

# Lengoiak, Konputazioa eta Sistema Adimendunak

*Kudeaketaren eta Informazio Sistemen Informatikaren Ingeniaritzako Gradua*  
*Bilboko Ingeniaritza Eskola (UPV/EHU)*  
*Lengoia eta Sistema Informatikoak Saila*  
*2. maila — 2018-19 ikasturtea*

5. gaia eta 7. gaiaren 1. zatia: Konputazioaren konplexutasuna,  
 murgilketa eta bukaerako errekursibitatea  
 0,6 puntu

2018-10-19

## 1 Murgilketa (0,300 puntu)

*batura\_handiagoak* izeneko funtzioa definitu nahi da Haskelllez. Datu gisa zenbaki osozko zerrendez eratutako *s* zerrenda bat jasoko du. Emaizta gisa honako baldintza hau betetzen duten *s*-ko zerrendez eratutako zerrenda itzuliko du: elementuen batura positiboa ( $\geq 1$ ) izatea eta *s* zerrendan aurretik agertzen den zerrenda bakoitzeko elementuen batura baino handiagoa izatea. Zerrenda hutsaren batura 0 dela kontuan hartu behar da. *s* zerrenda hutsa baldin bada, zerrenda hutsa itzuli behar da. *s* hutsa ez denean, *s*-ko lehenengo zerrenda emaitzan agertuko da bere elementuen batura positiboa ( $\geq 1$ ) baldin bada.

*batura\_handiagoak* ::  $[[Integer]] \rightarrow [[Integer]]$   
*batura\_handiagoak s* ...

### Adibideak:

*batura\_handiagoak*  $[[2, 5, 5], [11, 8], [29, 0, 1], [19, 8, 2]] = [[2, 5, 5], [11, 8], [29, 0, 1]]$   
 $[2, 5, 5]$  zerrendako balioen batura positiboa da (12) eta lehenengoa da.  
 $[11, 8]$  zerrendako balioen batura 19 da, eta aurretik dagoen zerrenda bakarreko elementuen batura 12.  
 $[29, 0, 1]$  zerrendako balioen batura 30 da, eta aurretik dauden zerrendetako baturak 12 eta 19 dira.

*batura\_handiagoak*  $[[0, -5, -4], [], [-3], [11, 8], [6, 4], [19, 8, 2], [0, 2, 4]] = [[11, 8], [19, 8, 2]]$   
 $[11, 8]$  zerrendako balioen batura 19 da, eta aurretik dauden zerrendetako baturak -9, 0 eta -3 dira.  
 $[19, 8, 2]$  zerrendako batura 29 da, eta aurretik dauden zerrendetako baturak -9, 0, -3, 19 eta 10 dira.

*batura\_handiagoak*  $[[0, -5, -4], [0, 0, 0], [0, 2, -4]] = []$

*batura\_handiagoak*  $[[ -5, 1], [], []] = []$

*batura\_handiagoak* funtzioa zuzenean definitu beharrean, **murgilketaren** teknika erabili behar da eta jarraian zehazten diren ezaugarriak dituen *batura\_handiagoak\_lag* izeneko funtzioa definitu behar da. Funtzio horrek, datu gisa, zenbaki osozkoak diren zerrendez osatutako *s* zerrenda eta *h* zenbaki oso bat jasoko ditu. Emaizta gisa, honako baldintza hau betetzen duten *s*-ko zerrendez eratutako zerrenda itzuliko du: batura *h* baino handiagoa izatea eta *s* zerrendan aurretik agertzen den zerrenda bakoitzeko elementuen batura baino handiagoa izatea. Zerrenda hutsaren batura 0 dela kontuan hartu behar da. *s* zerrenda hutsa baldin bada, zerrenda hutsa itzuli behar da. *s* hutsa ez denean, *s*-ko lehenengo zerrendak emaitzan agertu behar du bere elementuen batura *h* baino handiagoa baldin bada.

*batura\_handiagoak\_lag* ::  $[[Integer]] \rightarrow Integer \rightarrow [[Integer]]$   
*batura\_handiagoak\_lag s h* ...

**Adibideak:**

*batura\_handiagoak\_lag*  $[[2, 5, 4], [11, 8], [29, 0, 1], [19, 8, 2]]$  15 =  $[[11, 8], [29, 0, 1]]$   
 [11, 8] zerrendako balioen batura 19 da. Beraz, 15 eta 11 baino handiagoa.  
 [29, 0, 1] zerrendako balioen batura 30 da: 15, 11 eta 19 baino handiagoa.

*batura\_handiagoak\_lag*  $[[2, 5, 4], [11, 8], [29, 0, 1], [19, 8, 2]]$  7 =  $[[2, 5, 4], [11, 8], [29, 0, 1]]$   
 [2, 5, 4] zerrendako balioen batura 11 da. Beraz, 7 baino handiagoa.  
 [11, 8] zerrendako balioen batura 19 da. Beraz, 7 eta 11 baino handiagoa.  
 [29, 0, 1] zerrendako balioen batura 30 da: 7, 11 eta 19 baino handiagoa.

*batura\_handiagoak\_lag*  $[[ -5, 1], [0, 0, 0], [0, 2, -4]]$  (-10) =  $[[ -5, 1], [0, 0, 0]]$   
 [-5, 1] zerrendako balioen batura -4 da, -10 baino handiagoa.  
 [0, 0, 0] zerrendako balioen batura 0 da, -10 eta -4 baino handiagoa.

