Exercise 3.34.

Louis Reasoner wants to build a squarer, a constraint device with two terminals such that the value of connector `b` on the second terminal will always be the square of the value `a` on the first terminal. He proposes the following simple device made from a multiplier:

```
(define (squarer a b)
  (multiplier a a b))
```

There is a serious flaw in this idea. Explain.

Answer.

Louis's `squarer` constraint system above is erected as one-directional computation, which is only competent with computing the square of a number

```
(set-value! a 16 'user)
Probe: a = 16
Probe: b = 256
;Value: done
```

This procedure fails to give us the answer when is ordered to compute the square root of a number

```
(forget-value! a 'user)
Probe: a = ?
Probe: b = ?
;Value: done

(set-value! b 256 'user)
Probe: b = 256
;Value: done
```

even though the computations of both arise from the same equation.