Lengoaiak, Konputazioa eta Sistema Adimendunak

7. gaia: Haskell – 1,6 puntu – Bilboko Ingeniaritza Eskola (UPV/EHU)

2016/11/16

1 Murgilketa (0,300 puntu)

Osoa eta positiboa den z zenbaki bat hartuta, z-ren ondoren jarraian dauden zenbaki batzuk (gutxienez zenbaki bat) batuz z-ren aurreko zenbaki positibo denen batura bera lortzen bada, orduan z *erdikoa* dela esan ohi da. Beraz, z *erdikoa* izango da z baino handiagoa den eta honako hau betetzen duen y zenbakirik baldin badago:

$$\sum_{k=1}^{z-1} k = \sum_{j=z+1}^{y} j$$

Datu bezala z zenbaki oso bat emanda, z zenbakia *erdikoa* al den erabakiko duen $erdikoa\_al\_da$ funtzioa definitu nahi da. Emandako z zenbakia positiboa ez bada, errore-mezua aurkeztu beharko da.

$$\begin{array}{l} erdikoa\_al\_da :: Integer -> Bool \\ erdikoa\_al\_da\ \ z\ \ \ldots \end{array}$$

Adibideak:

$erdikoa\_al\_da\ \ 6\ =\ True$	$1+2+\ldots+5 = 7+8$ delako. Hor $y=8$ da.
$erdikoa\_al\_da\ \ 7\ =\ False$	$1+2+\ldots+6$ balioa ezin delako lortu $8, 8+9, 8+9+10$, $8+9+10+11$ eta era horretako beste baturekin.

Murgilketaren teknika erabiliz, positiboa eta osoa den z zenbakia eta z baino handiagoa edo z-ren berdina den w zenbaki osoa emanda, $(z+1)+(z+2)+\ldots+w$ balioari w-ren ondoren jarraian dauden zenbaki batzuk (gutxienez zenbaki bat) batuz z-ren aurreko zenbaki positibo denen batura bera lor al daitekeen erabakitzen duen $erdikoa\_al\_da\_lag$ **funtzioa definitu** behar da. Emandako z zenbakia positiboa ez bada edo emandako w zenbakia z baino txikiagoa baldin bada, errore-mezua aurkeztu beharko da. Kontuan hartu $w = z$ betetzen baldin bada, $(z+1)+(z+2)+\ldots+w = 0$ izango dela.

$$\begin{array}{l} erdikoa\_al\_da\_lag :: Integer -> Integer -> Bool \\ erdikoa\_al\_da\_lag\ \ z\ w\ \ \ldots \end{array}$$

Adibideak:

$erdikoa\_al\_da\_lag\ \ 6\ 6\ =\ True$	$1+2+\ldots+5 = 0+7+8$ delako. Hor, $0 = (z+1)+(z+2)+\ldots+w$ da.
$erdikoa\_al\_da\_lag\ \ 6\ 10\ =\ False$	$1+2+\ldots+5$ balioa ezin delako lortu $(7+8+9+10)+11$, $(7+8+9+10)+11+12, (7+8+9+10)+11+12+13$ eta era horretako beste baturekin. Hor, $7+8+9+10 = (z+1)+(z+2)+\ldots+w$ da.
$erdikoa\_al\_da\_lag\ \ 6\ 8\ =\ False$	$1+2+\ldots+5$ balioa ezin delako lortu $(7+8)+9$, $(7+8)+9+10, (7+8)+9+10+11$ eta era horretako beste baturekin. Hor, $7+8 = (z+1)+(z+2)+\ldots+w$ da.

$erdikoa\_al\_da\_lag$ funtzioa $erdikoa\_al\_da$ baino orokorragoa da, w parametroaren bidez z-ren ondoren jarraian dauden zenbakietatik gutxienez zenbat batu nahi diren finkatzeko aukera ematen baitu: gutxienez $(z+1)+(z+2)+\ldots+w$ batura hartu beharko da abiapuntu bezala, hau da, gutxienez, $(z+1), (z+2), \ldots, w$ zenbakiak batu behar dira.

Gainera, $erdikoa\_al\_da$ funtzioaren **konputazio-kostua aztertu** behar da.

