Dichotomic search P81966_en

Write an efficient recursive function that returns the position of @x@ in the subvector @v@[@left@..@right@]. The function must return -1 if @x@ does not belong to @v@[@left@..@right@] or if @left@ > @right@.

Precondition

The vector @v@ is sorted in strictly increasing order. Moreover, we have $0 \leq$ @left@ $\leq$ size of v and $-1 \leq$ @right@ $<$ size of v.

Interface

C++ **int** *position* (**double** *x*, **const vector**<**double**>& *v*, **int** *left*, **int** *right*);
C **int** *position* (**double** *x*, **double** *v*[ ], **int** *left* , **int** *right*);
Java **public static int** *position* (**double** *x*, **double**[ ] *v*, **int** *left* , **int** *right*);
Python *position* (*x*, *v*, *left* , *right*) # returns int
MyPy *position* (*x*: **float** , *v*: **list** [**float**], *left* : **int** , *right* : **int**) −> **int**

Observation

You only need to submit the required procedure; your main program will be ignored.

Problem information

Author: Salvador Roura
Translator: Carlos Molina

Generation: 2026-01-25T11:55:55.857Z