

Algorithm Design & Analysis (CSE222)

Lecture-7

M = { }^N

Fib(n):

if n = 1

return 0

if n = 2

return 1

if M(n) is not empty.

M(n)

else

M(n) = Fib(n-1) + Fib(n-2)

return M(n)

0, 1, 1, 2, 3, 5, ...

Fib(n) → Fib(n-1) → Fib(n-2)

⋮

Fib(1) ← Fib(2)
↓ ↓
0 1

Running: O(n).

Imp: n balls (weighted, ordered), $w: B \rightarrow \mathbb{N}^+$

$$B = b_1, b_2, b_3, \dots, b_n$$

Goal: Find $B' \subseteq B$ s.t. for every $1 \leq i \leq n-1$, the $\{b_i, b_{i+1}\} \in B'$
 $\sum_{b \in B'} w(b)$ is maximized.

E.g.

b_1	b_2	b_3	b_4	b_5	b_6
1	5	8	5	4	7

$$\text{iter 1: } w(B') = \{8\}$$

$$\text{iter 2: } w(B') = \{8, 7\}$$

$$\text{iter 3: } w(B') = \{8, 7, 1\}$$