2 Bukaerako errekurtsibitatea (0,300 puntu)

Har dezagun honako funtzio hau:

$$
\begin{array}{l}
zip\_berria :: Integer \text{ -> } [Integer] \text{ -> } [Integer] \text{ -> } [(Integer, Integer)] \\
zip\_berria \;\; z\ r\ s \\
\quad | \quad (z < 0) \;||\; (z > (minimum\ [(length\ r), (length\ s)])) \;\; = \;\; error\ ''1.\ datua\ ez\ da\ egokia.'' \\
\quad | \quad (z == 0) \;\; = \;\; [\,] \\
\quad | \quad otherwise \;\; = \;\; (((head\ r), (head\ s)) : \\
\qquad\qquad\qquad\qquad\quad (zip\_berria\ (z-1)\ (tail\ r)\ (tail\ s)))
\end{array}
$$

zip_berria funtzioak zenbaki osozko r eta s zerrendetan posizio berean dauden lehenengo z elementuekin bikoteak eratzen ditu.

$$
\begin{array}{l}
zip\_berria\ \ 3\ [1,3,5]\ [2,4,6] \;=\; [(1,2),(3,4),(5,6)] \\
zip\_berria\ \ 2\ [1,3,5]\ [2,4,6] \;=\; [(1,2),(3,4)] \\
zip\_berria\ \ 3\ [4,2,5]\ [8,6,1,7,9] \;=\; [(4,8),(2,6),(5,1)] \\
zip\_berria\ \ 2\ [4,2,5]\ [8,6,1,7,9] \;=\; [(4,8),(2,6)]
\end{array}
$$

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

- zip_berria_lag funtzioa: funtzio horrek zip_berria funtzioak jasotzen dituen z zenbakia eta r eta s zenbakizko bi zerrendetaz gain, emaitza bezala eraikiz joango den bikote-zerrenda gordez joateko erabiliko den b bikote-zerrenda izango du laugarren parametro bezala. Beraz, zip_berria_lag funtzioak b zerrenda eta zenbaki osozko r eta s zerrendetan posizio berean dauden lehenengo z elementuekin osatutako bikoteak dituen zerrenda elkartuz lortzen den bikote-zerrenda itzuliko du. z negatiboa baldin bada edo z-ren balioa r eta s zerrendetatik laburrena denaren luzera baino handiagoa baldin bada, errore-mezua aurkeztu beharko du.

  $$
  \begin{array}{l}
  zip\_berria\_lag\ \ 3\ [1,3,5]\ [2,4,6]\ [\,] \;=\; [(1,2),(3,4),(5,6)] \\
  zip\_berria\_lag\ \ 3\ [1,3,5]\ [2,4,6]\ [(7,8),(9,10)] \;=\; [(7,8),(9,10),(1,2),(3,4),(5,6)] \\
  zip\_berria\_lag\ \ 2\ [1,3,5]\ [2,4,6]\ \ [(7,8),(9,10)] = \; [(7,8),(9,10),(1,2),(3,4)] \\
  zip\_berria\_lag\ \ 3\ [4,2,5]\ [8,6,1,7,9]\ [(7,8)] \;=\; [(7,8),(4,8),(2,6),(5,1)] \\
  zip\_berria\_lag\ \ 2\ [4,2,5]\ [8,6,1,7,9]\ [(9,10)] \;=\; [(9,10),(4,8),(2,6)]
  \end{array}
  $$

- zip_berria_be funtzioa: funtzio horrek zip_berria funtzioak egiten duen gauza bera egin beharko du zip_berria_lag funtzioari egokiak diren parametroekin deituz.

  $$
  \begin{array}{l}
  zip\_berria\_be\ \ 3\ [1,3,5]\ [2,4,6] \;=\; [(1,2),(3,4),(5,6)] \\
  zip\_berria\_be\ \ 2\ [1,3,5]\ [2,4,6] \;=\; [(1,2),(3,4)] \\
  zip\_berria\_be\ \ 3\ [4,2,5]\ [8,6,1,7,9] \;=\; [(4,8),(2,6),(5,1)] \\
  zip\_berria\_be\ \ 2\ [4,2,5]\ [8,6,1,7,9] \;=\; [(4,8),(2,6)]
  \end{array}
  $$

Beraz, zip_berria funtzioak egiten duena zip_berria_be eta zip_berria_lag funtzioak erabiliz egin ahal izango da.

Gainera, zip_berria eta zip_berria_be funtzioen **konputazio-kostua aztertu eta alderatu** behar da.

