
Sluggish Knapsack**X59240_en**

We consider the classical Knapsack problem in its 0/1 variant; this means that each object is either taken fully into the knapsack, exactly once, or left fully out.

More precisely, given are a weight limit *limw* and a sequence of *n* objects consisting of weight and value of each object. Here *n* is a non-negative integer, and all the other values are *positive* integers. Solutions are subsets (rather, more precisely, subsequences) of objects whose sum of weights is at most *limw* and whose sum of values is as large as possible: one such solution is to be returned.

Thus, you are asked to write a function *knapsack(weights, values, n, limw)* that returns a list of integers less than *n* corresponding to objects whose total weight is at most *limw* and whose total value is as large as possible.

Observations

- 1/ The first two arguments are sequences of positive integers storing the information regarding each of the *n* objects.
- 2/ Note that there may be instances where the answer is an empty list, when no better solution is possible (recall that the sum of an empty list is zero).
- 3/ Sometimes, there will be several solutions: your function may return any solution that fulfills the given conditions.
- 4/ The time allowance of this problem is rather mild, and even quite inefficient solutions may be accepted (hence the "sluggish" adjective).

Problem information

Author: José Luis Balcázar

Generation: 2026-03-27T09:45:53.642Z

© Jutge.org, 2006–2026.

<https://jutge.org>