(IV) Knapsack Problem

↳ select some items from given items such that profit obtained is maximum.

(i) 0/1 Knapsack problem → Either pick completely or don't pick. (item not divisible).

↓

Solved with dynamic programming.

Eg: weights = {3, 4, 6, 5}
values = {2, 3, 1, 4}

W = 8 kg
n = 4.

pi	wi	i \ w	0	1	2	3	4	5	6	7	8
		0	0	0	0	0	0	0	0	0	0
2	3	1	0	0	0	2	2	2	2	2	2
3	4	2	0	0	0	2	max(3+0, 2) = 3	max(3+0, 2) = 3	max(3+0, 2) = 3	(3+2, 2) = 5	(3+3, 2) = 5
4	5	3	0	0	0	2	3	4+0, 3 = 4	4+0, 3 = 4	4+0, 5 = 5	4+2, 5 = 6
1	6	4	0	0	0	2	3	4	1+0, 4 = 4	1+0, 5 = 5	1+0, 6 = 6 → max profit

for $w_i > w \Rightarrow$ copy values from above.

else, $\max(p_i + (w_j - w_i)p, p_{i-1})$

General formula :-

$$m[i, w] = \max(m[i-1, w], m[i-1, w - w[i]] + p[i])$$

Steps to know which items are selected :-

- move the ptr. to the max profit block
- shift ptr. upward.

if (profit changes)
select that item
calculate the remaining wt. and move to that wt. column and repeat the same.

∴ Here, $x_i = \{1, 0, 0, 1\}$
(3 4 6 5)