Project 2 is *more or less* the same as project 1, but in LISP.

Some differences:

1. inputs will be in the form
   (input '(...))

   Note that the ' tells lisp to treat what follows as a list with elements rather than a function with arguments.

   Also note that this means that your program must define a function named input which takes a single argument (a list of words) and returns another list of words. Unlike in the last project, it is unnecessary to print anything to the console - correctness will be determined by the return value of the input function.

2. I/O specs (input -> output):

   (x is a/an y) ->
   if already known: (you already taught me that)
   else: (ok.)

   (is x a/an y) ->
   if known(x is a/an y): (yes)
   else if known(x is a/an (z != y)): (no)
   else (x has not been associated with any type): (I don't know the word x)

   (what is x) ->
   for each y0...yn such that known(x is a/an yi): (x is a/an y0. x is a/an y1...x is a/an yn)
   if no such y exists: (I don't know the word x)

   -note that this is different from the first project; your output must report each category that x belongs to. See below for an example.

   (anything else) -> (huh?)

3. Misc:

   ***important note: the facts "x is a y" and "x is an y" are to be treated separately for this project, and you should not assume that one implies the other. Therefore if your program has learned "x is a y" and is told "x is an y," it should say "ok" rather than "you already taught me that,"
and similarly for other valid queries.

-upper/lowercase distinctions are not important. All valid (non-gibberish) inputs may be assumed to be in lower case and contain no punctuation.

-listing the categories that a word belongs to can occur in any order, so long as all are listed with the correct article(s) associated.

Example IO sequence:

(input '(what is john)) -> (I don't know the word john)

(input '(apple is a noun)) -> (ok)

(input '(is apple a verb)) -> (no)

(input '(is john a verb)) -> (I don't know the word john)

(input '(apple is an adverb)) -> (ok)

(input '(is apple a adverb)) -> (no)

;;apple is an adverb, not a adve

(input '(what is apple)) -> (apple is a verb. apple is an adverb.) ;;these can be in any order.

(input '(apple is an adverb)) -> (you already taught me that)

(input '(what is adverb)) -> (I don't know the word adverb)

;;note that category names are not considered known words.

(input '(apple is a adverb)) -> (ok)

(input '(what is apple)) -> (apple is a verb. apple is an adverb. apple is a adverb.)