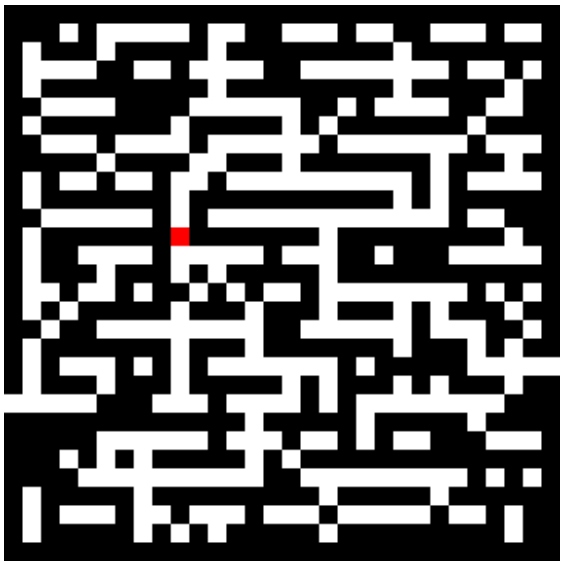
Task: Help the Robot get out of the maze

Objective of the task:

Write a Python program that will allow the robot to find the exit from the maze.



Task content:

The task is to create a function that, for a given maze, attached as a text file (.txt), determines whether there is a possible exit for the robot. The maze is represented by a list of strings, where ' ' (space) denotes a path, '#' is a wall, and 'r' is the robot's starting position. The function should return True if there is a path to the exit, or False if there is no such path. Mazes can have various shapes and do not have to be square or rectangular. The robot can move in four main directions: left, up, right, and down. If there is more than one robot in the maze, the function should return False . The program should also allow the maze to be loaded from an attached text file using a dialog box.

Evaluation criteria:

1. **Algorithm correctness:** Whether the function correctly returns True or False depending on whether the robot can find the exit from the maze for the mazes included in the task and whether it can retrieve it from the text file.
2. **Code Readability and Organization:** Is the code easy to understand, well commented, and organized?
3. **Additional tests:** Checking whether the algorithm passes additional tests that were not included in the task.

Input example:

```
['# ##',
'# r#',
'####']
```

Output example: True

Input example:

```
['####',
'# r#',
'####']
```

Output example: False

Below are some more complicated mazes, along with the correct answers.

```
maze1 = ["###",
         "#r#",
         "###"]
```

The function should return: False

```
maze2 = ["r"]
```

The function should return: True

```
maze3 = ["###",
         "#r",
         "###"]
```

The function should return: True

```
maze4 = ["r ",
         "r"]
```

The function should return: False

```
L abirynt5 = ["########",
              "# #r####",
              "# # # ",
              "# # # ##",
              "# # # ##",
              "# #",
              "########"]
```

The function should return: True

```
Maze6 = ["########",
              "#r# ## #",
              "# # # #",
              "# # # ##",
              "# # # #",
              "# ##",
              "########"]
```

The function should return: False

```
Maze7 = [' # # # ',
              ' ### ',
              '## # ## ',
              ' ##### ',
              ' # ##### ',
              '## ##r # #',
              ' # # # ',
              ' # # # ##',
              ' ### ####',
              ' # #']
```

The function should return: False

```
Maze 8 = ["################################",
            "### # ## ## # ## # #",
          "# ### ##### ######### ########",
          "# ## # ## # # #",
          "# ######### ######## ##### ##",
          "## #### ## ## ## ##",
          "# ####### ##### # ####### ####",
          "## # # # # # #",
          "##### ######### ####### ### ##",
          "# # # # # # #",
          "# ######## ########## # ######",
          "## # # # ###",
          "# ########r################## ##",
          "# ## ## # ## ### #",
          "# ### ### # ##### ############",
          "# # # # # # # # # # #",
          "# # ##### ### ### ### # ### ##",
          "## ## ## ## ## #",
          "# ####### ### ####### ### # ##",
          "# # # # # # # ##### #",
          "# ### ### ##### # ### # ### ##",
          "## ### # # # ###",
          "############## ##### ##### ####",
          "##### # # # # ### #",
          "### # # ##### # ##############",
          "#### ## # # #",
          "# ##### ######### ##### ######",
          "# # # # # # # # #",
          "# ##### # # ##### ######### ##",
          "#####################################"]
```

The function should return: False

```
Maze9 = ["#################################",
         "### # # ## # ## # #",
         "# ### ##### ######### ########",
         "# ## # ## # # #",
         "# ######### ######## ##### ##",
         "## #### ## # ## ##",
         "# ####### ##### # ####### ####",
         "## # # # # #",
         "##### #### ### ####### ### ##",
         "# # # # # # # #",
         "# ######## ########## # ######",
         "## # # # ###",
         "# #######r####### ######### ##",
         "# ## # # ## ### #",
         "# ### ### # ##### ###########",
         "# # # # # # # # # # #",
         "# # ##### ### ### ### # ### ##",
         "## ## ## # ## #",
         "# ####### ### ####### ### # ##",
         "# # # # # # # ##### ",
         "# ### ### ##### # ### # ### ##",
         " ### # # # ###",
         "############## ##### ##### ####",
         "##### # # # # ### #",
         "### # # ##### # #############",
         "#### ## # # #",
         "# ##### ######### ##### ######",
         "# # # # # # # # # #",
         "# ##### # # ##### ######### ##",
         "#####################################"]
```

The function should return: True