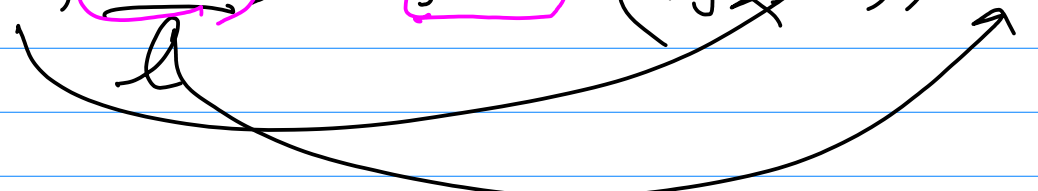


foldl f = folder
where

folder (z, []) = z
folder (z, x::xs) = folder (f z x, xs)



while l ≠ []
 z ← f z x
 l ← tail l
return z

tail recursion

int f(int x)

3

int y;

;

3

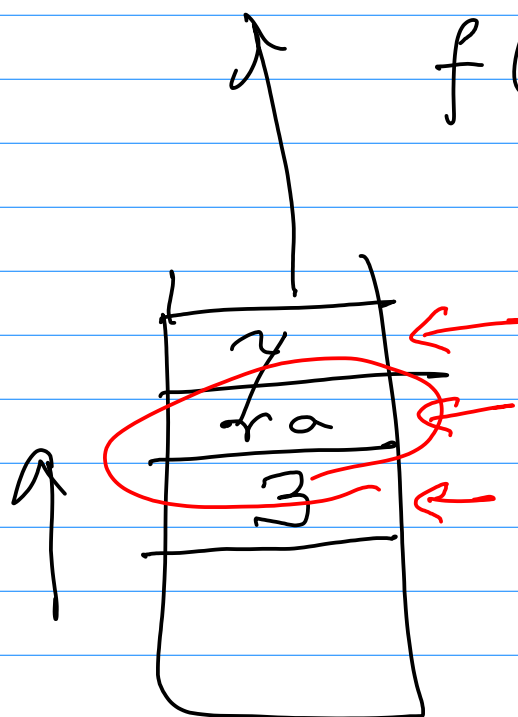
main ()

3

$$f(3)$$

bool

$f \rightarrow f$

$$f(x, y, z)$$
$$f(a, b, c)$$


A red curly brace is drawn on blue-lined paper. It is positioned vertically, spanning from the second line to the fourth line. The brace is open on the left side and closed on the right side.

Pauli

"It is not even wrong"

C Python

PyPy

foldl —

tail recursive

foldr —

?? recursive

etype $\llbracket a \rrbracket$ where

①

$(!:) : a \rightarrow \llbracket a \rrbracket \rightarrow \llbracket a \rrbracket$

②

$\llbracket \rrbracket : \llbracket a \rrbracket$

datatype Tree, a where

Lf : a → Tree.a

Bf : a → Tree.a → Tree.a → Tree.a

fold Tree