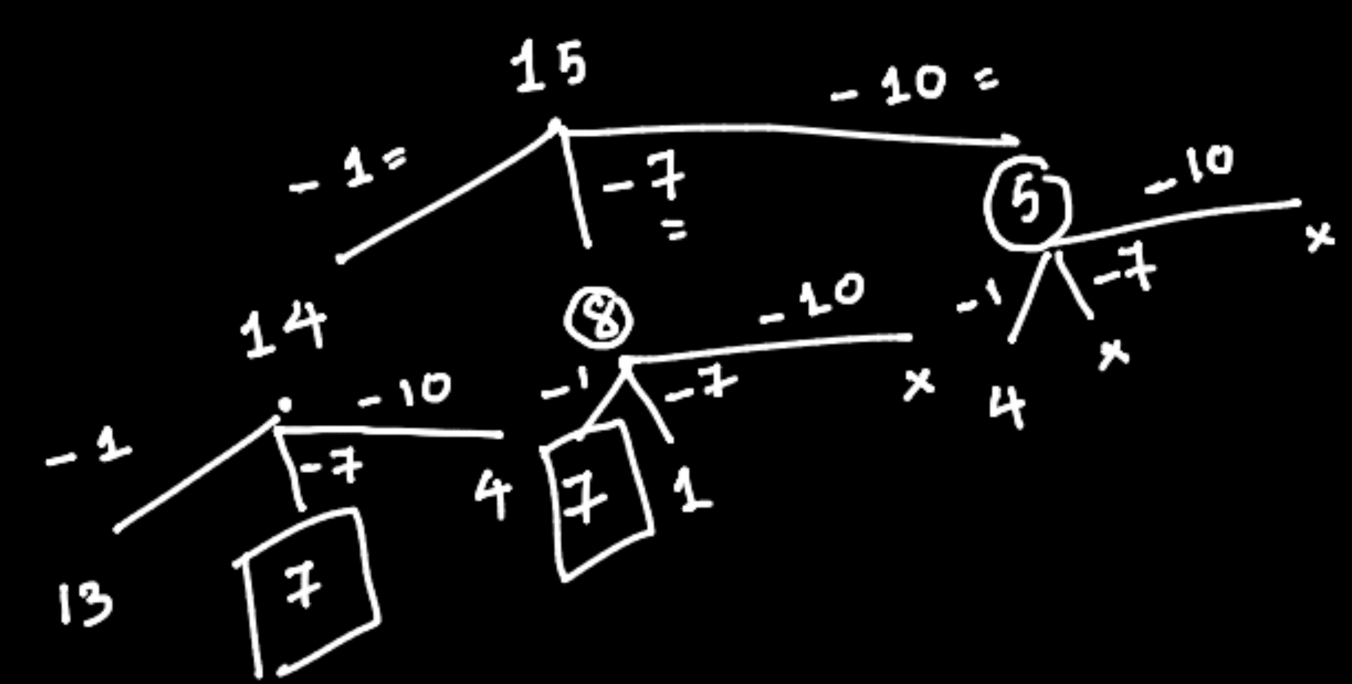
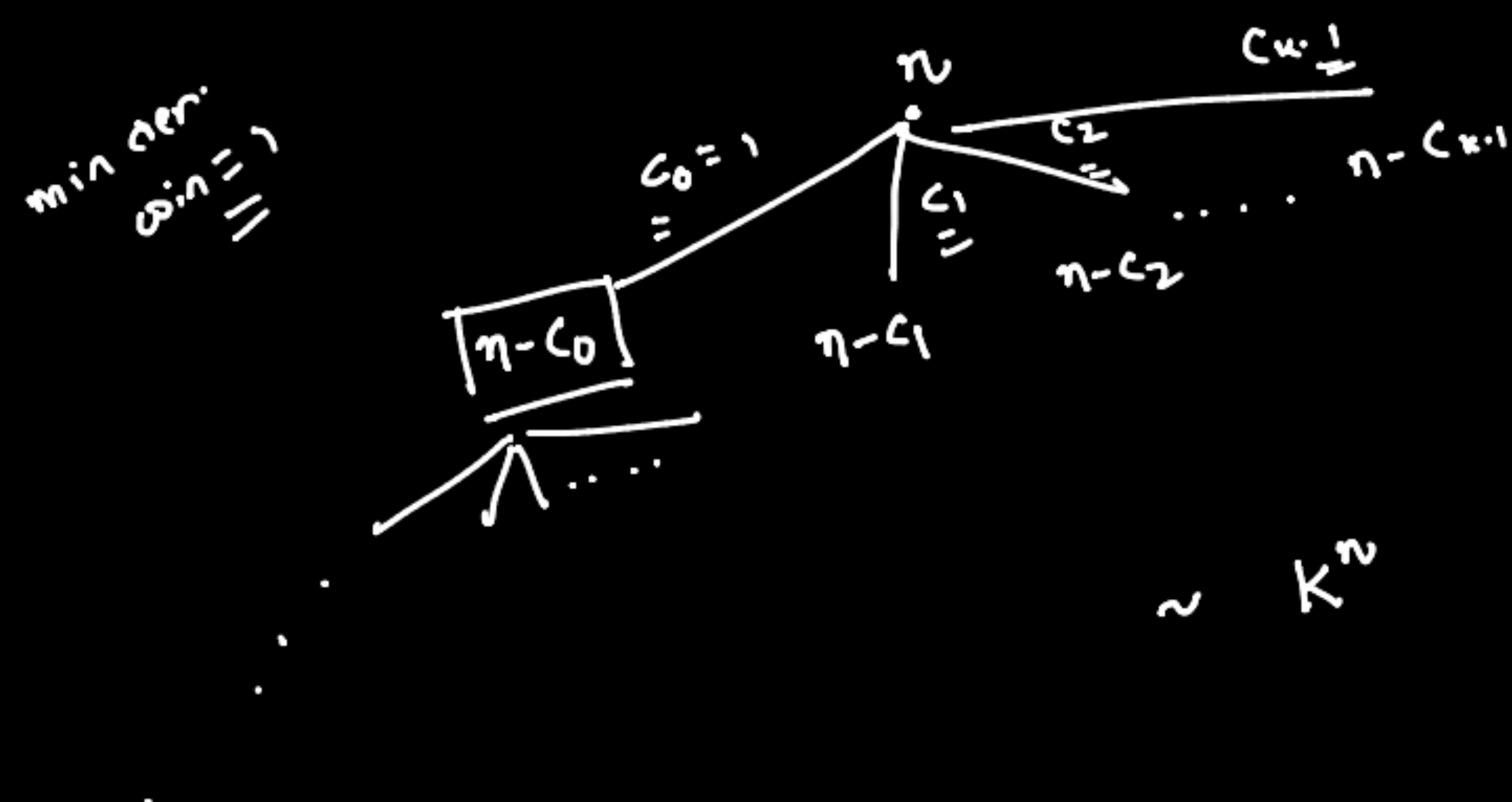
