CHAPTER 18

Artificial Intelligence

(Solutions to Practice Set)

Review Questions

1. An interrogator asks a set of questions that are forwarded to a computer and a human being. The interrogator receives two sets of responses: one from the computer and one from the human. After careful examination of the two sets, if the interrogator cannot definitely tell which set has come from the computer, the computer has passed the intelligent test. Some experts think that this is an accurate definition of an intelligent system; some think that the test is not necessarily the definition of an intelligent system.
2. An intelligent agent is a system that perceives its environment, learns from it, and interacts with it intelligently. Intelligent agents can be divided into two broad categories: software agents and physical agents.
3. LISP is a programming language that manipulates lists. LISP treats data, as well as a program, as a list,. This means a LISP program can change itself. This feature matches with the idea of an intelligent agent that can learn from the environment and improves its behavior. PROLOG is a language that can build a database of facts and a knowledge base of rules. A program in PROLOG can use logical reasoning to answer questions that can be inferred from the knowledge base.
4. If an artificial agent is supposed to solve some problems relating to the real world, it needs somehow to represent knowledge. Four common methods for representing knowledge are semantic networks, frames, predicate logic, and rule-based system.
5. Propositional logic is a language made of a set of sentences that can be used to do logical reasoning about the world. In propositional logic, a symbol that represents a sentence is atomic; it cannot be broken to find some information about its components. To do so, we need predicate logic, the logic that defines the relation between the parts in a proposition.
6. Semantic networks use vertices (nodes) to represent concepts and edges (arrows) to represent the relations between concepts. In frames, data structures (records) are used to represent the same knowledge. Nodes and edges in semantic networks becomes objects and slots, respectively, in frames.

7. A ruled-based system represents knowledge using a set of rules that can be used to deduce some new facts from already-known facts. The semantic network is a graphical representation of entities and their relationships.
8. Expert systems perform tasks expected from a human with expertise. Mundane systems perform tasks that does not require expertise (such as talking or walking).
9. The five stages of image processing are edge detection, segmentation, finding depth, finding orientation, and object recognition.
10. The four steps in language processing are speech recognition, syntactic analysis, semantic analysis, and pragmatic analysis.
11. Neural networks try to simulate the learning process of the human brain using a networks of artificial neurons.
12. A perceptron is an artificial neuron similar to a single biological neuron.

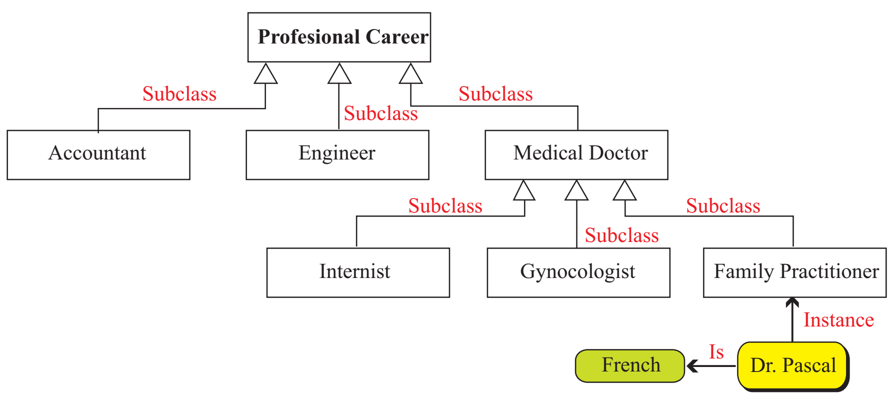
Multiple-Choice Questions

13. d	14. b	15. c	16. c	17. d	18. d
19. b	20. c	21. c	22. b	23. a	24. c
25. c					

Exercises

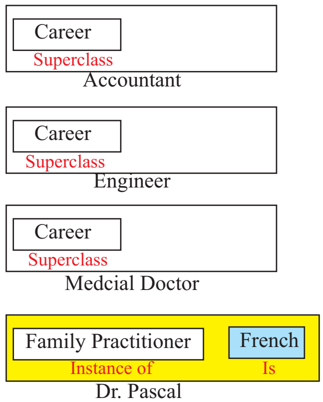
26. The semantic network is shown in Figure S18.26.

Figure S18.26 *Exercise 26*



27. The set of frames are shown in Figure S18.27.

Figure S18.27 *Exercise 27*



28.

a. ¬R

b. ¬S

c. (¬R) ∨ (¬S)

d. R ∧ S

e. S → (¬R)

f. R → (¬S)

g. S ↔ (¬R) or (¬R) ↔ S

h. ¬[R → (¬S)]

29.

a. It is not hot.

b. It is warm or it is hot.

c. It is warm and hot.

d. It is warm but it is not hot.

e. It is not true that it is warm and hot.

f. If it is warm, then it is hot.

g. If it is not cold, then it is warm.

h. It is not true that if is not warm, then it is hot.

i. It is hot if it is not warm.

j. It is not cold and hot, or it is cold and not hot.

