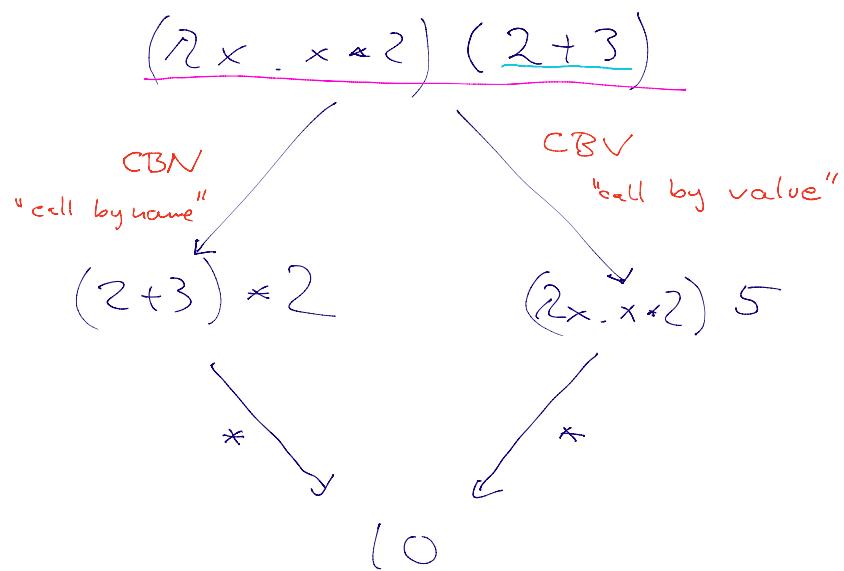
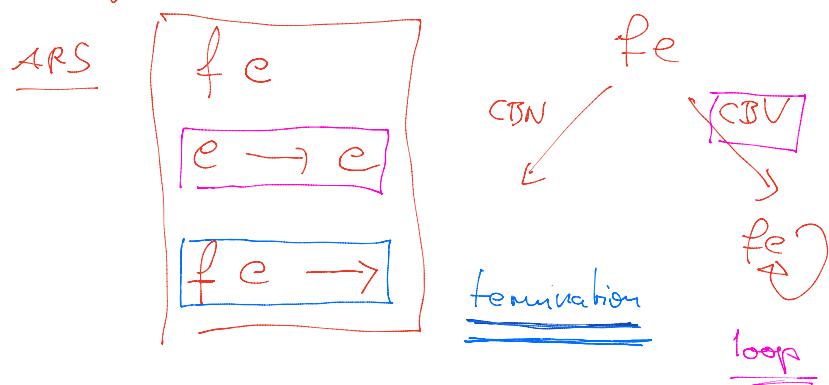


## CBN vs CBV



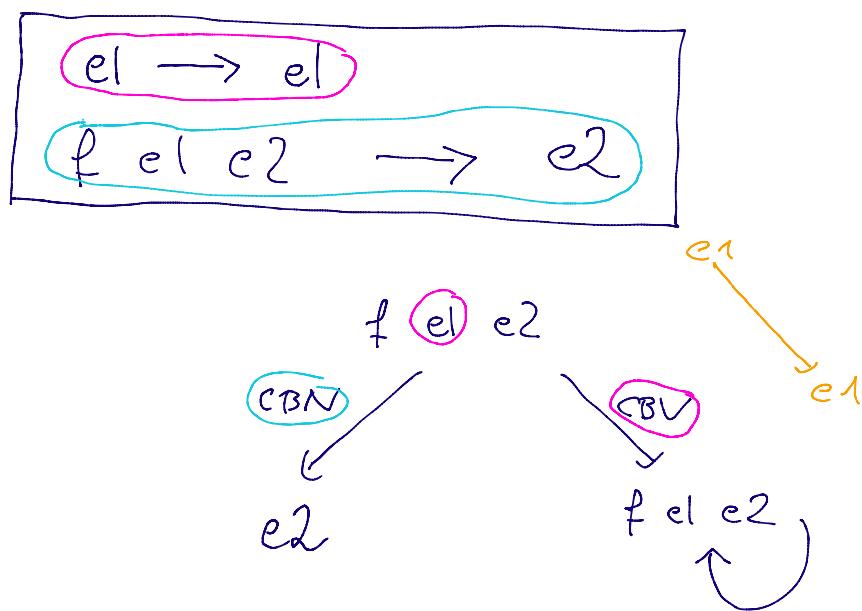
computations can be non terminating

CBV : what if evaluating the argument results in an infinite loop ?



$$(\lambda x. \lambda y. y) ((\lambda x.x) (\lambda x.x)) z$$

$f$                      $e_1$                      $e_2$



Haskell

"lazy"

Java

C

Python

Ocaml, ML, ...

"eager" "strict"

if              then              else

$e_1$  AND  $e_2$   
false

$e_1 \text{ AND } e_2 \rightarrow \text{false}$

if

$e_1 \rightarrow \text{false}$
$e_2 \rightarrow e_2$

$e_1 \text{ AND } e_2 \rightarrow e_1 \text{ AND } e_1$

if

$e_1 \rightarrow e_1$

How to add numbers to  $\lambda$ -calculus?

$\text{eval(CBN } (\underline{s(s0)}) \text{)} = \underline{s(s0)}$

$\text{eval(CBN } \underline{\circ} \text{)} = \underline{\circ}$
---

concrete syntax

how in abstract syntax  
(given by the names of the  
rules of the grammar)

IN THE INTERPRETER

$\text{eval(CBN } E_{\text{Nat0}} \text{)} = E_{\text{Nat0}}$
---

$\text{eval(CBN } (E_{\text{NatS}} e) = E_{\text{NatS}} (\text{eval(CBN } e))$
--

