pd2 hw33 2021-11-08   Ivan Chen

worked with no one; advised by no one

tracing `equalify`

```
(define weakAfterBoost                                    ; line 01
```

1. Look for symbol in the symbol table.
2. ...If a slot named for symbol is found, reuse that slot in the next steps, otherwise, reserve new slot and use the new slot in the following steps.

```
  (lambda (strong weak) (strongGetsMore weak strong))   ; line 02
```

1. Put the define on hold
2. ...Build a procedure, recording `strong, weak` as the procedure's list of parameters and package the instructions `(strongGetsMore weak strong)` as the expression to be evaluated in and when this procedure is invoked.

```
  )                                                       ; line 03
```

1. Continue the define.
2. ...Store the value in the symbol table slot for symbol.

```
(define strongGetsMore                                    ; line 05
```

1. Look for symbol in the symbol table.
2. ... If a slot named for symbol is found, reuse that slot in the next steps, otherwise, reserve new slot and use the new slot in the following steps.

```
  (lambda (strong weak) (+ strong (/ weak 2)))          ; line 06
```

1. Put the define on hold
2. ... Build a procedure, recording `strong, weak` as the procedure's list of parameters and package the instructions `(+ strong (/ weak 2))` as the expression to be evaluated in and when this procedure is invoked.

```
  )                                                       ; line 07
```

1. Continue the define.
2. ...Store the value in the symbol table slot for symbol.

```
(display                                                  ; line 09
```

1. Prepare to display dots of light on screen with the value of the expressions.
2. ...Continue to the arguments of display.

```
(weakAfterBoost 10 2)                                    ; line 10
```

1. Put display on hold.
2. ...Invoke the weakAfterBoost procedure, copy the value of each argument into the corresponding parameter of the procedure, evaluate the expressions in the procedure, using the copied values, and replace the invocation with the last value computed by the procedure. Value is 7.

```
)                                                        ; line 11
```

1. Continue the display.
2. ...Display dots of light on the screen in the shape of the resulting value (7) of the expression above.