საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ასოც.პროფ. ნათელ არჩვაძე

ქულათა ჯამი: 20

ქულათა გადანაწილების ინსტრუქცია: 1-4 საკითხი -2ქ., 5-8 საკითხი -3ქ. =16

1. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს მთელ რიცხვს n-ს და აბრუნებს 1-დან n -მდე 2-ის ხარისხების სიას.

-- 1. +2

orisxarisxebi :: (Integral b, Num a) => [b] -> [a]

orisxarisxebi [] = []

orisxarisxebi (x:xs) = (2^x) : orisxarisxebi xs

1. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს მთელ რიცხვს n-ს და სიმბოლოს, და აბრუნებს სტრიქონს, რომელშიც სიმბოლო n-ჯერ არის განმეორებული. მაგალითად, ff 5 '%'="%%%%%"

ff :: (Eq t, Num t) => t -> a -> [a] +2

ff 0 \_ = []

ff n m = m : ff (n-1) m

1. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს მთელ რიცხვებს a-ს და b-ს და აბრუნებს a-დან b -მდე ლუწი ხუთის ჯერადი რიცხვების სიას.

hh :: Integral a => a -> a -> [a] +2

hh n m = gg n m n

-- დამხმარე ფუნქცია

gg :: Integral a => t -> a -> a -> [a]

gg n m b

| b > m = []

| mod b 10 == 0 = b : gg n m (b+10)

| otherwise = gg n m (b+1)

1. ფუნქცია გამოყოფს მოცემული სიის n - ურ წევრს. შეცვალეთ ფუნქციის განმარტება - გაითვალისწინეთ შემთხვევა, როცა n-ის მნიშვნელობა სიის სიგრძეზე მეტია.

disc :: (Eq t, Num t) => t -> [p] -> p +1

disc 0 (x:xs) = x

disc \_ [] = errorBadArgument

disc n (x:xs) = disc (n-1) xs

1. განსაზღვრეთ ორი სიის ელემენტების აჯამვის ფუნქცია. აბრუნებს სიას, რომელიც შედგება პარამეტრი სიების ელემენტების ჯამისგან. თუ არგუმენტებს სხვადასხვა სიგრძე აქვთ, ფუნქციის შედეგი იყოს უფრო მოკლე სიის სიგრძის. მაგალითად, listsum [1,2,3,8,3][4,5]=[5,7]; listsum [8][1,2,3,4,5]=[9]

listsum :: Num a => [a] -> [a] -> [a] +3

listsum [] \_ = []

listsum \_ [] = []

listsum (x:xs) (y:ys) = x+y : listsum xs ys

1. ფუნქცია, რომელიც აადგილებს მოცემულ სიაში მეზობელ ლუწ და კენტ ადგილას მყოფ ელემენტებს ელემენტებს.

swaplist :: Integral a => [a] -> [a]

swaplist [] = []

swaplist (x:[]) = []

swaplist (x:y:xs)

| odd x && even y = y:x : swaplist xs

| even x && odd y = y:x : swaplist xs

| otherwise = swaplist xs

არ მუშაობს swaplist [2,4, 1,3]

[]

1. ფუნქცია delete :: Char -> String -> String, რომელიც იღებს შესასვლელზე სტრიქონს და სიმბოლოს და აბრუნებს სტრიქონს, რომლიდანაც ამოშლილია ყველა სიმბოლო გარდა მოცემული სიმბოლოსი. მაგალითად, delete ’l’ "Hello world!" უნდა დააბრუნოს "ll".

delete :: Char -> String -> String +3

delete n [] = []

delete n (x:xs)

| x /= n = delete n xs

|otherwise = x : delete n xs

1. განსაზღვრეთ ფუნქცია, რომელიც მოცემული არგუმენტიში (სიაში) დატოვებს მხოლოდ იმ a რიცხვებს, რომლებიც აკმაყოფილებენ პირობას: 150<a<400.

datoveba :: (Ord a, Num a) => [a] -> [a] +3

datoveba [] = []

datoveba (x:xs)

| x > 150 && x < 400 = x : datoveba xs

|otherwise = datoveba xs