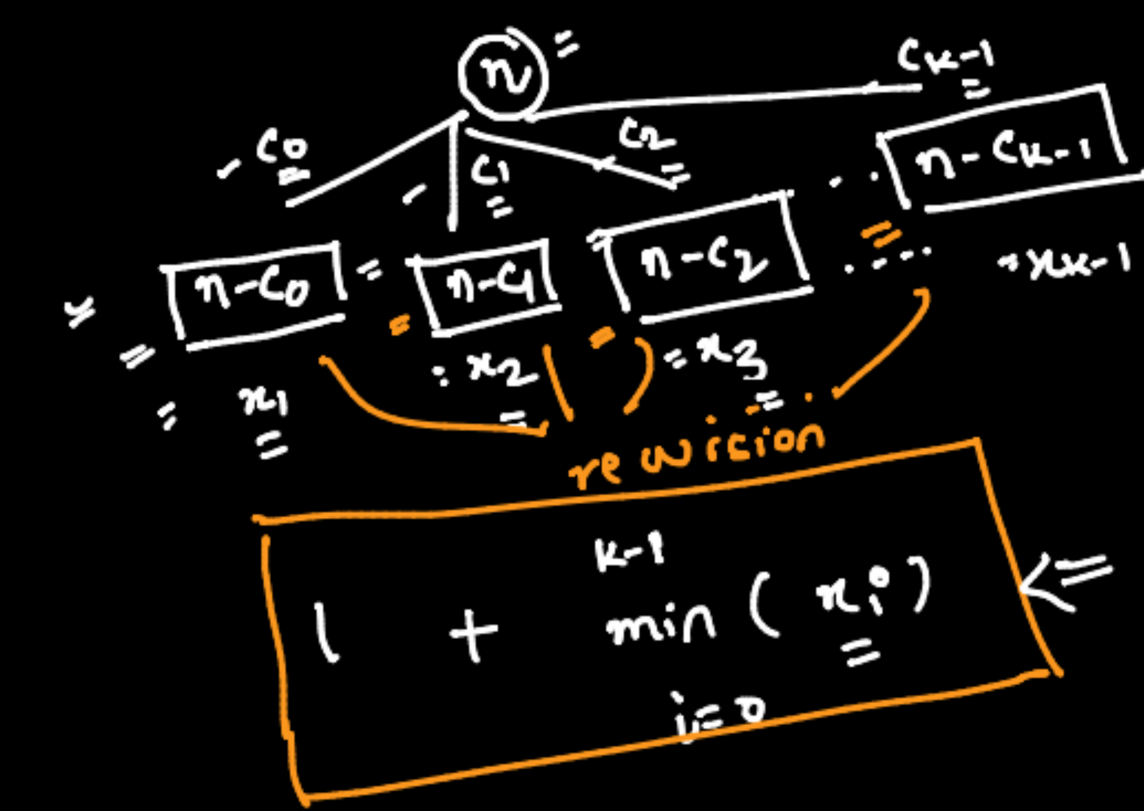
coins [ ] = [1, 7, 10] $n = 15$



