```
-- Pretend that list is not a builtin type, with
-- special syntax, of the language.
-- Define your own type list.
-- Define two functions on this type,
-- one to count own many elements are in a list,
-- the other to find whether some element is in a list.

data UserList a = Nil | Cons a (UserList a)

count Nil = 0
count (Cons _ xs) = 1 + count xs

find x Nil = False
find x (Cons y ys) = x==y || find x ys

-- count (Cons 1 (Cons 2 (Cons 3 Nil))) => 3
-- find 'a' Nil => False
-- find 'a' (Cons 'x' (Cons 'a' (Cons 'y' Nil))) => True
```