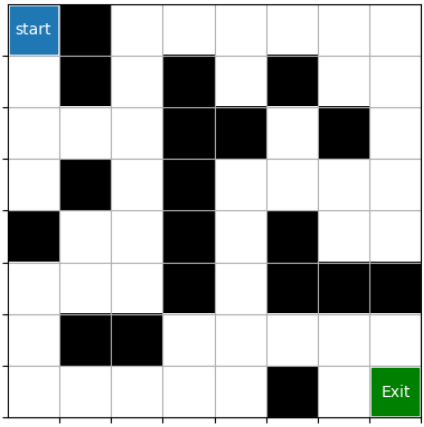
Tutorial 5 - Temporal Difference (TD)

Exercise N°1:

We consider the problem of solving a Maze with same presentation in 1st exercise in previous Lab. The agent begins at the start cell and must find the exit cell by moving through the maze

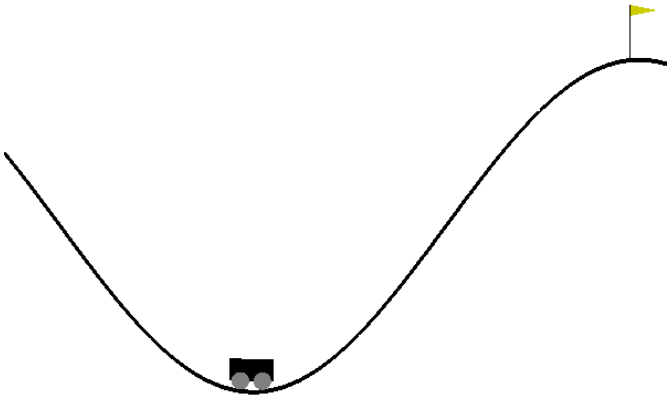
- The maze is represented as a 2D grid with three types of cells: blocked cells, free cells, and exit cell.
- To get to the exit, the agent moves through the maze in a series of steps. The are exactly 4 actions encoded as integers 0-3: [0 - up, 1 - down, 2 - left, 3 - right]
    - The possible actions deterministically cause the agent to move one cell in the respective directions
    - If the agent decides to take a non-permitted action (moving to blocked cells or moving outside the maze), it stays in the same state
- The reward function is defined as follow:
    - The agent receives a reward of (+50) for reaching the exit cell
    - The agent receives a penalty of (-10) for colliding with a wall (i.e. for trying to enter a blocked cell or moving out of the maze)
    - A penalty of (-1) is applied for a move which did not result in finding the exit cell
- The figure below shows an example of a maze environment
    - The start state is located in the (0, 0) cell (blue cell).
    - The exit state is located in the (7, 7) cell (green cell).
    - Black color indicates blocked cells.
    - White color indicates a free cell.
    - A cell is identified by (index_row, index_col) and takes the value of “0” if it is free and “1” if it is blocked.
    - The agent wins if it reaches the exit cell and fails if it exceeds the time limit.



1. Apply the following Temporal-Difference (TD) learning methods:
    a. SARSA
    b. Q-learning
    c. Double Q-learning
2. What is the main difference between MC, DP and TD learning methods?

Exercise N°2 (Additional):

The Mountain Car environment, available in Gymnasium (OpenAI Gym) presents a classic control challenge in reinforcement learning. A car is on a one-dimensional track, positioned between two mountains. The goal is to drive up the mountain on the right to reach the flag



1. Based on Gymnasium documentation:
    a. Formalize the problem as a Markov Decision Process (MDP).
    b. Define initial states, terminal states, and conditions for episode termination.
2. Install the required packages.
3. Apply Q-learning algorithms.
4. Over 200 episodes, evaluate the learned policy based on:
    a. Total reward for each episode.
    b. Number of steps required to reach the goal in each episode.