'n'   'k'



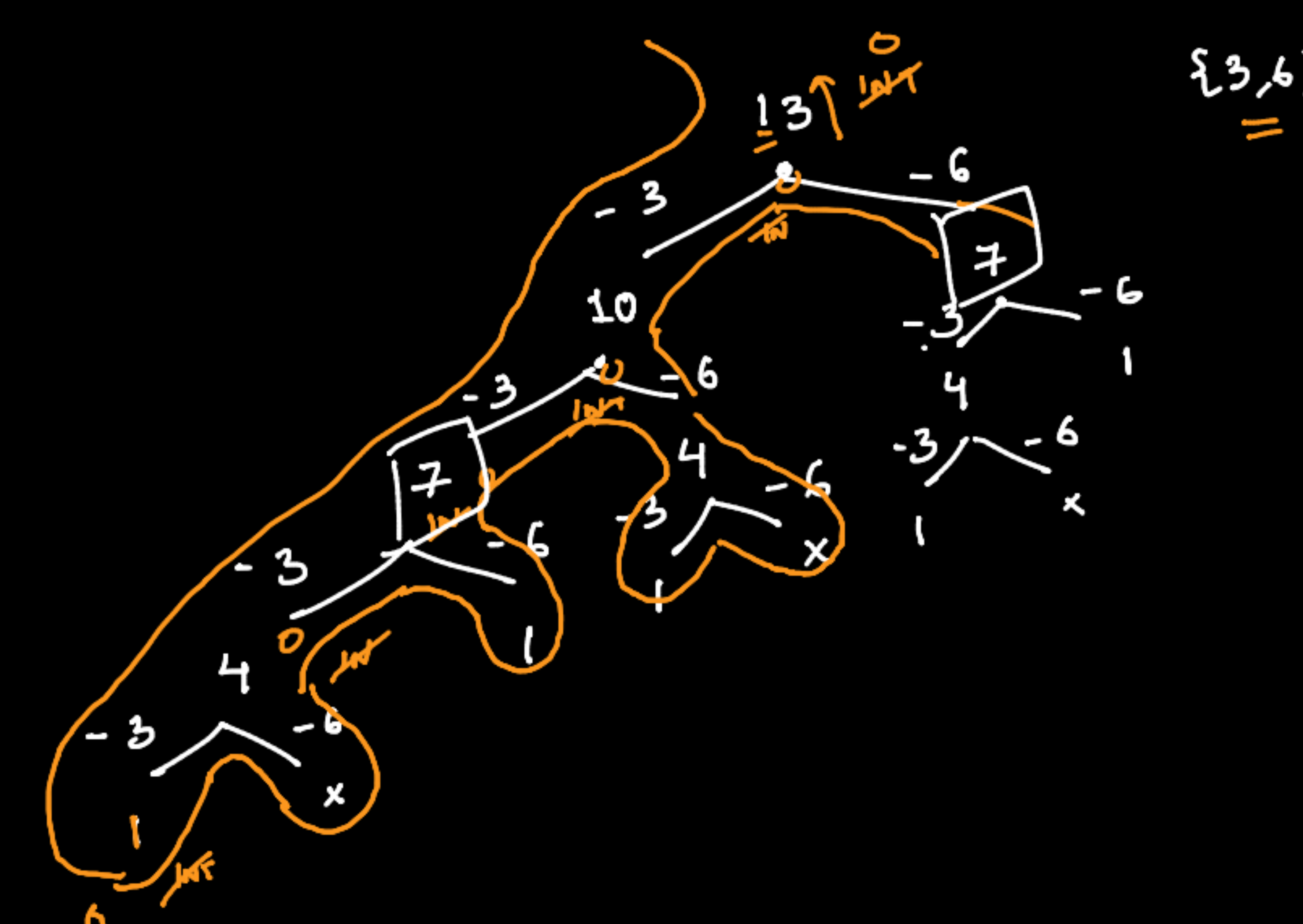
~ K^n



$f(n)$: min # coins req. to reduce n to 0

$$f(n) = 1 + \min_{i=0}^{k-1} \left(f(n-c_i) \right) \quad \text{] rec.}$$

$$f(0) = 0 \quad \text{] Base case}$$



{3, 6}

DP

1. int

2. $dp[0] = f(0)$
   $dp[1] = f(1)$ } $n+1$
   ⋮
   $dp[n] = f(n) \Leftarrow$ org.

