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. =14

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

xarisxilist :: Num a => [a] -> [a] -

xarisxilist [] = []

xarisxilist (k:ks) = k\*k : xarisxilist ks

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

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

function 0 \_ = []

function 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.

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

luwilist a b c +2

| c > b = []

| mod c 10 == 0 = c : luwilist a b (c+10)

| otherwise = luwilist a b (c+1)

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

functiond :: Eq a => a -> [a] -> [a] +1

functiond n [] = []

functiond n (k:ks)

| k == n = FunctionD n ks

| otherwise = x : FunctionD n ks

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]

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

sumlist [] = []

sumlist \_ [] = []

sumlist (k:ks) (x:xs) = k+x : sumlist ks xs

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

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

swap [] = []

swap (k:[]) = []

swap (k:x:ks)

| odd k && even x = x:k : swap ks

| even k && odd x = x:k : swap ks

| otherwise = swap ks

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 n [] = []+3

delete n (k:ks)

| x /= n = delete n ks

|otherwise = x : delete n ks

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

datoveba [] = [] +3

datoveba (k:ks)

| k < 150 && k > 400 = datoveba ks

|otherwise = k : datoveba ks