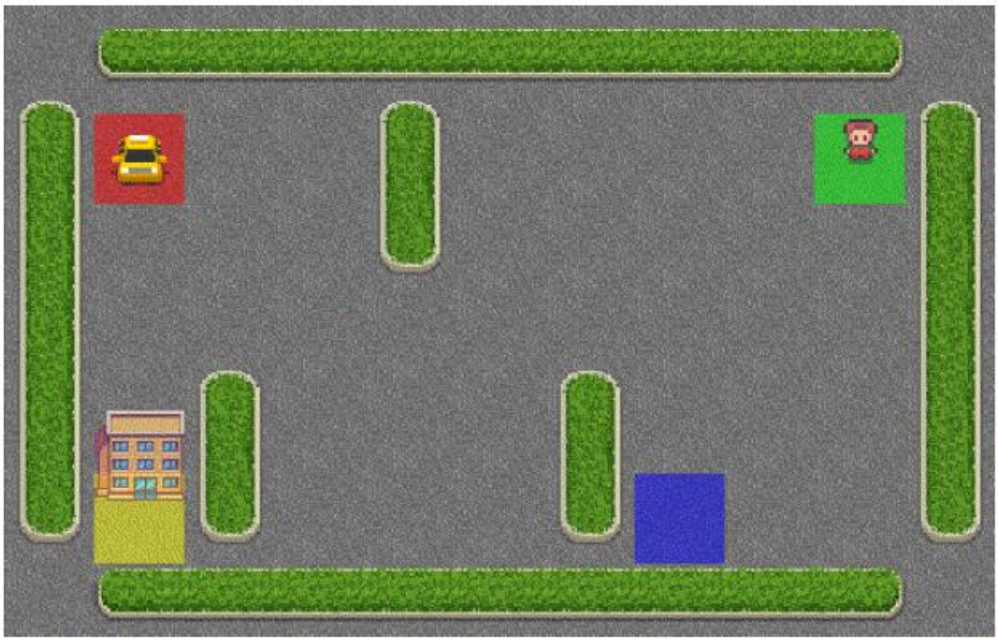
# Homework 4 – RL

The purpose of this exercise is to familiarize with algorithms for solving the MDP problem assuming that the environment is unknown Is. These methods are referred to as Model-Free methods in the literature. In this exercise, two analytical questions and an implementation topic that includes different parts were considered, during which SARSA, Expected SARSA algorithms were used., Q-learning, Tree Backup n-step.

Ali, who you don't know a subject, plans to start an internet taxi that needs a driver and an intelligent agent will take the passenger to their destination instead of the driver. For this purpose, he first wants to measure the possibility of this issue in a hypothetical environment. The city environment is a 5 x 5 table with walls all around and parts of it inside. To learn more about this environment you can use this link.



At any time, the operator can choose one of the four movements down (0), up (1), right (2), left (3), loading the passenger (4) and unloading the passenger (5). which remains in place in boundary conditions in choosing the illegal movement of the agent. On the other hand, depending on the number Your student, the origin of the traveler is one of the colorful houses and his destination is another house. factor per Finding a passenger + 20, will receive -10 for unauthorized boarding or disembarking and otherwise - 1 for lost time.

For this exercise, we are going to get familiar with Gym, which is a functional interface for travel and a set of different environments. In this link, a simple explanation of how to use its environment is given. Also, the code of this environment is given in the attached file, note that you make the environment equal to the last three digits of your student number. For seed, the environment must be reset in time .For example, if your student number is 81011116, the code will be as follows.

```
import gym
env = gym.make('Taxi-v3')
env.seed(seed=123)
Initial_state = env.reset()
```

Each state in this environment is represented by a number. I can use the following command to find the mode. For example, if we are in mode 189, the mode information can be found as follows.

```
taxi_row, taxi_col, pass_idx, dest_idx = env.decode(189)
```

## Questions

1. The q-learning algorithm reduces the value once to 0.1 score and again to value. Compare the implementation and the obtained results in terms of regret (convergence speed and converged value). State your chosen method to reduce the epsilon value during the process.

2. Prepared from modes in the introduced environment can be categorized. After describing these modes, it provides an algorithmic method to get the number of this mode and try that solution using the solution of the previous question.

3. Implement Sarsa and Tree Backup n-Step algorithms for three values of n and compare the obtained results in terms of regret (convergence speed and converged value).

4. Answer the following question according to your student number.

   **If the last digit of your student number is even:**

   A. Solve the problem by using the on-Policy MC method and do the requested items once for 4 and compare the results obtained in terms of / epsilon of reduction and also for epsilon 1 of the amount of regret (convergence speed and converged value) do.

   **If the last digit of your student number is odd:**

   B. Using the off-Policy MC method, answer the requirements of the problem and do the requirements and compare the obtained/obtained results once for the decreasing epsilon and also for the epsilon 0.1 in terms of the amount of regret (convergence speed and converged value) with each other (compare.) Note: Consider the behavioral policy as an epsilon-greedy policy and update it based on the latest Q-values at each step.

5. Does the speed of the last question make a noticeable difference compared to the previous questions? If the answer is positive, explain this issue.