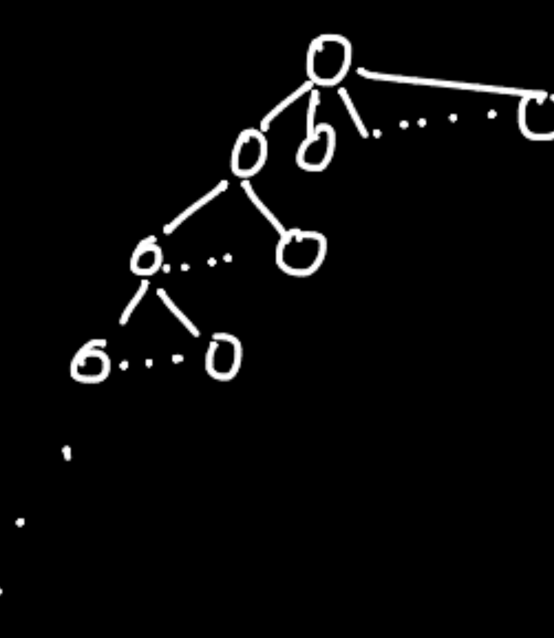
3. -1

Recursion

min coin = 1

(n/m)

$O(n/m)$



$\Rightarrow K^0$
$\Rightarrow K^1$
$\Rightarrow K^2$
⋮
$\Rightarrow K^{n/m}$, $z = n/m$

$(K^0 + K^1 + K^2 + \dots K^z)$

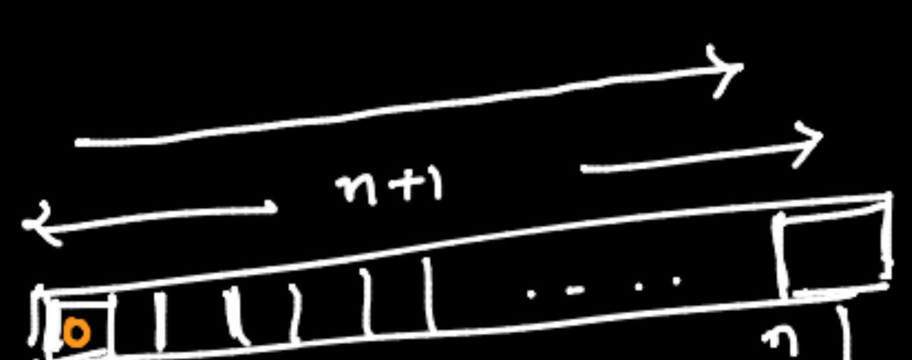
$a = K^0$
$r = K$

$\frac{K^{z+1} - 1}{K - 1} \sim K^z \sim K^{n/m}$

$\boxed{O(K^{n/m} \cdot K)}$ ⇝ recursion

Bottom-Up

$dp[\,]$



$n+1$

0 1 2 3 … n

0 ⇓ $f(0)$ ⇓ base ⇓ 0

n ⇓ $f(n)$ ⇓ org. problem.

$O(K^{n/m} \cdot K)$ time — K: # diff. coin. den.

$\Rightarrow O(n/m)$ space — n: amt., m: min coin deno.

✓ Bottom up
time $= O(nK)$
space $= O(n)$

Top Down
$n \Rightarrow \underbrace{f(0), f(1), \dots f(n)}_{n+1}$
time $= O(nK)$
Space $= O(n + n/m) \sim O(n)$

$P_0 \quad P_1 \quad P_2 \quad P_k \quad P_{n-1}$

$0 \quad 1 \quad 2 \quad k \quad n-1$

n wins

$P_1 * y \Rightarrow P_1 * 3$

$y = 3$

$P_k * y \Rightarrow P_k * 3$

$\Rightarrow [\, i,\ i+1, \ldots\ldots,\ j-1,\ j\,],\ y$

$P_i \quad P_j$

$(i, j),\ y$

$+ (P_i * y) \qquad + (P_j * y)$

$(i+1, j),\ y+1 \qquad (i, j-1),\ y+1$

$= [i+1 \ldots j]$ $\quad$ recursion $\quad$ $[i \ldots j-1]$

$x_1 \qquad\qquad x_2$

$$\boxed{\max\begin{pmatrix} (P_i * y + x_1) \\ (P_j * y + x_2) \end{pmatrix}}$$

idx $\quad 0 \quad 1 \quad 2 \quad 3 \quad 4$

$\{\, 2,\ 3,\ 5,\ 1,\ 4 \,\},\ y = 1$

$i \longleftarrow n = 5 \longrightarrow j$

$\Rightarrow +2,\ i{+}{+} \qquad +4,\ j{-}{-}$

$= \{\, 3,\ 5,\ 1,\ 4 \,\},\ y = 2 \qquad \{\, 2,\ 3, 5, 1 \,\},\ y=2$

$+3 \cdot 2,\ i{+}{+} \qquad +4 \cdot 2,\ j{-}{-}$

$\{\, 5,\ 1,\ 4 \,\} \qquad \{\, 3,\ 5,\ 1 \,\}$

$$f(i, j) = \max\begin{pmatrix} P_i * y + f(i+1, j) \\ P_j * y + f(i, j-1) \end{pmatrix}$$

