

1: Opsem and Language Design

a)

$$\frac{\langle e, \rho, \sigma \rangle \Downarrow \langle v, \rho, \sigma \rangle}{\langle \text{VAL}(x, e), \rho, \sigma \rangle \Downarrow \langle v, \rho\{x \rightarrow l\}, \sigma\{l \rightarrow v\} \rangle} \text{DEFINEGLOBAL}$$

b)

```
(val y 0)
(val f (lambda () y))
(val y 1)
(if (= (f) 0) 'new-semantic 'old-semantic)
```

The first line binds a new variable either way, binding y to a memory location, l_1 and binding that memory location to 0. then, it defines a function that evaluates to whatever is in l_1 . In the old semantics, the 3rd line will change l_1 to be 1, but in the old semantics, this line will create a new binding of y to a different memory location, l_2 , and bind that location to 1. In the next line, we evaluate f which tells us whatever is in l_1 . If it is still 0, we know it is the new semantics because l_1 wasn't updated, otherwise the 3rd line must have updated it, consistent with the old semantics.

c) I prefer the current method of evaluating `val` because the new rule seems prone to errors. If you forget that you had previously bound a variable and are accessing it indirectly, that source could not have been updated causing unexpected issues.