# Answers to Exercises in the Sutton and Barto book (link to book below):

# [http://incompleteideas.net/book/the-book.html](http://incompleteideas.net/b ook/the-book.html)

You’re welcome <https://github.com/LyWangPX/Reinforcement-Learning-2nd-Edition-by-Sutton-Exercise-Solutions>

### 2.1

The greedy action will be performed with probability 1 – 0.5 + 0.5 / 2 = 0.5 + 0.25 = 0.75

### 2.2

The table below shows the rewards received for the actions. the e-greedy action occurred at t = 2, 4, it is possible that it could have also occurred at t = 3 too.

|  |  |  |  |
| --- | --- | --- | --- |
| A1 | A2 | A3 | A4 |
| -1 | 1, -2, 2 | 0 |  |

### 2.3

best epsilon is 0.1... blah blah blah

(Rest of section not super relevant for 2020 syllabus)

### 3.1

example 1: Climbing stairs, states = each step, actions = up/down, rewards = reaching top,

example 2: Social media advertising, states = users friends, posts, what they are currently viewing on social media wall, actions = present advert to user at given time, rewards = user clicks / dwells on advert.

example 3: Simple basketball game, states = distance from hoop, angle to hoop, actions = shot angle, shot power / velocity, rewards = getting ball in hoop

### 3.2

MDP framework is not adequate to represent *all* goal-directed learning tasks, for example, (a situation where the agent doesn’t have to do anything?).

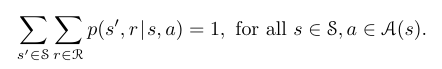
### 3.3

This is a somewhat existential question. I would say that actions are defined and can be much more easily measured in terms of “doing” things, thus the first example given of doing the action of pressing the brake, accelerator and moving the steering wheel is a good place to draw the line. Otherwise the actions and their affects are difficult to measure. However, it ultimately comes down to free choice and it at the discretion of the reinforcement learner person.

### 3.4

Quite complex and cba to make the table.

### 3.5



instead of using for all s’ ∈ S we use for all s’ ∈ T where T are the states visited in the episode.

### 3.6

The return would simply be –yk-1 where k is the number of steps before failure.

This differs from the continuing formulation as the return will be > 0 whereas in this case it is always negative. Additionally, the reward in this case is much more sparse, so over time the agent has less of an idea of whether it is performing the correct action or not.

### 3.7

Here we see a similar issue to the last comment in the above question. The reward is too sparse and hence the robot is unsure what the correct action to take is until it gets close to the end of the maze. This issue could be fixed by any of the following:

* Increasing the discount factor.
* Increasing the reward at the end.
* Adding more reward values throughout the maze.

### 3.8

Assuming GT is the return for time T

G5 = 0 (since there is no R6)

G4 = R5 = 2

G3 = R4 + ½ R5 = 3 + 1 = 4

G2 = R3 + ½ R4 + ½2 R5 = 6 + 1 ½ + ½ = 8

G1 = R2 + ½ R3 + ½2 R4 + ½3 R5 = 2 + 3 + ¾ + ¼ = 6

G0 = R1 + ½ R2 + ½2 R3 + ½3 R4 + ½4 R5 = -1 + 1 + 1 ½ + 3/8 + 1/8 = 2

### 3.9

G0 would be 2 + sum to infinity of 7 x 0.9k

G1 would be G0 - 2

### 3.10

cba to do the rest... (geometric series)