
Interval Subset Sum**X56351_en**

We consider a relaxed version of the standard Subset Sum problem (like in, e.g., P40685). However, in this problem, instead of a goal sum, we are provided a lower bound L and an upper bound U , and we are to add up values from a list t so that the sum falls between them. More precisely, given are: a lower limit L , an upper limit U , and a sequence t consisting of n values. Here n is a non-negative integer, and all the other values are *positive* integers. Solutions are subsets (rather, more precisely, subsequences) of t whose sum falls in the interval $[L, U)$ (closed at the left and open at the right: this should sound usual to pythonistas). Thus, you are asked to write a function `sub_sum_interval(L, U, t)` that returns a list of values taken from t that is as short as possible, and whose sum is at least L and strictly smaller than U . The solution may have repeated values only if they are already repeated in the given sequence. Note that solutions are always nonempty (why?); if no solution exists, your function should signal that by returning an empty list.

Observation

Often there will be several solutions: your function may return any solution that fulfills the given conditions. On the other hand, the time allowance of this problem is rather mild, and even quite inefficient solutions may be accepted.

Problem information

Author: José Luis Balcázar

Generation: 2026-01-25T17:00:21.690Z

© Jutge.org, 2006–2026.

<https://jutge.org>