

CSE 20

11-30-20

11

- o lab4: ext. 1 final day
- o Pa7: ext. 2 more days

Inverting a dictionary

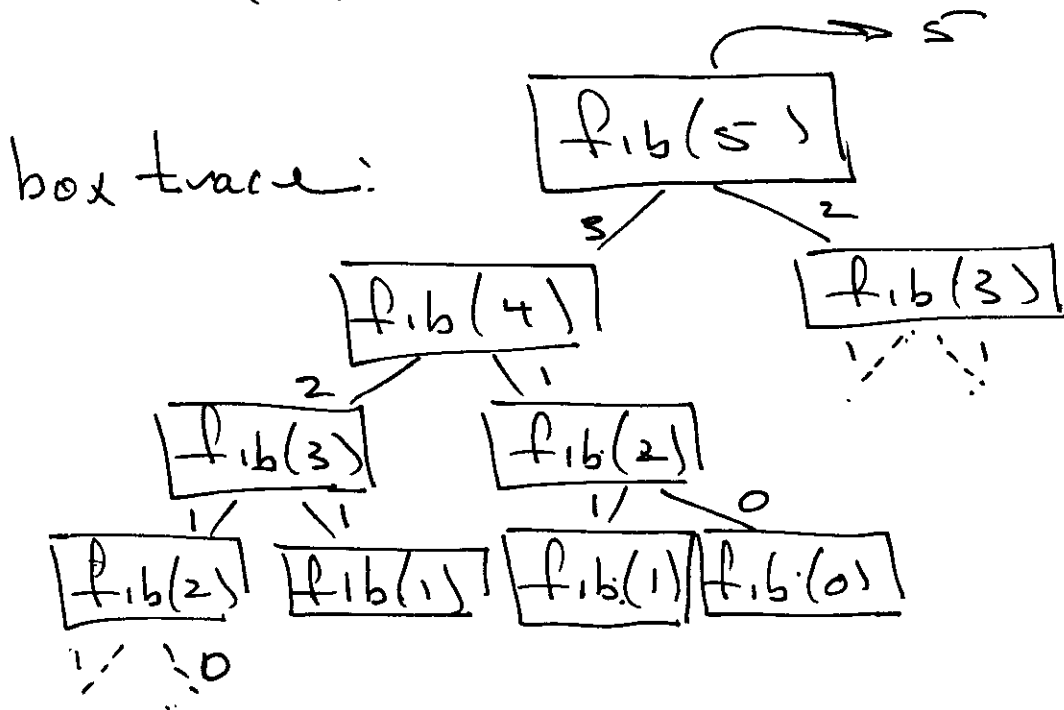
1 : 'a'		'a' : [1, 3]
2 : 'b'		'b' : [2, 5]
3 : 'a'	invert →	'c' : [4]
4 : 'c'		
5 : 'b'		

Ex. Fibonacci 3. Py
4. Py

Fibonacci seq:

ind: 0 1 2 3 4 5 6 7 8 9
0, 1, 1, 2, 3, 5, 8, 13, 21, 34,
initial
terms

$$fib(n) = fib(n-1) + fib(n-2)$$



use a dictionary for memorization.
 each time $\text{fib}(n)$ is called
 check a dictionary of known
 values. if $\text{fib}(n)$ is there,
 return it, otherwise recur on
 $\text{fib}(n-1)$ and $\text{fib}(n-2)$, and
 return the sum.

	<u>keys: n</u>	<u>values: fib(n)</u>
initial	0	0
	1	1
	2	1
	3	2
	4	3
	5	5

14.9 module

4

we've seen many module

- math
- random
- time

How to build a module --

next time !

lab4 stuff:

command line mode

\$ Calc 1 + 1

sys.args

0	1	2	3
'Calc'	'1'	'+'	'1'

len
4

sys.args[1:]

' '.join()

eval()

\$ Calc

sys.args

'Calc'

interactive mode

len
1

Calc

A

$$A[0] = 1 + 1$$

$$[2]$$

$$A[1] = 2 + 2$$

$$[2, 4]$$

$$A[2] = A[0] + A[1]$$

$$[2, 4, 6]$$

Ex.

$$L = [5, 7]$$

$$\text{Print}(\text{eval}('L[0]')) \quad \# \quad 5$$

$$\text{Print}(\text{eval}('L[1]')) \quad \# \quad 7$$

$$\text{Print}(\text{eval}('L')) \quad \# \quad [5, 7]$$