30.

a. $\exists x[Fl\,(x) \wedge Wh\,(x)]$

b. $\exists x[Fl\,(x) \wedge \neg\, Re(x)]$

c. $\neg\forall x[Fl\,(x) \rightarrow Re\,(x)]$

d. $\exists x\{Fl\,(x) \wedge [Re\,(x) \vee Wh\,(x)]\}$

e. $\neg\exists x[Fl\,(x) \wedge \mathrm{Gr}(x)]$

f. $\neg\exists x[Fl\,(x) \wedge \mathrm{Gr}(x)]$

g. $\exists x[Fl\,(x) \wedge \neg\, Wh(x)]$

31.

a. $\exists x[Cat\,(x) \rightarrow Has\,(\mathrm{John}, x)]$

b. $\forall x[Cat\,(x) \rightarrow Loves\,(\mathrm{John}, x)]$

c. $Loves\,(\mathrm{John}, \mathrm{Anne})$

d. $\exists x[Dog\,(x) \wedge Loves\,(\mathrm{Anne}, x)]$

e. $\exists x[\neg\, Cat\,(x) \wedge Loves\,(\mathrm{John}, x)]$

f. $\exists x[\, Cat\,(x) \wedge \neg\, Loves\,(\mathrm{Anne}, x)]$

g. $\exists x\{[\, Cat\,(x) \wedge \neg\, Loves\,(\mathrm{John}, x)] \rightarrow Loves\,(\mathrm{Anne}, x)\}$

h. $\exists x\{[\, Cat\,(x) \wedge \neg\, Loves\,(\mathrm{John}, x)] \leftrightarrow Loves\,(\mathrm{Anne}, x)\}$

32.

a. $\forall x[Expensive\,(x)]$

b. $\forall x[Cheap\,(x)]$

c. $\forall x[Cheap\,(x) \rightarrow Buys\,(\mathrm{Bob}, x)]$

d. $\forall x[Expensive\,(x) \rightarrow Sells\,(\mathrm{John}, x)]$

e. $\neg\forall x[Expensive\,(x)]$

f. $\neg\forall x[Cheap\,(x)]$

g. $\exists x\{[\, Cheap\,(x)] \rightarrow \neg\, Expensive\,(x)\}$

33.

a. $\neg\, Identical\,(\mathrm{John}, \mathrm{Anne})$

b. $\exists x[\mathrm{John}\,(x)]$

c. $\neg\exists x[\mathrm{Anne}\,(x)]$

d. $\exists x$

e. $\neg\exists x$

f. $\exists x \exists y[\neg\, Identical\,(x, y)]$

34. The truth table is shown below. The argument $\{P \rightarrow Q, P\} \vdash Q$ is valid.

P	Q	$P \rightarrow Q$	P	Q
F	F	T	F	**F**
F	T	T	F	**T**
T	F	F	T	**F**
T	T	T	T	**T**
		Premise	Premise	Conclusion

OK

35. The truth table is shown below. The argument {P ∨ Q, P} |– Q is not valid:

P	Q	P ∨ O	P	Q	
F	F	F	F	**F**	
F	T	T	F	**T**	
T	F	T	T	**F**	**Counterexample**
T	T	T	T	**T**	**OK**
		Premise	Premise	Conclusion	

36. The truth table is shown below. The argument {P ∧ Q, P} |– Q is valid:

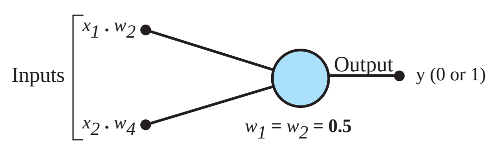
P	Q	P ∧ O	P	Q	
F	F	F	F	**F**	
F	T	F	F	**T**	
T	F	F	T	**F**	
T	T	T	T	**T**	**OK**
		Premise	Premise	Conclusion	

37. The truth table is shown. The argument {P → Q, Q → R} |– (P → R) is not valid

P	Q	R	P → Q	Q → R	P → R	
F	F	F	T	T	**T**	**OK**
F	F	T	T	T	**T**	**OK**
F	T	F	T	F	**T**	**Counterexample**
F	T	T	T	T	**T**	**OK**
T	F	F	F	T	**F**	
T	F	T	F	T	**T**	
T	T	F	T	F	**F**	
T	T	T	T	T	**T**	**OK**
			Premise	Premise	Conclusion	

38. The design of the neural network, with weights $w_1 = w_2 = 0.5$ and the threshold of $T = 0.5$, is shown in Figure S18.38.

Figure S18.38 *Exercise 38*

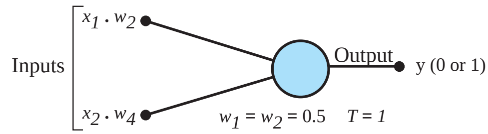


The truth table for this neural network is shown below. It is the same as the truth table for an OR gate.

