Knapsack no repetition in disguise

capacity = n

Table of len(group) \* n

group = weights, profits = values

But it’s not maximization, it’s meeting a goal = minProfit

So each row try to meet the goal using profit(i) and profit(1...(i-1))

If a row meets minProfit then +1 to num\_schemes

T(i,j) = total profit using profit(1...i) but stop at capacity n, fill knapsack with crimes until n is reached, check if minProfit achieved.

Recurrence:

For i = 2 to m where m is len(profit):

j = 0

While j <= n:

if j >= group(i):

T(i,j) = max{T(i-1,j), T(i, (j – group(i)) + profit(i))}

if T(i,j) >= minProfit:

num\_schemes +=1

break

else:

T(i,j) = T(i-1,j)

j+=1

return num\_schemes