a) Write a DefineGlobal rule to show the operational semantics of such a val

$$\frac{\langle set(x,e), \rho\{x \mapsto \ell\}, \sigma\rangle \Downarrow \langle v, \sigma'\rangle}{\langle VAL(x,e), \rho, \sigma\rangle \rightarrow \langle \rho\{x \mapsto \ell\}, \sigma'\rangle}$$

(Notice that whether $x \in dom\ \rho$ or $x \notin dom\ \rho$ doesn't matter)

b) (define check_Val_semantics
    (begin
    (val x 2) ← if the DefineGlobal rule is applied, this will be defined and you can access that variable [If it isn't, that will produce the error]
    (print x)
)

I'm trying to determine whether ↓ val has created a new variable

c) Compare and contrast the 2 ways of defining val.
I like the old method where a val binding of a name that is already bound is equivalent to set. Although it might be "easier" if val always creates a new binding, you could get some unexpected side effects when you are setting values when they don't actually exist. This would mean that when you try to access a variable it may or may not be there and it may or may not be what you are looking.

Ian [illegible]
#37