$x_1 \quad x_2$

$[i \ldots j]$

$[i+1 \ldots j]$

$[i \ldots j-1]$

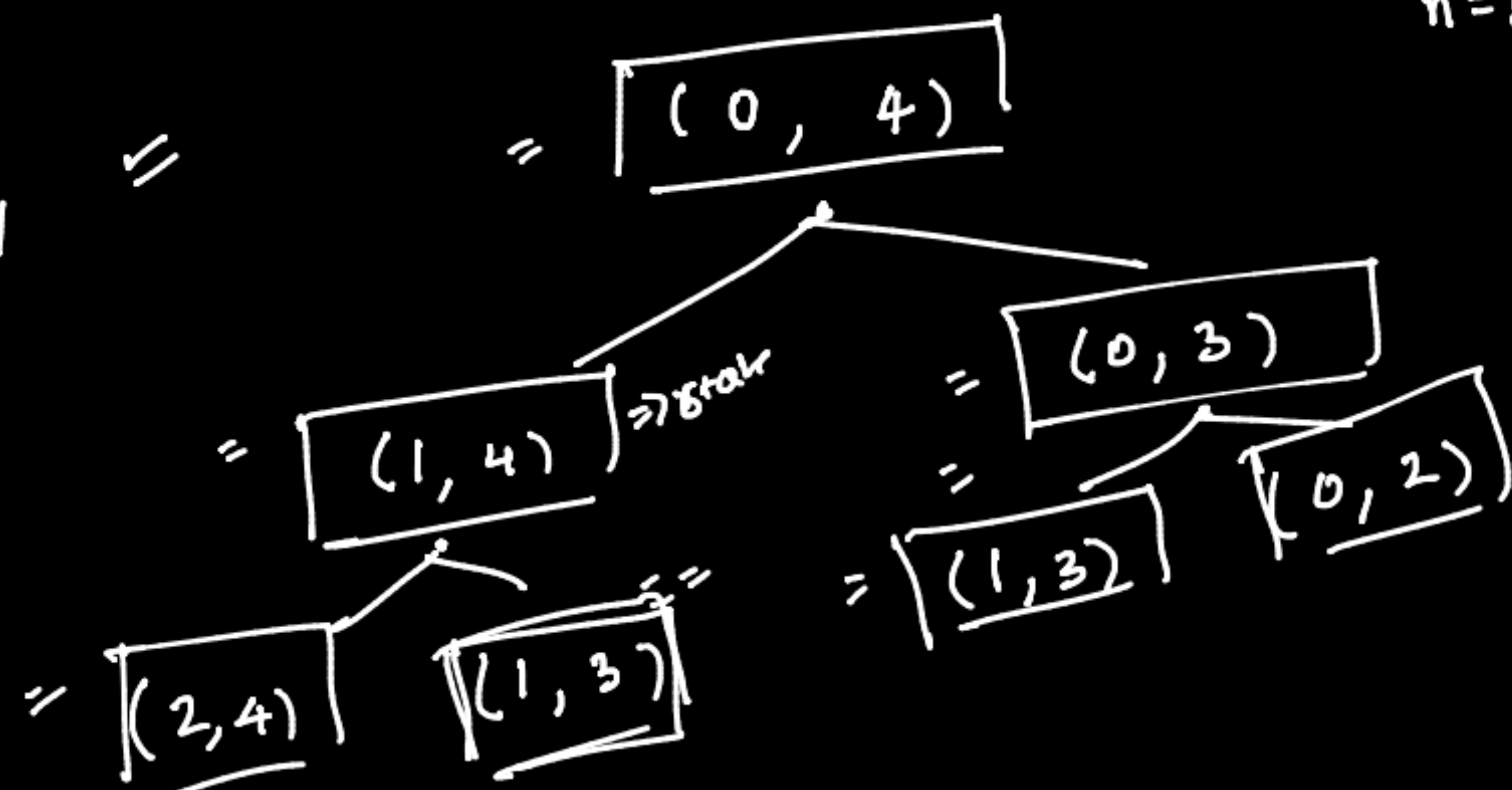
Base case

$$\boxed{y \cdot P_i \text{ or } y \cdot P_j \quad i = j}$$

$0 \qquad i > j$

$n = 5 \quad i = 0 \quad j = 4$

DP



(i, j)

$n * n \Leftarrow n^2$

$j = n - 1$

$i = 0 \rightarrow \underbrace{0 \text{ to } n-1}_{n}$

$\underbrace{n-1 \text{ to } 0}_{n}$

$n - (j - i) = 5 - (0 - 0) = 5$

dp[ ][ ]

$2 * 5$

$i > j = 0$

$i == j$

$(i = j)$

0 1 2 3 4

→ final ans

$n = 5$

dp(0)(4)

= f(0)(4)

[0 .... 4]

5×5

$\hookrightarrow i > j$

$f(i, j) = \max \begin{cases} p_i \cdot y + f(i+1, j) \\ p_j \cdot y + f(i, j-1) \end{cases}$

$\boxed{dp(i)(j)} = \max \begin{cases} p_i \cdot y + dp(i+1)(j) \\ p_j \cdot y + dp(i)(j-1) \end{cases}$

