

procedure RUN-ENVIRONMENT(*state*, UPDATE-FN, *agents*, *termination*)

inputs: *state*, the initial state of the environment

UPDATE-FN, function to modify the environment

agents, a set of agents

termination, a predicate to test when we are done

repeat

for each *agent* **in** *agents* **do**

PERCEPT[*agent*] ← GET-PERCEPT(*agent*, *state*)

end

for each *agent* **in** *agents* **do**

ACTION[*agent*] ← PROGRAM[*agent*](PERCEPT[*agent*])

end

state ← UPDATE-FN(*actions*, *agents*, *state*)

until *termination*(*state*)

Figure 2.14 The basic environment simulator program. It gives each agent its percept, gets an action from each agent, and then updates the environment.

function RUN-EVAL-ENVIRONMENT(*state*, UPDATE-FN, *agents*,

termination, PERFORMANCE-FN) **returns** *scores*

local variables: *scores*, a vector the same size as *agents*, all 0

repeat

for each *agent* **in** *agents* **do**

PERCEPT[*agent*] ← GET-PERCEPT(*agent*, *state*)

end

for each *agent* **in** *agents* **do**

ACTION[*agent*] ← PROGRAM[*agent*](PERCEPT[*agent*])

end

state ← UPDATE-FN(*actions*, *agents*, *state*)

scores ← PERFORMANCE-FN(*scores*, *agents*, *state*)

until *termination*(*state*)

return *scores*

/ change */*

Figure 2.15 An environment simulator program that keeps track of the performance measure for each agent.