Proof Problems

Exercise 46:

a)

$$\frac{\ell \notin \text{dom } \sigma \qquad \langle e, \rho\{x \mapsto \ell\}, \sigma\{\ell \mapsto \mathit{unspecified}\}\rangle \Downarrow \langle v, \sigma'\rangle}{\langle VAL(x, e), \rho, \sigma\rangle \rightarrow \langle \rho\{x \mapsto \ell\}, \sigma'\{\ell \mapsto v\}\rangle} \quad \text{DEFINE-GLOBAL}$$

Explanation: Since val always creates a new binding and never behaves like set, we always want val to allocate a fresh location ℓ and extend the environment to bind variable x to ℓ. Thus, DEFINE-GLOBAL is almost identical to the DEFINE-NEW-GLOBAL rule. However, we need to make sure that it binds when x is not in ρ as well. Thus, the only tweak we have to make is to get rid of the first premise.

b)

Code: (from cqs)

(val f (lambda () y))

(val y 10)

(f)

Explanation: Under the Scheme semantics, the lambda already captures a spot in memory so that second val behaves like set and sets f to 10. Under the new semantics, this code would create a runtime error because y sets a new spot of memory that isn't defined in the first line of code.

c)

I think that the new semantics make the overall semantics much easier, but the code and coding style would be a lot more complex to avoid the run time errors of undefined variables. I would prefer the old semantics.