3 Zerrenda-eraketa (1,000 puntu)

3.1. (0,100 puntu) z zenbaki osoa eta zenbaki osozko zerrendez eratutako s zerrenda emanda, s-ko zerrenda bakoitzari z zenbakia ezkerretik erantsiz lortzen diren zerrenda denez eratutako zerrenda itzuliko duen $erantsi$ izeneko funtzioa definitu. Hor, s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$erantsi :: Integer \rightarrow [[Integer]] \rightarrow [[Integer]]$$
$$erantsi\ z\ s\ \ldots$$

Adibideak:

$$erantsi\ 5\ [[20,3,9,2],[\,],[9,10]] = [[5,20,3,9,2],[5],[5,9,10]]$$
$$erantsi\ 1\ [[9]] = [[1,9]]$$

3.2. (0,100 puntu) Zenbaki osozko r zerrenda eta zenbaki osozko zerrendez eratutako s zerrenda emanda, s-ko zerrenda bakoitzari r-ko elementu bakoitza ezkerretik erantsiz lortzen diren zerrenda denez eratutako zerrenda itzuliko duen $erantsi\_bakoitza$ izeneko funtzioa definitu. r zerrenda hutsa baldin bada, errore-mezua aurkeztu beharko da. s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$erantsi\_bakoitza :: [Integer] \rightarrow [[Integer]] \rightarrow [[Integer]]$$
$$erantsi\_bakoitza\ r\ s\ \ldots$$

Adibideak:

$$erantsi\_bakoitza\ [5,7]\ [[20,3,9],[\,],[9,10]] = [[5,20,3,9],[5],[5,9,10],$$
$$[7,20,3,9],[7],[7,9,10]]$$
$$erantsi\_bakoitza\ [1]\ [[2,3,4,2]] = [[1,2,3,4,2]]$$

Aukera bat aurretik definitutako $erantsi$ funtzioa eta aurredefinitutako $concat$ funtzioa erabiltzea da.

3.3. (0,100 puntu) Zenbaki osozko r zerrenda eta zenbaki osozko zerrendez eratutako s zerrenda emanda, s-ko zerrenda denez eta s-ko zerrenda bakoitzari r-ko elementu bakoitza ezkerretik erantsiz lortzen diren zerrenda denez eratutako zerrenda itzuliko duen $erantsi\_mantenduz$ izeneko funtzioa definitu. r zerrenda hutsa baldin bada, errore-mezua aurkeztu beharko da. s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$erantsi\_mantenduz :: [Integer] \rightarrow [[Integer]] \rightarrow [[Integer]]$$
$$erantsi\_mantenduz\ r\ s\ \ldots$$

Adibideak:

$$erantsi\_mantenduz\ [5,7]\ [[20,3,9],[\,],[9,10]] = [[20,3,9],[\,],[9,10],$$
$$[5,20,3,9],[5],[5,9,10],$$
$$[7,20,3,9],[7],[7,9,10]]$$
$$erantsi\_mantenduz\ [1]\ [[2,3,4,2]] = [[2,3,4,2],[1,2,3,4,2]]$$

Aukera bat aurretik definitutako $erantsi\_bakoitza$ funtzioa erabiltzea da.

3.4. (0,150 puntu) Zenbaki osozko r zerrenda eta zenbaki osozko zerrendez eratutako s zerrenda emanda, s-ko zerrenda denez eta s-ko zerrenda bakoitzari r-ko elementu bakoitza ezkerretik behin eta berriz erantsiz lortzen diren zerrenda denez eratutako zerrenda itzuliko duen $behin\_eta\_berriz$ izeneko funtzioa definitu. r zerrenda hutsa baldin bada, errore-mezua aurkeztu beharko da. s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$behin\_eta\_berriz :: [Integer] \rightarrow [[Integer]] \rightarrow [[Integer]]$$
$$behin\_eta\_berriz\ r\ s\ \ldots$$

Adibideak:

$$\begin{aligned}
&behin_eta_berriz\ \ [5,7]\ [[20,3,9],[\,],[9,10]] = [[20,3,9],[\,],[9,10],\\
&\qquad\qquad [5,20,3,9],[5],[5,9,10],\\
&\qquad\qquad [7,20,3,9],[7],[7,9,10],\\
&\qquad\qquad [5,5,20,3,9],[5,5],[5,5,9,10],\\
&\qquad\qquad [5,7,20,3,9],[5,7],[5,7,9,10],\\
&\qquad\qquad [7,5,20,3,9],[7,5],[7,5,9,10],\\
&\qquad\qquad [7,7,20,3,9],[7,7],[7,7,9,10]]\\
&\qquad\qquad [5,5,5,20,3,9],[5,5,5],[5,5,5,9,10],\\
&\qquad\qquad [5,5,7,20,3,9],[5,5,7,],[5,5,7,9,10],\\
&\qquad\qquad \ldots]\\
&behin_eta_berriz\ \ [1]\ [[2,3,4,2]] = [[2,3,4,2],[1,2,3,4,2],[1,1,2,3,4,2],\ldots]\\
&behin_eta_berriz\ \ [0,1]\ [[\,]] = [[\,],[0],[1],\\
&\qquad\qquad [0,0],[0,1],[1,0],[1,1],\\
&\qquad\qquad [0,0,0],[0,0,1],[0,1,0],[0,1,1],\\
&\qquad\qquad [1,0,0],[1,0,1],[1,1,0],[1,1,1],\ldots]
\end{aligned}$$

