

მრავალარგუმენტიანი ფუნქციები

ამასთან ერთად, მრავალარგუმენტიანი ფუნქციები ასევე შეიძლება განისაზღვროს რეკურსიის გამოყენებით ორ ან მეტ არგუმენტზე. მაგალითად, საბიბლიოთეკო *zip* ფუნქცია, რომელიც იღებს შესასვლელზე ორ სიას და წყვილების შექმნის სიას გვიბრუნებს, შემდეგი სახით განისაზღვრება:

$$\begin{aligned} \text{zip} &:: [a] \rightarrow [b] \rightarrow [(a, b)] \\ \text{zip } [] _ &= [] \\ \text{zip } _ [] &= [] \\ \text{zip } (x : xs) (y : ys) &= (x, y) : \text{zip } xs \ ys \end{aligned}$$

მაგალითად:

$$\begin{aligned} &\text{zip } ['a', 'b', 'c'] [1, 2, 3, 4] \\ = &\{ \text{zip ფუნქციის გამოყენება } \} \\ &('a', 1) : \text{zip } ['b', 'c'] [2, 3, 4] \\ = &\{ \text{zip ფუნქციის გამოყენება } \} \\ &('a', 1) : ('b', 2) : \text{zip } ['c'] [3, 4] \\ = &\{ \text{zip ფუნქციის გამოყენება } \} \\ &('a', 1) : ('b', 2) : ('c', 3) : \text{zip } [] [4] \\ = &\{ \text{zip ფუნქციის გამოყენება } \} \\ &('a', 1) : ('b', 2) : ('c', 3) : [] \\ = &\{ \text{ჩაწერა სიის სახით } \} \\ &[('a', 1), ('b', 2), ('c', 3)] \end{aligned}$$

ყურადღება მიაქციეთ, რომ *zip* ფუნქციის განსაზღვრებაში საჭიროა ორი საბაზო გამოსახულება, ვინაიდან არგუმენტთა ორი სიიდან ნებისმიერი შეიძლება აღმოჩნდეს ცარიელი.

განვიხილოთ რამდენიმე არგუმენტზე რეკურსიის კიდევ ერთი მაგალითი. სახელდობრ, საბიბლიოთეკო *drop* ფუნქცია, რომელიც სიაში ანადგურებს ელემენტების მოცემულ რაოდენობას ამ სიის დასაწყისიდან, შემდეგი სახით შეიძლება განისაზღვროს:

$$\begin{aligned} \text{drop} &:: \text{Int} \rightarrow [a] \rightarrow [a] \\ \text{drop } 0 \ xs &= xs \\ \text{drop } (n + 1) \ [] &= [] \\ \text{drop } (n + 1) \ (_ : xs) &= \text{drop } n \ xs \end{aligned}$$

აქ კვლავ ორი საბაზო (საყრდენი, საყრდნობი) გამოსახულებაა საჭირო: ერთი – ნულოვანი რაოდენობის ელემენტთა გასანადგურებლად სიაში, ხოლო მეორე – ცარიელ სიაში ერთი ან რამდენიმე ელემენტის განადგურების მცდელობისას.