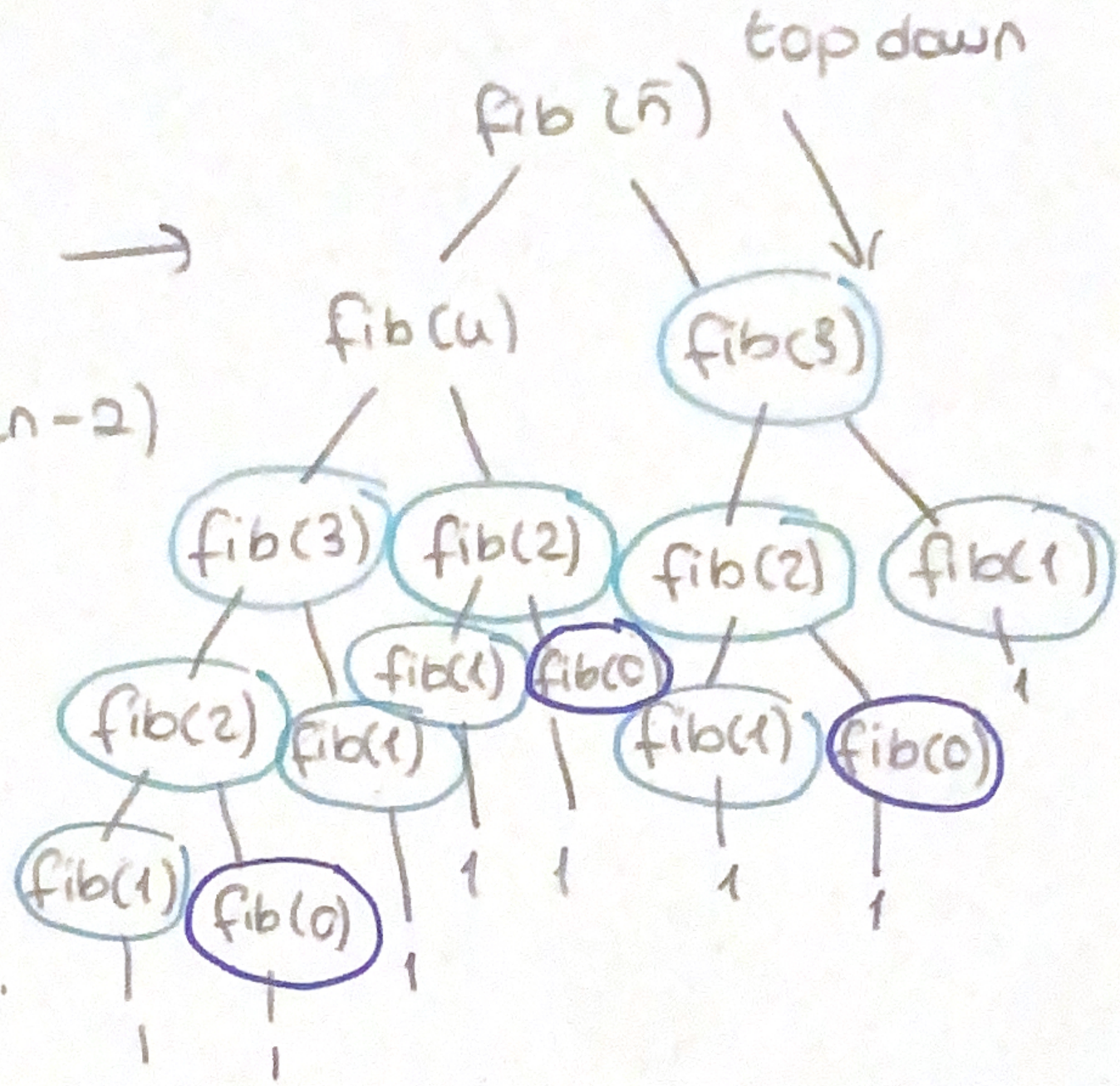
Dynamic Programming

Fibonacci

```
fib(n)
    if n=0 or n=1
        return 1
    else fib(n-1)+fib(n-2)
```

top down



Overlapping Subproblems?

Use memoization!
Store smaller subproblems.

0	1	2	..
1	1	2	..

do bottom up approach!

Fib(n)

F[0] = 1
F[1] = 1

$$F(n) = \begin{cases} 1 & \text{if } n=0 \\ & n=1 \\ F[n-1] + F[n-2] & \end{cases}$$

↓

Recurrence equation

```
F[0] = 1
F[1] = 1
i = 2
while i<n
    F[i] = F[i-1] + F[i-2]
return F[i]
```

Sadece n kez hesaplıdık (linear time)

↑ (not efficient with space)