This week you will implement the fundamental algorithms of policy and value iteration. You'll see how your agent's behaviour changes over time and hopefully have your first successful training runs.

1. **Policy Iteration for the MarsRover**
   In the *env.py* file you'll find the first environment we'll work with: the MarsRover. You have seen it as an example in the lecture: the agent can move left or right with each step and should ideally move to the rightmost state. In this first exercise, the environment will be deterministic, that means the rover will always execute the given action. Your task is to complete the given code stub in *policy_iteration.py* with the algorithm from the lecture.

2. **Value Iteration for the probibalistic MarsRover**
   For this second exercise, we modify the MarsRover environment, now the rover may or may not execute the requested action, the probability is 50%. You will complete the code in *value_iteration.py* in order to train an agent on this variation of our environment.

3. **Exploration & Observations from both algorithms**
   Now we ask you to experiment with different environment setups in the file *observations.py*. If you need a reference on how to create an environment, feel free to look at the code for the previous exercises. Please document your experiences and observations in a file called *observations.txt*. Use one line for each answer. The questions for you to answer are:

   - How many iterations does value iteration run for if the transition probabilities are 1 for all state-action pairs?
   - What part of the environment can you change (except for the transition probabilities!) to change this?
   - In the mean, which method takes more iterations for transition probabilities of 1?
   - Which method takes more iterations for transition probabilities of 0.5?
   - Does changing γ change the resulting policy or value function?