order of operations

Yedoh Kang

PD 6

HW 19

2015-10-14

# procedure calls, in chronological order

|  |  |  |  |
| --- | --- | --- | --- |
| **procedure** | **justification** | **addition to REPL pane** | **other effects on Racket** |
| (display "hw19: nested procedure calls") | (display value) → Displays *value* in the REPL pane | hw19: nested procedure calls |  |
| (newline) |  | **⤶** |  |
| (lambda…) |  |  | instructions are bound into a procedure |
| (define diff0fSquares…) |  |  | procedure is bound to the symbol diff0fSquares |
| (lambda…) |  |  | instructions are bound into a procedure |
| (define square…) |  |  | procedure is bound to the symbol square |
| (diff0fSquares 2 1) |  |  | call diff0fsquares |
| (display “via diff0fSquares”) |  | via diffOfSquares |  |
| (newline) |  | **⤶** |  |
| (square b) |  |  | call square |
| (display “number = “) |  | number = |  |
| (display num) |  | 1 |  |
| (newline) |  | **⤶** |  |
| (\* 1 1) |  |  |  |
| (square a) |  |  | call square |
| (display "number = “ |  | number = |  |
| (display num) |  | 2 |  |
| (newline) |  | **⤶** |  |
| (\* 2 2) |  |  | 4 is held in Racket’s memory |
| (- 1 4) |  |  | -3 is held in Racket’s memory |
| (display -3) |  | -3 |  |
| (newline) |  | **⤶** |  |
| (display “—————“) |  | —————————— |  |
| (newline) |  |  |  |
| (diff0fSquares -3 4) |  |  |  |
| (display “via diff0fSquares”) |  | via diffOfSquares |  |
| (newline) |  | **⤶** |  |
| (square b) |  |  | call square |
| (display “number = “) |  | number = |  |
| (display num) |  | 4 |  |
| (newline) |  | **⤶** |  |
| (\* 4 4) |  |  | 16 is held in Racket’s memory |
| (square a) |  |  | call square |
| (display "number = “ |  | number = |  |
| (display num) |  | -3 |  |
| (newline) |  | **⤶** |  |
| (\* -3 -3) |  |  | 9 is held in Racket’s memory |
| (- 16 9) |  |  | 7 is held in Racket’s memory |
| (display 7) |  | 7 |  |
| (newline) |  | **⤶** |  |

# 