Knapsack problem—traditional dynamic programming

for $b := 0$ to B do $K[0, b] := 0$

for $i := 1$ to n

  for $b := 1$ to B

    if $w_i \leq b$ and $K[i-1, b-w_i] + c_i > K[i-1, b]$

      $K[i, b] := K[i-1, b-w_i] + c_i$

    else

      $K[i, b] := K[i-1, b]$

return $K[n, B]$

Running time: $O(nB)$

Knapsack problem—alternative dynamic programming

for $c := 0$ to C do $F[0, c] := \infty$
$F[0, 0] := 0$

for $i := 1$ to n
  for $c := 0$ to C
    if $c_i > c$ and $F[i-1, 0] + w_i \leq F[i-1, c]$
      $F[i, c] := F[i-1, 0] + w_i$
    else if $c_i \leq c$ and $F[i-1, c-c_i] + w_i \leq F[i-1, c]$
      $F[i, c] := F[i-1, c-c_i] + w_i$
    else
      $F[i, c] := F[i-1, c]$

return maximum c, for each $F[n, c] \leq B$

Running time: $O(n^2 \max\{c_i\})$

Definition: Approximation algorithm $A(x, \varepsilon)$ is a **polynomial-time approaximation scheme (PTAS)** if for any constant $\varepsilon > 0$

$$A(x, \varepsilon) \leq (1 + \varepsilon)OPT(x) \quad \text{(min. problem)}$$

resp. $A(x, \varepsilon) \geq (1 - \varepsilon)OPT(x)$ (max. problem)

and its running time is polynomial with respect to $|x|$.

It is a **fully-polynomial-time approximation scheme (FPTAS)** if its running time is also polynomial with respect to $1/\varepsilon$.