

## Homework 2 (IROS)

### Conditions:

- Worth 25% of the final grade.
- Work in groups (up to 3 members recommended) or individually.
- Deadline: Friday 12/07/2024 – 23:45.
- For assistance, you can ask me or your friends.
- Submit your package folder on Moodle, put the name of group members as a comment inside setup.py.

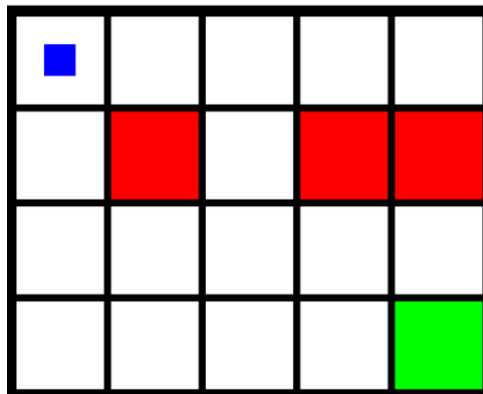
# Grid World

## Objective:

Finish the **grid\_world** package by finishing the following subtasks:

1. Read **World\_Grid.py** carefully to understand what variables you need from this class.
2. Complete all **TODO** tasks inside **world.py** and **setup.py**.
3. Publish world state to topic '**world\_state**' as float array message.
4. Publish score to topic '**score**' as float message.
5. Both of score and state must be published every 1 second.
6. Subscribe to '**usr\_cmd**' topic which send commands as a string message which has one string with the following possibilities:
  - a. 'u' : Player moves up
  - b. 'd' : Player moves down
  - c. 'l' : Player moves left
  - d. 'r' : Player moves right
7. If the command leads to an obstacle block or outside world boundary, player stays in its place.
8. Each time command is received the score decreased by 1.
9. If player reached the terminal block, display the score on the terminal.
10. **Bonus**: reset the game after player reach the terminal block.
11. Add your node to '**setup.py**'.
12. Build and source.
13. Run the node.
14. Display the world using rqt.

The following is an example for the grid world with size 4x5 which is the default size:



**Red** blocks mean obstacles, they always have a random positions.

**Green** block means Terminal block and it is always at the lower right corner.

**Blue** small square means the player position and it always start from the upper left corner.

State matrix for the previous example:

$$state = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$