

Exercise 01.01 (part of 02.03 AIMA3). For each of the following assertions, say whether it is *true* or *false* and support your answer with examples or counterexamples where appropriate.

- a) An agent that senses only partial information about the state cannot be perfectly rational [*Hint*: Use the working definition of rationality.];
- b) There exist task environments in which no pure reflex agent can behave rationally [*Hint*: Think about the limitations of reflex agents.];
- c) There exists a task environment in which every agent is rational [*Hint*: Think about the usual properties of the task environment.];
- d) The input to an agent program is the same as the input to the agent function;
- e) Every agent function is implementable by some program/machine combination;
- f) Suppose an agent selects its action uniformly at random from the set of possible actions. There exists a deterministic task environment in which this agent is rational [*Hint*: Check question c).];
- g) It is possible for a given agent to be perfectly rational in two distinct task environments;
- h) Every agent is rational in an unobservable environment [*Hint*: Think about the function of the model.];
- i) A perfectly rational poker-playing agent never loses.

Exercise 01.02 (part of 02.04 AIMA3). For each of the following activities, give a PEAS description of the task environment and characterise it in terms of the usual properties [i.e., i) Fully observable v partially observable; ii) Single agent v multi-agent; iii) deterministic v stochastic; iv) Episodic v sequential; v) Static v dynamic; vi) Discrete v continuous; and, vii) Known v unknown.]

- a) Playing soccer;
- b) Shopping for used AI books on the Internet;
- c) Playing a tennis match;
- d) Practicing tennis against a wall;
- e) Performing a high jump;
- f) Bidding on an item at an auction.

Exercise 01.03 (02.05 AIMA3). Define in your own words the following terms:

- a) agent;
- b) agent function;
- c) agent program;

- d) rationality;
- e) autonomy;
- f) reflex agent
- g) model-based agent;
- h) goal-based agent;
- i) utility-based agent;
- j) learning agent.

Exercise 01.04 (part of 02.06 AIMA3). This exercise explores the differences between agent functions and agent programs.

- a) Can there be more than one agent program that implements a given agent function? Give an example, or show why one is not possible.
- b) Given a fixed machine architecture, does each agent program implement exactly one agent function?
- c) Suppose we keep the agent program fixed but speed up the machine by a factor of two. Does that change the agent function? [*Hint*: Consider whether the environment is static or dynamic.]

Exercise 01.05. Recently, an earthquake, measuring 6.2 ± 0.016 hit Central Italy on 24 August 2016. at 03:36:32 CEST (01:36 UTC). Early reports indicated severe damage in the town of Amatrice. You, as a trainee, was called up to develop an intelligent agent able to remove people from a region inaccessible to humans in Amatrice. This rescuer agent runs on a drone and is described as follows:

- **Percepts:** Each rescue agent gets a three-element percept vector on each turn. The first element, the proximity sensor, should be 1 if the droid is in front of a wall or a rock (in the direction that the droid has). The second comes from the presence sensor under the machine, which emits 1 if there is human there and 0 otherwise. The third comes from an infrared sensor, which emits 1 when the agent have to take off (and go back to the paramedics), and 0 otherwise.
- **Actions:** There are five actions available: Go forward, turn right by 90° , turn left by 90° , grab a human and take off.
- **Goals:** The goal for each agent is to rescue people and go home. To be precise, the performance measure will be 100 points for each person rescued, minus 1 point for each action taken.
- **Environment:** The environment consists of a grid of squares. Some squares contain obstacles (walls around and some rock structures) and other squares are open space. Some of the open squares contain humans. Make sure that each square only contains or a human or a rock or is empty. Each “go forward” action moves one square unless there is an obstacle (wall or rock) in that square, in which case the agent, have to turn. A “grab human” action always rescue a human. A “take off” command ends the simulation.

Design and implement of a Model Based Agent for the aforementioned environment and measure its performance. Remember that the Model Based Agent also can 'remember' things that made in the past and note that there'll many Models that can achieve the problem. The environment is a 15×20 rectangular room, where each square has a 10% chance of containing human (30 humans), 33% chance of containing a rock (100 rocks). Also remember to represent the walls around the rectangular room. Explain why it is impossible to have a Reflex Agent that when receives the take off signal returns to the entrance and just takes off. Speculate on what the best possible reflex agent could do. What prevents a Reflex agent from doing very well?

Exercise 01.06. Made several Agents and Environments defined int the Exercise 01.05 and Measure their performance. How close do they come to the ideal agent for this earthquake environment?

You can implement the code in Python or Java, using the code that is provided in the site
GitHub Lista 1