```haskell
module LKSA_2018_10_19_ where
import Data.List

---------------------------------------
-- MURGILKETA
---------------------------------------
--1

batura_handiagoak_lag :: [[Integer]] -> Integer -> [[Integer]]
batura_handiagoak_lag s h
        | null s                          = s
        | (sum (head s)) > h              = (head s):(batura_handiagoak_lag (tail s) (sum (head s)))
        | otherwise                       = batura_handiagoak_lag (tail s) h


---------------------------------------
batura_handiagoak :: [[Integer]] -> [[Integer]]

batura_handiagoak s = batura_handiagoak_lag s 0


---------------------------------------

---------------------------------------
--BUKAERAKO ERREKURTSIBITATEA
---------------------------------------
--2

simetrikoa :: [Integer] -> [Integer] -> [(Integer,Integer)]
simetrikoa r s = (zip r s) ++ (reverse(zip s r))

---------------------------------------

simetrikoa_lag :: [Integer] -> [Integer] -> [(Integer,Integer)] -> [(Integer,Integer)] -> [(Integer,Integer)]
simetrikoa_lag r s l m
   | (null r) || (null s)   = l ++ m
```

```
    | otherwise                = simetrikoa_lag (tail r) (tail s) (l ++ [(head r, head s)]) ((head s, head r):m)


--------------------------------------
simetrikoa_be :: [Integer] -> [Integer] -> [(Integer,Integer)]
simetrikoa_be r s = simetrikoa_lag r s [] []
```