CHAPTER 2

INTELLIGENT AGENTS

---

function TABLE-DRIVEN-AGENT(*percept*) **returns** an action
   persistent: *percepts*, a sequence, initially empty
           table, a table of actions, indexed by percept sequences, initially fully specified

   append *percept* to the end of *percepts*
   action ← LOOKUP(*percepts*, *table*)
   return *action*

Figure 2.7 The TABLE-DRIVEN-AGENT program is invoked for each new percept and returns an action each time. It retains the complete percept sequence in memory.

---

function REFLEX-VACUUM-AGENT([*location*,*status*]) **returns** an action

   if *status* = *Dirty* **then return** *Suck*
   else if *location* = *A* **then return** *Right*
   else if *location* = *B* **then return** *Left*

Figure 2.8 The agent program for a simple reflex agent in the two-location vacuum environment. This program implements the agent function tabulated in Figure **??**.

---

function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
   persistent: *rules*, a set of condition–action rules

   state ← INTERPRET-INPUT(*percept*)
   rule ← RULE-MATCH(*state*, *rules*)
   action ← *rule*.ACTION
   return *action*

Figure 2.10 A simple reflex agent. It acts according to a rule whose condition matches the current state, as defined by the percept.

---

function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an action
  persistent: *state*, the agent's current conception of the world state
           transition_model, a description of how the next state depends on
               the current state and action
           sensor_model, a description of how the current world state is reflected
               in the agent's percepts
           rules, a set of condition–action rules
           action, the most recent action, initially none

  state ← UPDATE-STATE(*state*, *action*, *percept*, *transition_model*, *sensor_model*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← *rule*.ACTION
  return *action*

Figure 2.12 A model-based reflex agent. It keeps track of the current state of the world, using an internal model. It then chooses an action in the same way as the reflex agent.