X

06 April 2025 22:25

In the 0-1 Knapsack Problem, we are given a Knapsack or a Bag that can hold weight up to a certain value. We have various items that have different weights and values associated with them. Now we have to fill the knapsack in such a way so that the sum of the total weights of the filled items does not exceed the maximum capacity of the knapsack and the sum of the values of the filled items is maximum.

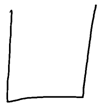
Given a Knapsack with maximum weight limit as W and two arrays value[] and weight[]. You have to fill the knapsack in such a way so that the total weight of the filled items is less than or equal to W and the sum of the values of the filled items is maximum. value[i] and weight[i] will store the value and weight associated with ith item. You can not partially fill an item in the knapsack.

0 – Don't take 1 – Take.

Object	ob1	ob2	ob3
Weight	2	4	8
Profit	20	25	60

1 kg Price 20/2 = 10 6.25 7.5

Knapsack capacity = 12 kg.



Pumpkin (ands) → 2 kg.

Fractional Knapsack.

Object	wt	Profit	Remaining wt
1	2	20	12−2=10
3	8	60	10−8=2.

80 → by Greedy.

By observation –

obj 2 & obj 3

Price → 25 + 60 = 85.

wt → 4 + 8 = 12

0–1 Knapsack.

Object	ob1	ob2	ob3
Weight	2	4	8
Profit	20	25	60.

Knapsack Capacity, W = 12 kg

Knapsack capacity (W)

No of items (i)

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	20											
2	0												
3	0												

(1,4)

(1,1)
= v[0][1]
= 0
v[0][1−2]
= 0

(1,2)
v[0][2]
= 0

v[i−1] [W−weight[i]] + profit[i].
= v[0] [2−2] + 20
= 0 + 20
= 20.

$$v[i,W] = \text{Max}\big(v[i-1, W],\ v[i-1, W - \text{weight}[i]] + \text{profit}[i]\big) \quad > 0.$$

obj → Take, Don't Take.

Take → $W >= \text{weight}[i]$ → Take.

$W < \text{weight}[i]$ → Don't Take

i\w	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	20	20	20	20	20	20	20	20	20	20	20
2	0	0	20	20	25	25	45	45	45	45	45	45	45
3	0	0	20	20	25	25	45	45	60	60	65	65	85