Complete Knapsack

完全背包

问题：

你面前摆放着n种珠宝，每种都有无穷多个，已知珠宝s_i的价值是v_i，重量是w_i。给你一个背包，你可以自由挑选珠宝装到背包中，但背包可以装载的最大重量为t。求背包能够装载珠宝的最大价值v。

解法：

设$f(i,j)$为背包中放入前i件物品，重量不大于j的最大价值，其中$i \in [1,n]$，$j \in [0,t]$。有如下状态转移方程：

$$f(i,j) = \begin{cases} 0 & \text{（初始化）} i = 0 \\ max(f(i-1,j), f(i-1,j-k \times w_i) + k \times v_i) & i > 0 \text{ 且} j > 0 \text{ 且} k \geq 0\text{，} j \geq k \times w_i \end{cases}$$

(1) 用数组中的下标 0 来存储初始的固定值，背包中没有放入任何珠宝时，$f(0,j) = 0$；

(2) 对于第i件珠宝s_i，背包的剩余重量（还能装载的重量）为W，可以装进k个该珠宝（其中$k \geq 0$，且$W \geq k \times w_i$），那么背包的价值增大$k \times v_i$，剩余重量减小$k \times w_i$，即$f(i,j) = f(i-1,j-k \times w_i) + k \times v_i$；若不装入背包，则一切维持不变，即$f(i,j) = f(i-1,j)$。选择这两种情形中的最大值；

$f(n,t)$即为n个珠宝中重量不超过t的最大价值。该算法的时间复杂度是$O(n \times t^2)$，因为状态转移方程中的参数k的规模与背包最大重量t线性相关。