$(i, j) \rightarrow (i+1)(j)$, $(i)(j-1)$

dp(0)(0) = f(0, 0) ⇒ [0 .... 0] [0]

wines = [2, 3, 5, 1, 4]

dp(i)(j)

dp(i)(i)

$= n * \text{wines}(i)$

$\hookrightarrow n - (j - i)$

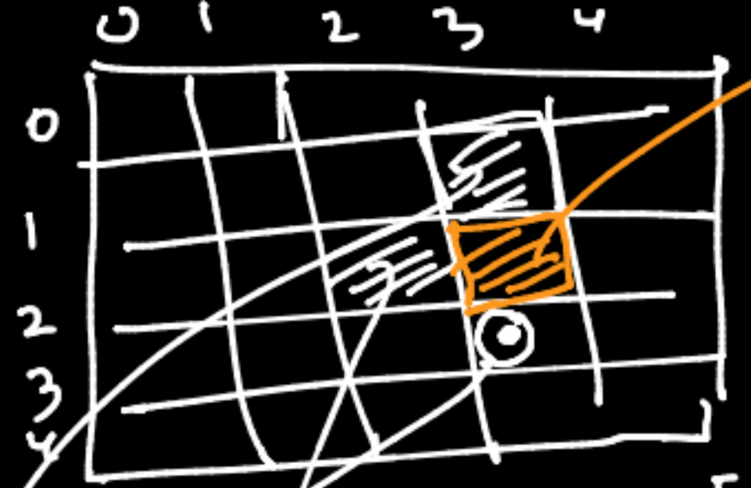
dp(1)(1) = f(1, 1) ⇒ [1 .... 1] ⇒ [1]

$3 * 5 = 15$ ....

$(i, j) \Rightarrow n - (j - i)$

2D-DP

0 1 2 3 4



5×5

$i = 2, \ j = 3$

$n = 5$

$i = 0$ to 4

$j = 4$ to 0

$5 - (3 - 2) = 4^{th}$

$n - (j - i)$

dp[2][2] ⇒ f(2, 2)

dp[1][3] = f(1, 3)

[1 .... 3]

$5 - (j - i)$

$5 - (3 - 3) = 5$

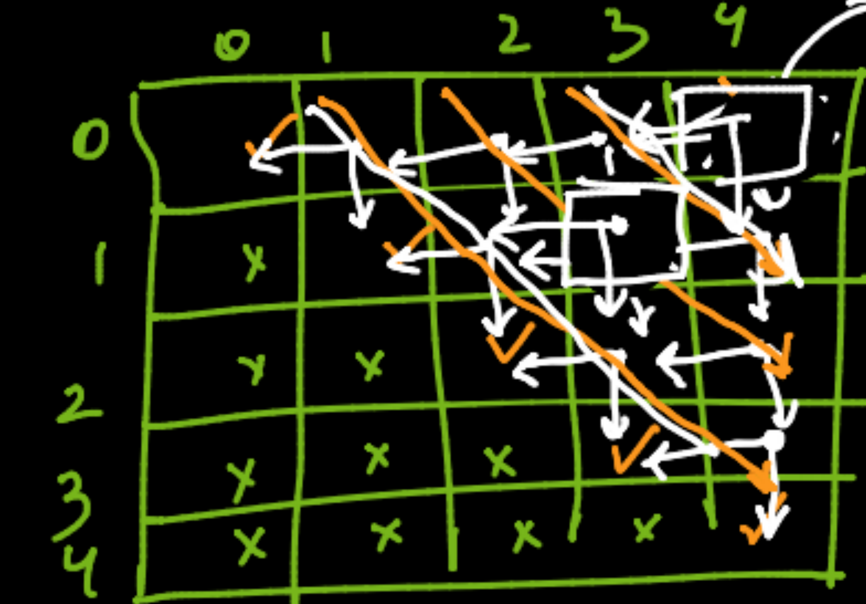
$(3, 3) \equiv 5^{th}$

dp[i][j] = f(i, j) ⇒ [i .... j]