Oro har, infinitua izango den zerrenda itzuliko duen $behin\_eta\_berriz$ funtzioa definitzeko, aukera bat aurretik definitutako $erantsi\_bakoitza$ funtzioa erabiltzea eta $behin\_eta\_berriz$ funtzioa errekurtsiboa izatea da.

3.5. (0,100 puntu) Zenbaki osozko zerrendez eratutako s zerrenda emanda, lehenengo osagai bezala 0 zenbakia duten zerrendak kenduz gelditzen den zerrenda itzuliko duen $zerodunak\_kendu$ izeneko funtzioa definitu. **Salbuespena** 0 bakar batez osatutako zerrenda izango da, hasu horretan zerrenda ez baita kendu behar. s zerrendan zerrenda hutsa agertzen baldin bada, errore-mezua aurkeztu beharko da. s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$\begin{aligned}
&zerodunak_kendu :: [[Integer]] -> [[Integer]]\\
&zerodunak_kendu\ \ s\ \ldots
\end{aligned}$$

Adibideak:

$$\begin{aligned}
&zerodunak_kendu\ \ [[2,3,9,0],[4,8,8],[0,6,10]] = [[2,3,9,0],[4,8,8]]\\
&zerodunak_kendu\ \ [[0,0,1,0],[1,1,1],[1,0,1],[0]] = [[1,1,1],[1,0,1],[0]]\\
&zerodunak_kendu\ \ [[0,0,1,0]] = [\,]\\
&zerodunak_kendu\ \ [[0]] = [[0]]
\end{aligned}$$

3.6. (0,150 puntu) Zenbaki osoz eratutako s zerrenda emanda, ezkerretik hasi eta lehenengo berretzaile bezala 0 hartuz eta gero berretzaileak unitateka handituz, s-ko elementu bakoitza dagokion 10 zenbakiaren berreturaz biderkatuz lortzen den zerrenda itzuliko duen $berreturak$ izeneko funtzioa definitu. s zerrenda hutsa baldin bada, zerrenda hutsa itzuli beharko da.

$$\begin{aligned}
&berreturak :: [Integer] -> [Integer]\\
&berreturak\ \ s\ \ldots
\end{aligned}$$

Adibideak:

$$\begin{aligned}
&berreturak\ \ [9,3,4,8,8] = [9,30,400,8000,80000]\\
&berreturak\ \ [1,0,1,1] = [1,0,100,1000]\\
&berreturak\ \ [15,-20,7,100] = [15,-200,700,100000]
\end{aligned}$$

Aukera bat aurredefinitutako $length$ eta zip funtzioak erabiltzea da.

3.7. (0,150 puntu) Digituz (0 eta 9ren arteko zenbaki osoz) eratutako s zerrenda emanda, zerrendak adierazten duen zenbakizko balioa itzuliko duen $zenbakizko\_balioa$ izeneko funtzioa definitu. s zerrendan digitua ez den zenbakiren bat baldin badago, errore-mezua aurkeztu beharko da. s zerrenda hutsa baldin bada, errore-mezua aurkeztu beharko da.

$$zenbakizko\_balioa :: [Integer] -> Integer$$
$$zenbakizko\_balioa\ \ s\ \ldots$$

Adibideak:

$$zenbakizko\_balioa\ \ [9,3,4,8,8] = 93488$$
$$zenbakizko\_balioa\ \ [1,0,1,1] = 1011$$
$$zenbakizko\_balioa\ \ [8,8,4,3,9] = 88439$$

Aukera bat aurredefinitutako $reverse$ eta sum funtzioak eta aurreko ariketan definitutako $berreturak$ funtzioa erabiltzea da.

3.8. (0,150 puntu) $zenbakizko\_balioa$ funtzioaren konputazio-kostua aztertu.