arguments and default values

specification

conditions
guarantees

What is recursion?

a way to design solutions to problems by divide and conquer or decrease and conquer

a programming technique where a function calls itself

to be able to use recursion 1 or more base cases need to be defined

thus the larger problem must simplifiable

iterations

looping constructs (while and for loops) lead to iterative algorithms

can capture computation in a set of state variables that update on each iteration through a loop.

a + a + a ..... + a = a * b

b times

a * b = a + a + a ..... + a

b-1 times

a * b = a + a * (b - 1)

a + a * (b -2)

The base case: when b = 1, a * b = 1

nf = n*(n-1) * (n-2) * (n-3) ….. * 1

add last add - def fact(n):

n = 1

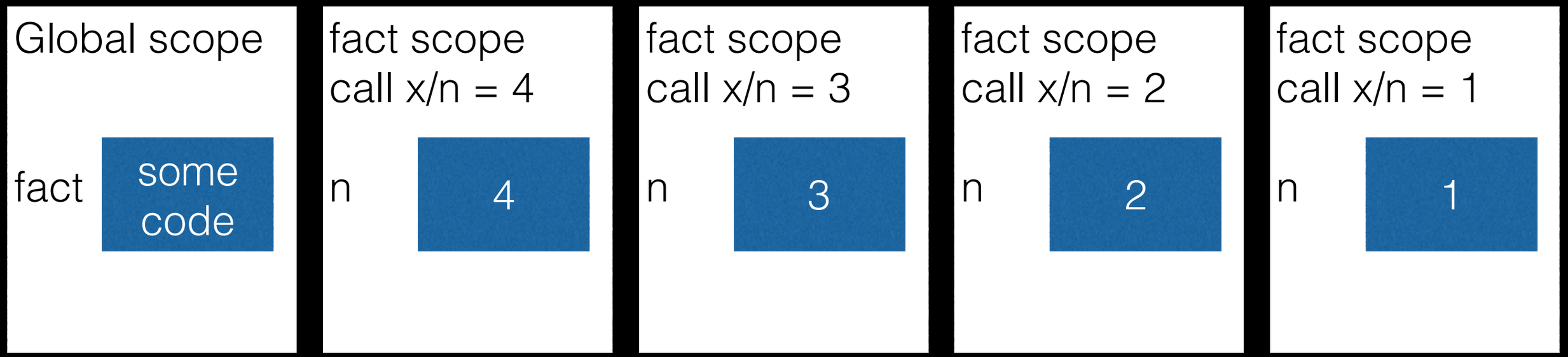
```
if n == 1:
    return 1
```

base case

n*(n - 1)

```
else:
    return n*fact(n-1)
```

recursive step



print fact(4)

return 4* fact(3)

return 3* fact(2)

return 2* fact(1)

return 1

print 24

return 4*6

return 3*2

return 2*1

base case

iteration vs. recursion

Finally

FINE