Inputs		$S = x_1 . w_1 + x_2 . w_2$	Compare S with T	Output
0	0	0	$S < T$	0
0	1	0.5	$S = T$	1
1	0	0.5	$S = T$	1
1	1	1	$S > T$	1

39. The design of neural network, with weights $w_1 = w_2 = 0.5$ and the threshold of $T = 1$, is shown in Figure S18.39.

Figure S18.39 *Exercise 39*

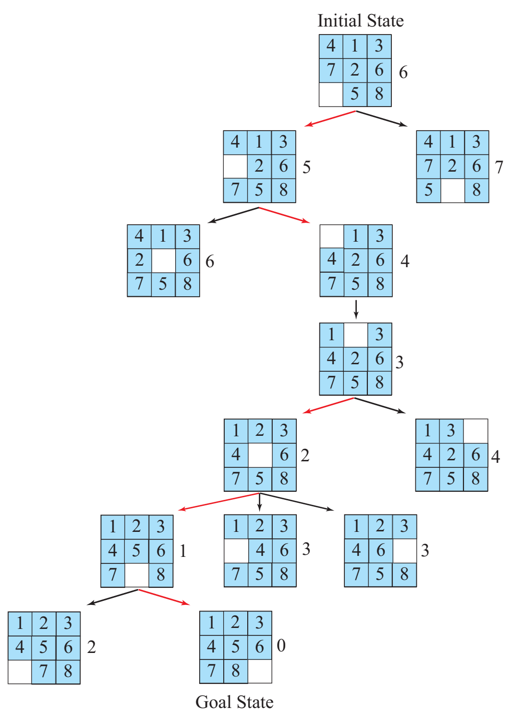


The truth table for this neural network is shown below. It is the same as the truth table for an AND gate.

Inputs		$S = x_1 \,.\, w_1 + x_2 \,.\, w_2$	Compare S with T	Output
0	0	0	$S < T$	0
0	1	0.5	$S < T$	0
1	0	0.5	$S < T$	0
1	1	1	$S = T$	1

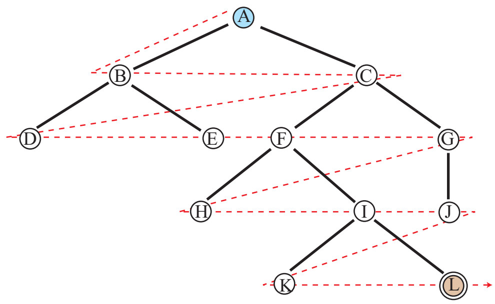
40. The heuristic search tree for solving the puzzle is shown in Figure S18.40.

Figure S18.40 *Exercise 40*



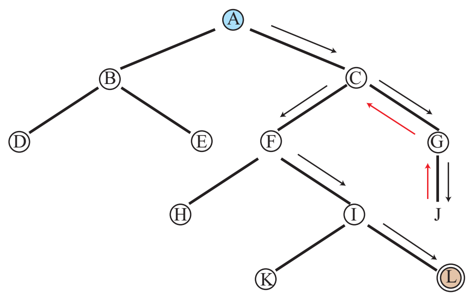
41. Figure S18.41 shows the breadth-first search for the tree diagram.

Figure S18.41 *Exercise 41*



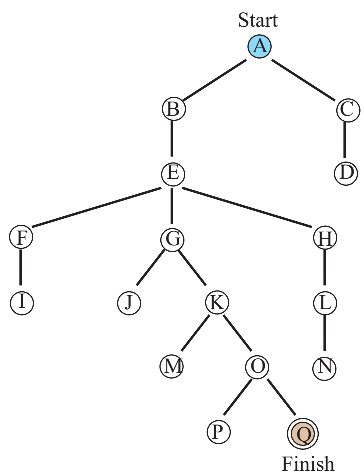
42. Figure S18.42 shows the depth-first search for the tree diagram of Exercise 41.

Figure S18.42 *Exercise 42*



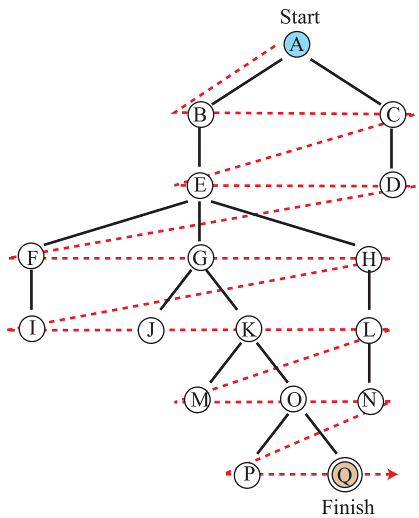
43. Figure S18.43 shows the tree diagram for the maze.

Figure S18.43 *Exercise 43*



44. Figure S18.44 shows the breadth-first search for Exercise 43.

Figure S18.44 *Exercise 44*



45. Figure S18.45 shows the depth-first search for Exercise 43.

Figure S18.45 *Exercise 45*