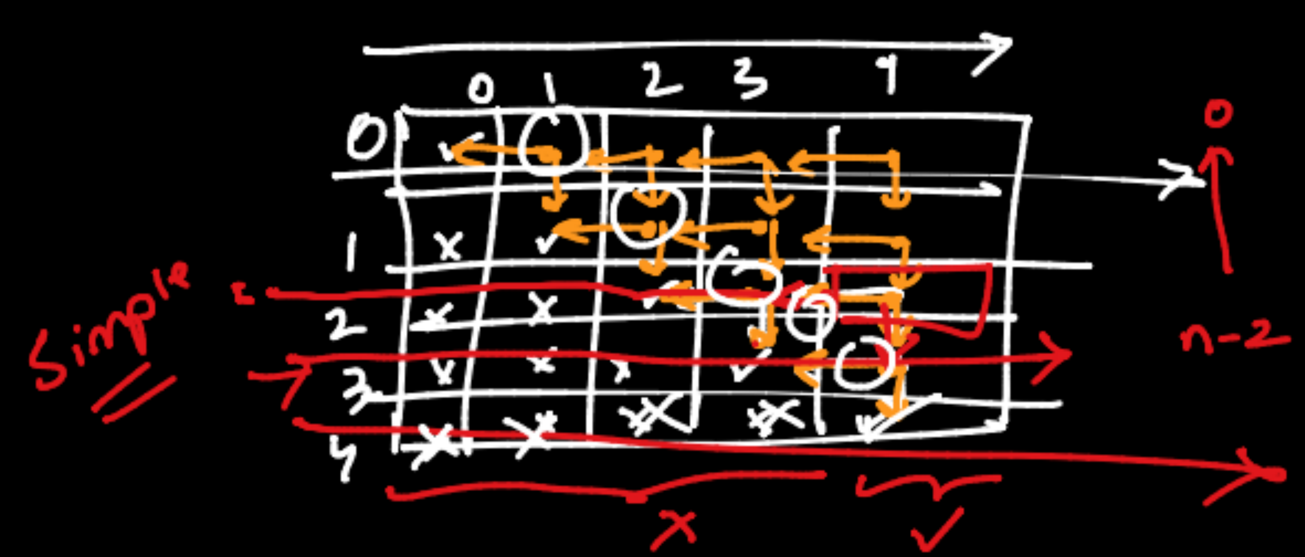
dp[2][3] = f(2, 3) ⇒ [2 .... 3]

fill diagonally (row)

0 1 2 3 4



→ final ans.



↳ row-wise in BU

↳ each row left to right

(n-2)th to 0th

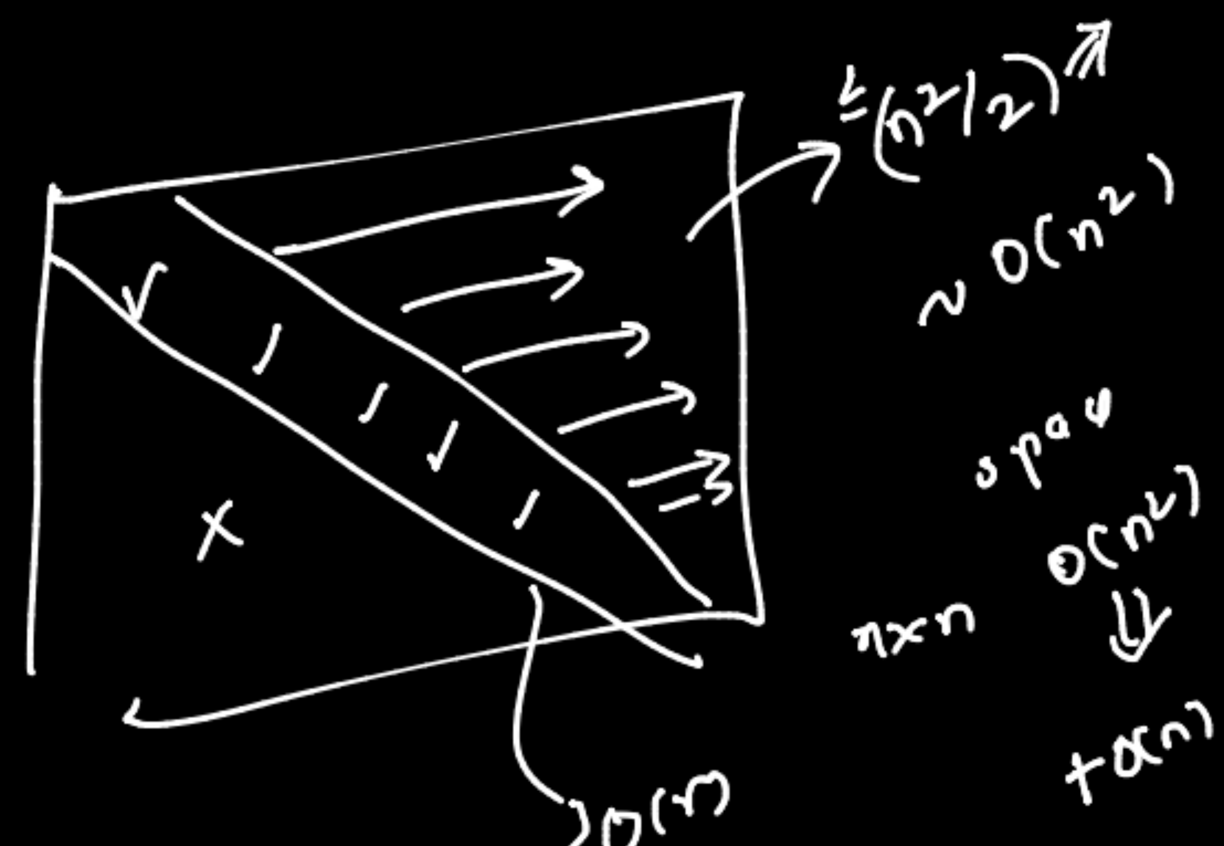
wines

1. wines → try to fill array diagonally

friday

2. Draw the recursion for top-down

3. Fill table by doing dry run using bottom



wines [ ] = [\$1 \$2 \$3] y = 1

1·1 + 2·2 + 3·3 = \$14

[2 1 3]

