Reinforcement Learning Exercise

1. With reference to the animation at `http://www.cim.mcgill.ca/~jer/courses/ai/applets/RL`, imagine a random walker who starts at (1,1) and chooses a direction (N, S, E, W) from any of the available moves (i.e., that do not bump into a wall or the inaccessible square) with equal probability. The outcome of each action is deterministic.

   Every time the walker reaches a flag position (terminal state), he receives the associated reward, and is teleported back to (1,1).

   Using TD-learning with $\alpha = 0.1$ (learning rate) and $\gamma = 1$ (no decay of rewards), determine the utility of each state for this random policy, π.

   Recall that the TD-learning update rule is:
   $U^\pi(s) \leftarrow U^\pi(s) + \alpha(R(s) + \gamma U^\pi(s') - U^\pi(s))$

2. Now, assume that the walker is greedy, and follows a policy that attempts to choose actions that maximize the utility of the next state, that is, he chooses the action $\arg\max_a \sum_{i \in s'} P(s, a, s')U(s')$ with probability $1 - \epsilon$ but a random action with probability ϵ.

   If all possible actions result in equal utility of the next state, which is likely to be the case early on if all states are initialized with a utility estimate of zero, then the agent will pick an action randomly.

   Assume a value of $\epsilon = 0.1$, what are the resulting utlity estimates for each state? How does the agent's path change over time?

3. Without modifying your implementation, what changes would be necessary if the outcome of each action was *not* deterministic, i.e., for any action, the probability of "success" is only 0.8?

4. In the previous steps, we explicitly estimated the utility of each state to guide the agent's choice of action. At this point, modify your code to use Q-Learning instead, for which the learning rule is:
   $Q(s, a) \leftarrow Q(s, a) + \alpha(R(s) + \gamma max_{a'} Q(s', a') - Q(s, a))$
   Implement the same "greedy walker" behaviour as in step 2.

5. Now, modify the walker behaviour so that it is encouraged to explore previously unvisited states in the hope of finding a more efficient path to the postive reward. How does the behaviour change over time?

6. For extra bonus, provide an updated interactive animation (applet) that **correctly** implements both direct utility estimation and TD learning for the associated 4×3 world.