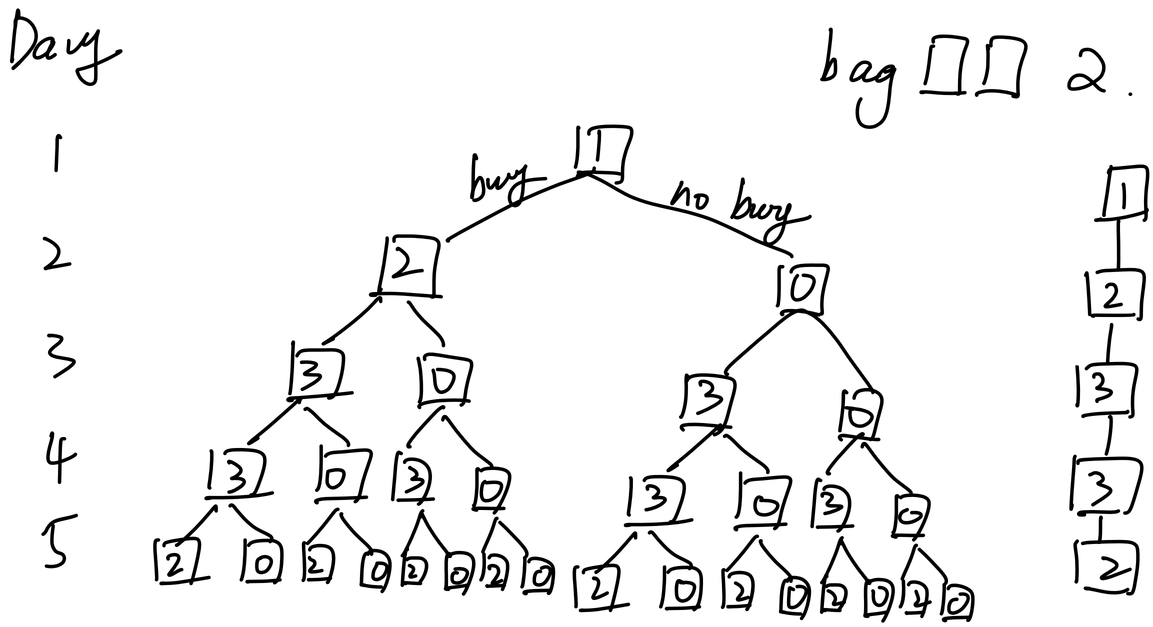
Q2. Supply problems on the hike

Example 1 $n=5$ $k=2$ data = 1 2 3 3 2

output 9

bag □□ 2.



① define a DP state : minimum cost after finishing each day

at the end of day i, with j units of food left

② transition

compute DP[i][j]

At the stare of day i

we have m leftover unit

buy y unit at cost data[i-1] per unit

After buying, total unit m+y

consume 1 this day, $\underbrace{(m+y-1)}_{\text{define as } j=m+y-1}$ unit left

③ Constraints

(1) carrying capacity

After buying $m+y = m+(j-m+1) = j+1 \le k$.

So $j \le k-1$, j from $[0, k-1]$

(2) $y = j-m+1 \ge 0$

So $j \ge m-1$, $j \ge 0$ $m \ge 0$

(3) we know the previous day

$DP[i-1][m]$

$cost = g \times data[i-1]$

④ transition formula is

$DP[i][j] = \min(DP[i-1][m] + (j-m+1) \times data[i-1])$

over all m

where $j \leq k-1$

$j \geq m-1 \rightarrow j-m+1 \geq 0$

⑤ Base Case

On day 1, we start with 0 uints, end with j units

$DP[0][0] = 0$

$DP[1][j] = DP[0][0] + (j-0+1) \times data[0]$

$= (j+1) \times data[0]$

where $j+1 \leq k$

$$result = \min(DP[n][j]) \text{ for } j \text{ in } [0, k-1]$$

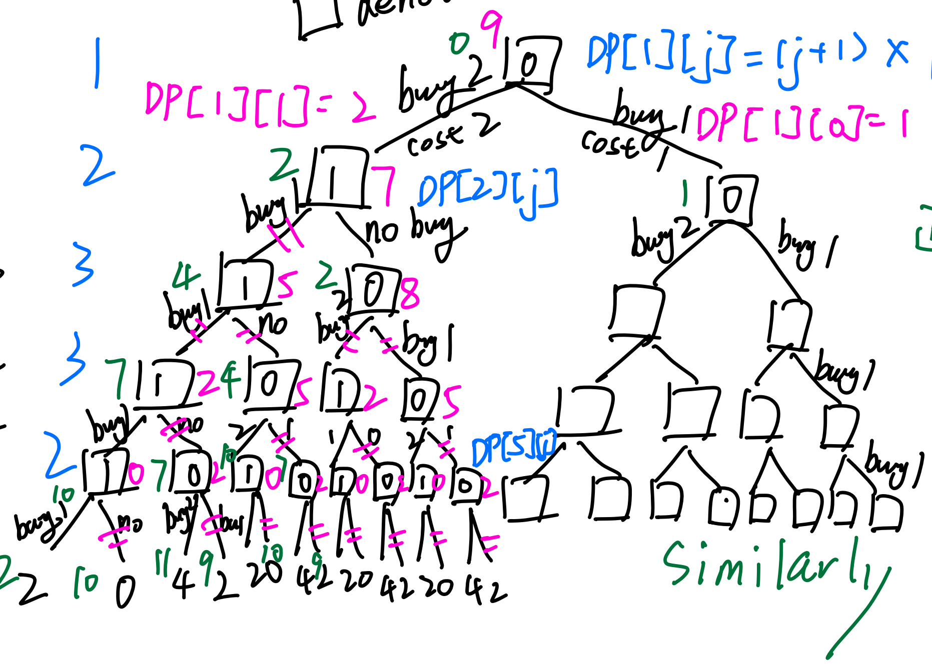
Example 1

$DP[i][j]$ at the end of day i
j unit left
buy y unit

$DP[0][0] = 0$

☐ denote start unit

Day	cost
1	1
2	2
3	3
4	3
5	2



constrain $m-1 \le j \le k-1$ m 一天开始数
k 购买数

↙固定的

Day 1 $-1 \le j \le 1$

So 可以剩 0 个 or 1 个

$j = 0, 1$

Code Implementation

① Define INF initization.

DP = int[n+1][k+1]

② DP[0][0] = 0

③ for loop

天 i \ 剩 j	0	1	2
0	0	∞	∞
1	buy 1 1	buy 2 2	∞
2	m 0 min cost3 2 cost 2	4	∞
3	4	7	∞
4	7	10	∞
5	9	11	∞

constrain

$m-1 \le j \le k-1$

if $j+1 > k$
continue

if $y = j - m + 1 < 0$
continue

if DP == INF
continue

cost = y × data[i-1]

newcost = DP[i-1][m]
+ costToday

if newcost < DP[i][j]

=

④ find the optimal answer

j   DP[5][j] < ans

0   return ans

1

2