it.

1.3 Good practices

Code has been developed following good practices and style guides, mainly PEP-8.

Code quality is ensured with a linter, pylint, and is checked every time the code is pushed to Git, using Github Actions. Only code quality higher than 8.5 is allowed

Likewise, when the code is pushed, unit tests are carried out, with pytest, to verify that the code works correctly.

These tests and linter ensure a good quality of the code and a good functioning at a very basic level, being able to develop a complete CI/CD strategy in which on the main (prod) and develop branches, changes are only made if certain standards are met. , very demanding on the branch prod.