*batura\_handiagoak\_lag*  $[[ -5, 1], [], [0, 2, -4]]$  (-10) =  $[[ -5, 1], []]$   
 [-5, 1] zerrendako balioen batura -4 da, -10 baino handiagoa.  
 [] zerrendako balioen batura 0 da, -10 eta -4 baino handiagoa.

*batura\_handiagoak\_lag* funtzioa *batura\_handiagoak* baino orokorragoa da; izan ere, *s*-ko zerrenda bat emaitza izango den zerrendan agertuko al den ala ez erabakitzeke, kontuan hartu beharreko beste balio bat ipintzea ahalbidetzen du *h* parametroaren bidez. Parametro berri hori aurreko zerrenda denei dagozkien baturetatik handiena gordez joateko erabiliko da. Horrela, *s*-ko zerrenda bakoitzera iritsitakoan, aurretik zeuden zerrenda denei dagozkien baturetatik handiena eskura izango dugu *h* parametroan.

Gainera, *batura\_handiagoak* funtzioaren **konputazio-kostua aztertu** behar da.

## 2 Bukaerako errekurtsibitatea (0,300 puntu)

Har dezagun honako funtzio hau:

$$\begin{aligned} \text{simetrikoa} &:: [\text{Integer}] \rightarrow [\text{Integer}] \rightarrow [(\text{Integer}, \text{Integer})] \\ \text{simetrikoa } r \ s &= (\text{zip } r \ s) ++ (\text{reverse}(\text{zip } s \ r)) \end{aligned}$$

*simetrikoa* funtzioak bikotez eratutako zerrenda bat itzuliko du. Zerrenda horretako lehenengo erdian *r* eta *s* zerrendetan posizio berean dauden elementuez osatutako bikoteak egongo dira. Bigarren erdia lehenengo erdiarekiko guztiz simetrikoa izango da: bikoteak alderantzizko ordenean ipinita eta bikote bakoitzaren osagaiak ere alderantzizko ordenean ipinita lortzen den zerrenda hain zuzen ere. *r* eta *s* zerrendek elementu kopuru desberdina baldin badute, bikote-eraketa zerrenda laburrena bukatzen denean amaituko da.

$$\begin{aligned} \text{simetrikoa } [1, 2, 3, 4] \ [8, 9, 10] &= [(1, 8), (2, 9), (3, 10), (10, 3), (9, 2), (8, 1)] \\ \text{simetrikoa } [1, 2] \ [8, 9, 10] &= [(1, 8), (2, 9), (9, 2), (8, 1)] \\ \text{simetrikoa } [1, 2, 3] \ [8, 9] &= [(1, 8), (2, 9), (9, 2), (8, 1)] \\ \text{simetrikoa } [] \ [8, 9, 10] &= [] \end{aligned}$$

*simetrikoa* funtzioak ez du bukaerako errekurtsibitaterik. Bukaerako errekurtsibitatea edukitzeko, **honako bi funtzio hauek definitu** behar dira:

- *simetrikoa\_lag* funtzioa: funtzio horrek *simetrikoa* funtzioak jasotzen dituen *r* eta *s* zenbaki osozko zerrendez gain, zenbaki osozko bikotez osatutako *l* eta *m* zerrendak jasoko ditu. Gure ikuspuntutik, *l* zerrenda emaitza gisa eraiki behar den zerrendaren lehenengo erdiko bikoteak gordez joateko erabiliko da eta *m* zerrenda bigarren erdiko bikoteak gordez joateko erabiliko da. Formalki, *simetrikoa\_lag* funtzioak honako lau zerrenda hauek elkartuz lortuko den zerrenda itzuliko du: **(1)** *l* zerrenda, **(2)** *r* eta *s* zerrendetan posizio berean dauden elementuez eratutako bikoteez osatutako zerrenda, **(3)** *s* eta *r* zerrendetan posizio berean dauden elementuez eratutako bikoteez osatutako zerrendaren alderantzizkoa eta **(4)** *m* zerrenda. *r* eta *s* zerrendek elementu kopuru bera ez badute, bikote-eraketa zerrenda laburrena bukatzen denean amaituko da.

$$\begin{aligned} \text{simetrikoa\_lag } [1, 2, 3, 4] \ [8, 9, 10] \ [(5, 6), (5, 7)] \ [(0, 11), (0, 12)] \\ = [(5, 6), (5, 7), (1, 8), (2, 9), (3, 10), (10, 3), (9, 2), (8, 1), (0, 11), (0, 12)] \end{aligned}$$

- *simetrikoa\_be* funtzioa: funtzio horrek *simetrikoa* funtzioak egiten duen gauza bera egin beharko du *simetrikoa\_lag* funtzioari egokiak diren parametroekin deituz.

$$\text{simetrikoa\_be } [1, 2, 3, 4] \ [8, 9, 10] = [(1, 8), (2, 9), (3, 10), (10, 3), (9, 2), (8, 1)]$$

Beraz, *simetrikoa* funtzioak egiten duena *simetrikoa\_be* eta *simetrikoa\_lag* funtzioak erabiliz egin ahal izango da.

Gainera, *simetrikoa* eta *simetrikoa\_be* funtzioen **konputazio-kostuak aztertu eta bata bestearekin alderatu** behar dira.