2) dp[i][j] = minimum total weigth to achieve a total value j by using the first i items.

Base cases

dp[i][0] = 0 $\forall i$, dp[0][j] = $+\infty$ $\forall j \geq 1$

Recursion

for i = 1,...,n

  for j = 0,...,V

    if v[i-1] ≤ j:

      dp[i][j] = min (dp[i-1][j], dp[i-1][j- v[i-1]] + a[i-1])

    (v[i-1]: value, a[i-1]: weigth)

    else:

      dp[i][j] = dp[i-1][j]

Look at the largest value of j such that dp[n][j] ≤ total capacity

Running time: O(n·V)