Massachusetts Institute of Technology

6.034 Artificial Intelligence

Assignment # **3** Due: **9/29/98** 6.034 Item # **6**

PS 3 Solutions

Problem 1

See next page for tree, annotated with path.

Problem 2

Part A

The indication that a rule is "irrelevant" indicates that the THEN portion of the rule cannot be matched to the hypothesis under consideration, as for example `(Tony is a bird)` cannot match `(Tony is a mammal)`.

Part B

The problem is that the `ask?` function does not record the answers given by the user to questions. To repair this we add a `remember-assertion` call to `ask?` and keep track of negative/unknown answers as well:

```
;;; this will keep track of questions answered (n) or (u)
(define *questionsansno* '())

(define (ask? goal bindings)
;;;
;;; Prompts the user to see whether goal is true
;;;
  (let ((i-goal (instantiate-variables goal bindings)))
    (cond ((member i-goal *assertions*) ; check if already known true
   bindings)                     ; if so, return bindings
  ((member i-goal *questionsansno*)  ; check if already known false
   '())                              ; if so, return nil w/o asking
  (else
   (begin
     (format #t (indentit "Have no assertion or successful rule to determine whether"))
     (format #t " ~a." i-goal)
     (format #t (indentit "Is this true [(y) or (n)] or (u)? "))
     (let ((ans (read)))
       (cond ((equal? ans '(y))
   ;;; if we get a yes answer, record the assertion
      (remember-assertion i-goal)
      bindings)
     ((or (equal? ans '(n)) (equal? ans '(u)))
   ;;; if we get a no or unknown answer, remember that we asked
      (set! *questionsansno* (cons i-goal *questionsansno*))
      '())
     (else
      (format #t "~%Answer must be either (y) or (n) or (u).")
      (ask? goal bindings)))))))))
```

With this modification, the system keeps track of positive answers in the assertions and the other answers in the global variable. As a result it asks the question about whether `(Tony eats meat)` only once, and when it is done, it has a larger collection of assertions in its database.

```
Final data base contents:
(tony is a tiger)
(tony has black stripes)
(tony has tawny color)
(tony is a carnivore)
(tony has forward-pointing eyes)
(tony has claws)
(tony has pointed teeth)
(tony is a mammal)
(tony gives milk)
;Unspecified return value
```

Problem 3

Part A

The system infers that Tony is a mammal.

Part B

The system does this because all of its conclusions are written in the form of `(Tony is a ...)`, which is too general a pattern, given that the rules are actually making conclusions about several different kinds of things about the animals, such as their diet, their class (ungulate, bird), and their species (tiger, cheetah, etc.)

Part C

The following set of rules makes the appropriate distinctions, solving the problem at the knowledge level, without depending on the accident of rule ordering.

```
(define (make-zoo-rules)
  (clear-rules)
  (remember-rules
   '(Z1 IF
        ((? animal) has hair)
        THEN ((? animal) type is mammal))
   '(Z2 IF
        ((? animal) gives milk)
        THEN ((? animal) type is mammal))
   '(Z3 IF
        ((? animal) has feathers)
        THEN ((? animal) class is bird))
   '(Z4 IF
        ((? animal) flies)
        ((? animal) lays eggs)
        THEN ((? animal) class is bird))
   '(Z5 IF
        ((? animal) type is mammal)
        ((? animal) eats meat)
        THEN ((? animal) diet is carnivore))
   '(Z6 IF
        ((? animal) type is mammal)
        ((? animal) has pointed teeth)
        ((? animal) has claws)
        ((? animal) has forward-pointing eyes)
        THEN ((? animal) diet is carnivore))
   '(Z7 IF
        ((? animal) type is mammal)
        ((? animal) has hoofs)
        THEN ((? animal) class is ungulate))
```

```
 '(Z8 IF
      ((? animal) type is mammal)
      ((? animal) chews cud)
      THEN ((? animal) class is ungulate))
 '(Z9 IF
      ((? animal) diet is carnivore)
      ((? animal) has tawny color)
      ((? animal) has dark spots)
      THEN ((? animal) species is cheetah))
 '(Z10 IF
       ((? animal) diet is carnivore)
       ((? animal) has tawny color)
       ((? animal) has black stripes)
       THEN ((? animal) species is tiger))
 '(Z11 IF
       ((? animal) class is ungulate)
       ((? animal) has long neck)
       ((? animal) has long legs)
       ((? animal) has dark spots)
       THEN ((? animal) species is giraffe))
 '(Z12 IF
       ((? animal) class is ungulate)
       ((? animal) has black stripes)
       THEN ((? animal) species is zebra))
 '(Z13 IF
       ((? animal) class is a bird)
       ((? animal) does not fly)
       ((? animal) has long legs)
       ((? animal) has long neck)
       ((? animal) is black and white)
       THEN ((? animal) species is ostrich))
 '(Z14 IF
       ((? animal) class is bird)
       ((? animal) does not fly)
       ((? animal) swims)
       ((? animal) is black and white)
       THEN ((? animal) species is penguin))
 '(Z15 IF
       ((? animal) class is bird)
       ((? animal) flies well)
       THEN ((? animal) species is albatross)))
(format #t "~%Zookeeper rules recorded."))
```