

$dp = 1$

a_1	a_2	a_3	a_n
-------	-------	-------	---	---	---	---	-------

n stone

$$\hookrightarrow \underbrace{f(i)} = \min \left(\underbrace{|a_{i-1} - a_i| + f(i-1)}_{\text{one choice}}, \underbrace{|a_{i-2} - a_i| + f(i-2)}_{\text{second choice}} \right)$$

which returns

the minimum
cost to reach
from stone 1 to

stone i

$\left[\underline{\underline{dp[i]}} \right] \rightarrow$

Q $\Rightarrow 2$

$$\underline{f(i)} = \min (|a_i - a_{i-j}| + f(i-j))$$

↓
min cost
to reach at i
from first

slow

$$\forall j \in \underline{(i-k, i-1)}$$

Q. 3

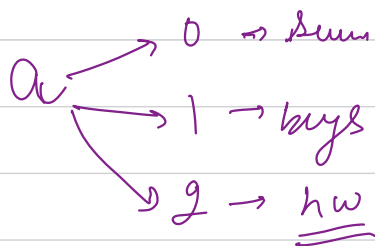
$$f(a, i) =$$



$f(i)$ ← max

max happiness

till the i^{th} day



0th activity on day i

$$h_{i0} + \max(f(1, i-1), f(2, i-1))$$

$$h_{i1} + \max(f(0, i-1), f(2, i-1))$$

$$h_{i2} + \max(f(0, i-1), f(1, i-1))$$

dimension of dp → 2d

Q.4

$i_1 \quad i_2 \quad i_3 \quad \dots \quad i_j \quad \dots \quad i_n$

$\{1, 2, 3\}$

\downarrow
 2^n ways

$\left\{ \begin{array}{l} \{1\}, \{3\}, \{1, 3\}, \{1, 2, 3\}, \\ \{2\}, \{1, 2\}, \{2, 3\}, \underline{\underline{\{1, 2, 3\}}} \end{array} \right\}$

target wt \rightarrow w

$f(i, Tw)$

↓
max profit/value
we can get till
the i^{th} item with
target wt $\rightarrow Tw$

=

$f(i-1, Tw)$

if
 $wt[i] > Tw$
(not pick)

→ not pick
 $\max(f(i-1, Tw),$

$val[i] + f(i-1, Tw - wt[i])$)

↓
pick

if
 $wt[i] \leq Tw$
pick
or
not pick)

$dp[i][j]$

=

if $(wt[i] > j)$ {
 $dp[i-1][j]$

} else {

$\max(dp[i-1][j], val[i] +$
 $dp[i-1][j - wt[i]])$

}

↙
max profit till
 i^{th} element with
J as the wt of
knapsack

2d array

→

make 2 1d arrays

$n \times w$

$O(w)$

$w=8$
now if this is picked

tw

$O(n \times w)$

values

	0	1	2	3	4	5	6	7	8
3, 30	0	0	0	0	0	0	0	0	0
4, 50	0	0	0	30	30	30	30	30	30
5, 60	0	0	0	30	50	50	50	80	80
	0	0	0	30	50	60	60	60	90

90

$dp[i-1][j]$ not pick

$dp[i][j] \rightarrow$ max profit

$val[i]$ + $dp[i-1][j-w+1]$ pick
 30 + 0

50 + $dp[1][2-4]$
 50 + 30 = 80

Qns

$f(i, v)$

that gives the
min wt reqd
so that the value
of selected items is
 v till the i^{th}
element

$$= \left\{ \begin{array}{l} f(i-1, v) \\ \min (f(i-1, v), \\ \text{wt}[i] + f(i-1, v - \text{val}[i])) \end{array} \right.$$

→ not pick

$\min (f(i-1, v),$

$\text{wt}[i] + f(i-1, v - \text{val}[i]))$

else

↓
pick

$$10^3 \times 10^2 \rightarrow \frac{10^3}{10^2} \rightarrow 10^1$$

if $\text{val}[i] > v$

Ans \rightarrow for last row \rightarrow $i = n-1$

max value of the column j such that

$$\underline{dp[n-1][j]} \leq \underline{\text{Knapsack wt}}$$

max $f \rightarrow$ ans

S, 16

2

3

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0														
0	∞	∞	3	6	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
0	∞	∞	3	∞	4	∞	∞	7	6	6	∞	∞	∞	∞
0	∞	∞	3	∞	4	5	∞	∞	8	∞	9	6	∞	12

full array is filled
cell $+\infty$

$\infty, \underline{\underline{4+6}}$

$$\frac{\infty}{4} + \infty$$

→ 0

00, 473

$$\omega = f$$