

## Lecture 4

What function calls "mean" (Python version)

def ite( $e_1$ ,  $e_2$ ,  $e_3$ ): return  $e_2$  if  $e_1$  else  $e_3$

ite( $3 > 2$ , 'Big', 'S'+ 'mall')  $\leftarrow$

When you see a function call:

① Evaluate all the arguments

$3 > 2 \Rightarrow^* \text{True}$

'Big'  $\Rightarrow^* \text{'Big'}$

'S'+ 'mall'  $\Rightarrow^* \text{'Small'}$

② Substitute the results into the fn. body

if True then 'Big' else 'Small'

③ Evaluate the grounded fn body

'Big'.

"Eager"  
evaluation

cf.  
"Lazy"  
evaluation

ite (3 > 2, 'Big', head [])

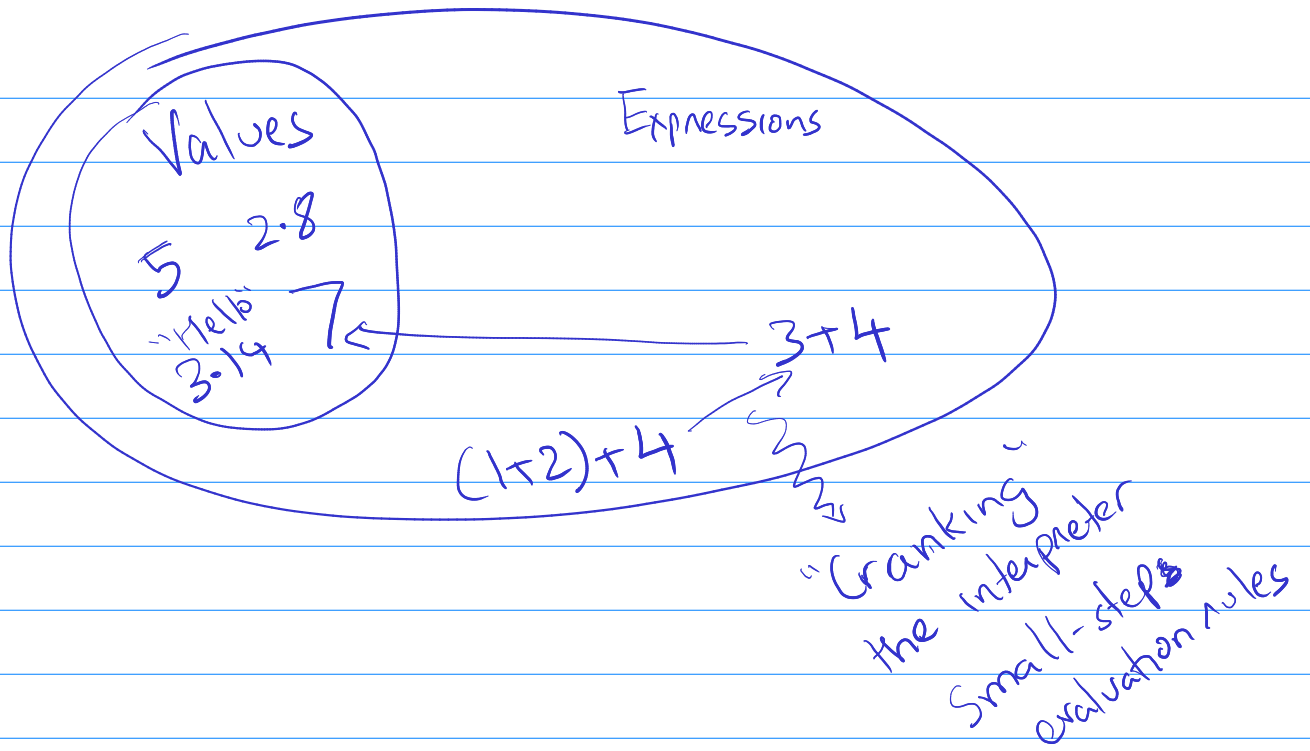
① Evaluate all arguments

3 > 2  $\Rightarrow^*$  True

'Big'  $\Rightarrow^*$  'Big'

head []  $\Rightarrow^*$  Explosion





## Rules for evaluating conditionals

if  $e_1$  then  $e_2$  else  $e_3$

$\Rightarrow^*$  if  $v_1$  then  $e_2$  else  $e_3$

$e_1$  is fully reduced to some value  $v_1$

$e_2$  &  $e_3$  are not touched.

Guarantee: Either  $v_1 = \text{true}$  or  $v_1 = \text{false}$ .

(If  $v_1 = \text{true}$ )  $\Rightarrow^*$   $e_2 \Rightarrow^* \dots \Rightarrow^*$   $v_2$

(Otherwise, if  $v_1 = \text{false}$ )  $\Rightarrow^*$   $e_3 \Rightarrow^* \dots \Rightarrow^*$   $v_3$

List.length ([1;2;3]@[4;5;6])

How to crank function applications

$f\ e \Rightarrow^* f\ v$  ← Fully evaluate expression  $e$ .

Then, substitute  $v$  into the body of  $f$ .

Finish evaluating / fn-body.  
grounded