2·1 + 1·2 + 3·3

= 2 + 2 + 9 \$13

$\cancel{2}$ $\cancel{3}$ 5 $\cancel{4}$ $\cancel{1}$

$2*1 + 3*2 + 4*3 + 1*4 + 5*5$

$= 2 + 6 + 12 + 4 + 25$

$= 49 \Rightarrow 50$

$\cancel{2}$ $\boxed{3}$ 5 1 $\boxed{4}$

$2*1 + 4*2 + 1*3 + 3*4 + 5*5$

$= 2 + 8 + 3 + 12 + 25$

$= 50$

Greedy

$(i,j) \leftarrow$ optimization

$(i+1,j)$ $(i,j-1)$

Greedy

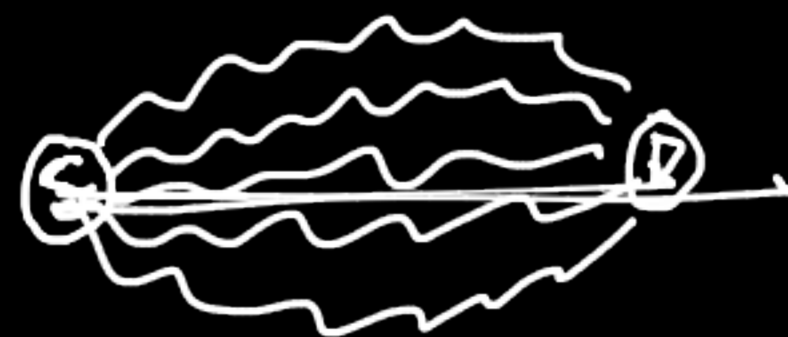
greedy

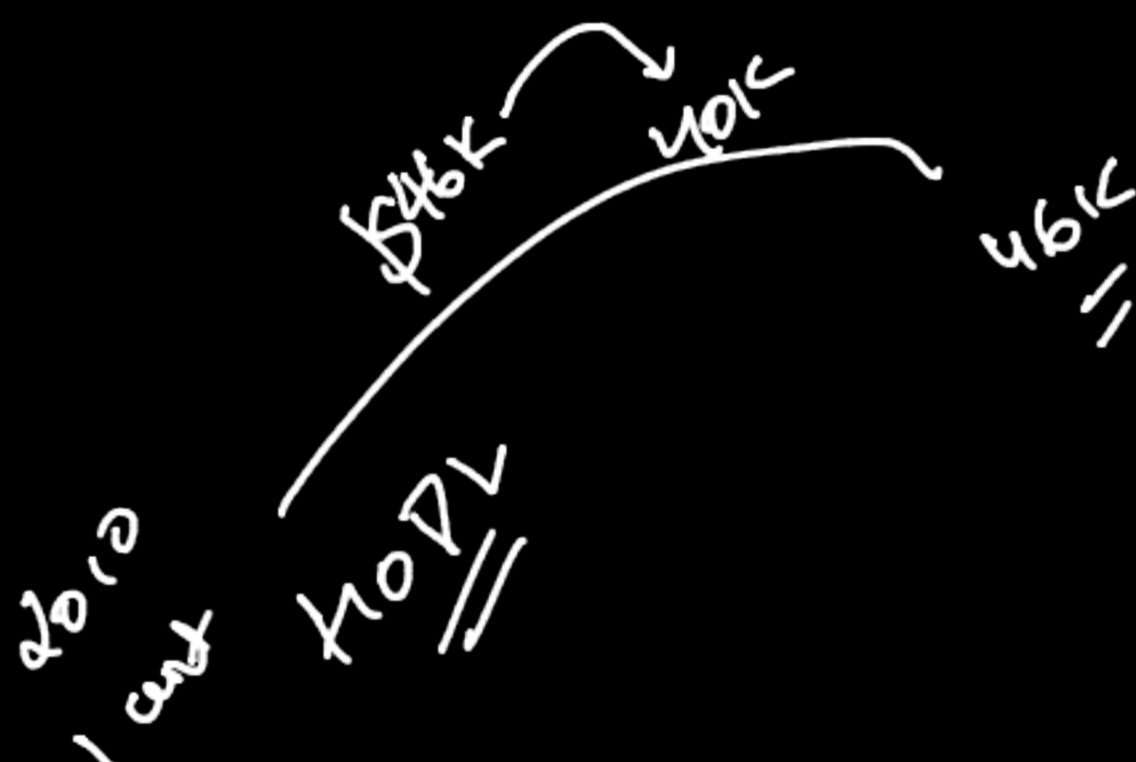
Adv: fast, intuitive

Dis: Not works always

$\Rightarrow$ does it guarantee global optimum

$\Rightarrow$ No mathematical


S
D


$46k
40k
46k
2010
1 cent
HODL