Glass Falling

n floors, m glass sheets.

a): Assume we throw a egg in k-th floor,

it broke, we don't care "$k+1 \ldots n$" floor, we find $f(n,m) = f(1 \ldots (k-1), m-1) + 1$

It not broke, we don't care "$1 \ldots k$" floors, : $f(n,m) = f(\underbrace{k+1 \ldots n}_{(n-k)}, m-1) + 1$

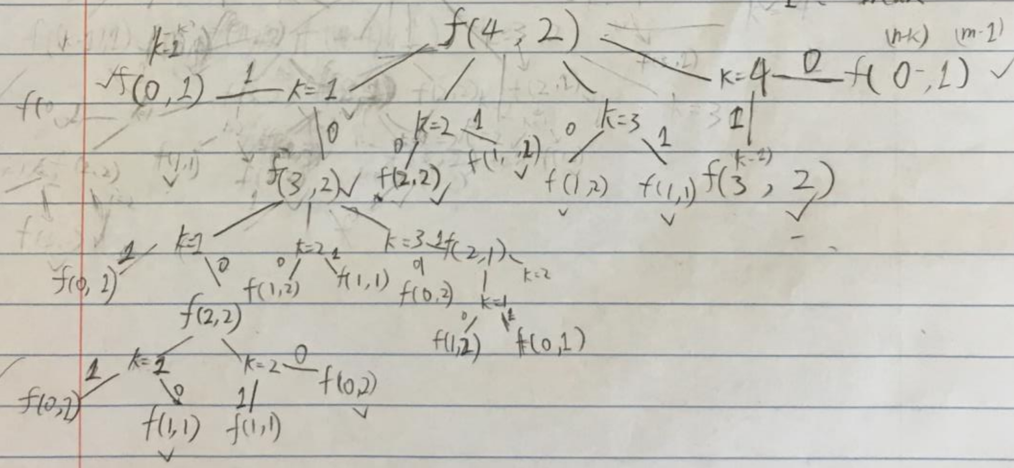
$$\begin{cases} f(n, m) = n, & m = 1 \\ f(n, m) = \min\{ \max(f((k-1), (m-1)), f((n-k), m)) + 1 \} \end{cases}$$

$1 \le k \le 4$

b:

0 = not break

1 = break



memoization by DP-table.

d: There are 16 "distinct" subproblem

e: There are m^n distinct subproblem.

f: I would using DP-Table to memoize

$f(n,1) = n$

$f(0,m) / (n,0) = 0$

col / row m \ floor n	0	1	2	3	4	... n
0	0	0	0	0	0	
1	0	1	2	3	4	
2	0	1	2	2	3	
3						
4						
⋮						
m						

$f(n,m) = \max\{ f(k-1, m-1),\ f(n-k, m) \}$ current/semi

$1 + \max[\ldots]$

2 (floor):

$\min \begin{cases} 1 + f(1,2) \begin{cases} \text{br } f(0,1)=0 \\ \text{not } f(1,2)=1 \end{cases} \max = 1,\ +1 = 2 \\ 1 + f(2,2) \begin{cases} \text{br } f(1,1) = 1 \\ \text{not } f(0,2) = 0 \end{cases} \max = 1+1 = 2 \end{cases} \Downarrow 2$

3 "k":

$\min \begin{cases} f(1,2) \begin{cases} \text{br } f(0,1) = 0 \\ \text{not } f(2,2) = 2 \end{cases} \max = 2+1 = 3 \\ f(2,2) \begin{cases} \text{br } f(2,1) = 1 \\ \text{not } f(1,2) = 1 \end{cases} \max\ 1+1 = 2 \\ f(3,2) \begin{cases} \text{br } f(2,1)\ 2 \\ \text{not } f(0,2)\ 0 \end{cases} 2+1 = 3 \end{cases} \Downarrow 2$

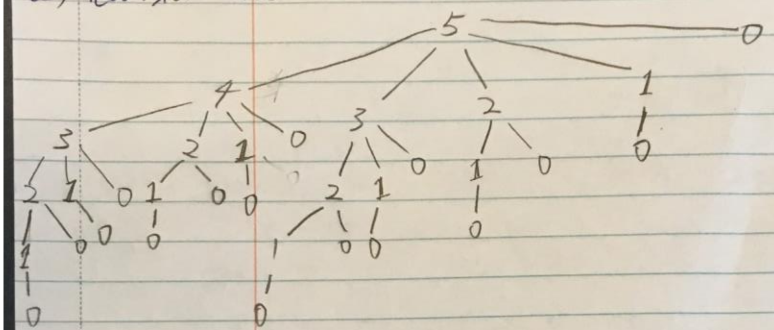
4:

$\min \begin{cases} f(1,2) \begin{cases} \text{br } f(0,1) = 0 \\ \text{not } f(3,2) = 3 \end{cases} 3+1 = 4. \\ f(2,2) \begin{cases} \text{br } f(1,1) = 1 \\ \text{not } f(2,2) = 2 \end{cases} 2+1 = 3 \\ f(3,2) \begin{cases} \text{br } f(2,1) = 2 \\ \text{not } f(1,2) = 1 \end{cases} 2+1 = 3 \\ f(4,2) \begin{cases} \text{br } f(3,1) = 3 \\ \text{not } f(0,2) = 0 \end{cases} \max \Rightarrow 3+1 = 4 \end{cases}$

$\boxed{\begin{matrix} 2 \\ 1 \end{matrix}}$

Rod Cutting

a) recursion tree:



b) 15.1-2:

Length i	1	2	3	4	5	6	7	8	9	10
Price p_i	1	5	8	9	10	17	17	20	24	30
	1	$5/2$	$8/3$	$9/4$	$10/5$	$17/6$	$17/7$	$20/8$	$24/9$	$30/10$
	1	2.5	2.6	2.25	2	2.8	2.4	2.5	2.6	3

$\frac{p_i}{i}$ → $n=4$

Sort: 1 $10/5$ $9/4$ $17/7$ $5/2$ $20/8$ $8/3$ $24/9$ $17/6$ $30/10$

$1 \le i \le n$

When $n = 4$ then us greedy Algorithm we will first pick $i=3$ which has the highest density. then $n-3 = 1$, result will be $\underset{8}{3}+\underset{1}{1} = 9$, but we know correct result is $\underset{5}{2}+\underset{5}{2} = 10$. so we can't use greedy Algorithm