საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ნათელ არჩვაძე

დრო: 2 სთ

ქულათა ჯამი: 25=3+0+5+0+4=12

ქულათა გადანაწილების ინსტრუქცია: 5 საკითხი, თითოეული 5 ქულა.

ბილეთი =12

1. განსაზღვრეთ შემდეგი ფუნქცია ორი ხერხით: პირობითი ოპერატორისა და დაცული განტოლებების გამოყენებით:  მნიშვნელობის
   1. y x = if x >= (-2) && x /= 0.4 then (3x^2 – 7x -3)/(5x -2)

else if x == 0.4 then 4x^3 +100/3x - 5 gamravlebebi \*

else 7(3.5+2x) -1/2.5(1-x)^2

y z x | (x >= (-2) && x /= 0.4) = (3x^3 -7x -3)/(5x -2) არგ. რაოდ

| (x == 0.4) = 4x^3+100/3x-5

| otherwise = 7(3.5+2x) -1/2.5(1-x)^2 +3

\*Main> :reload

Ok, modules loaded: Main.

\*Main> y 1.2

0.18199999999999994

it :: Double

(0.02 secs, 1677020 bytes)

\*Main> y 5.3

6.035795918367346

it :: Double

(0.02 secs, 1714148 bytes)

\*Main> y (-2)

0.75

it :: Double

(0.00 secs, 1729000 bytes)

\*Main>

გამოსათვლელად. ფუნქციის ტიპი განსაზღვრეთ ცხადად. როგორ მიმართავთ ამ ფუნქციას? -1

1. განსაზღვრეთ შემდეგი ფუნქციები λ ნოტაციის გამოყენებით და მიუთითეთ შესაბამისი ტიპები: 0

* ფუნქცია, რომელსაც გადაეცემა ორი არგუმენტი და ითვლის ნამდვილ ხარისხს.

**Prelude>** 1) Let f1= ( \(x,y) ->(y,x)) sxvaa

F1 : : (t1 , t) -> (t, t1)

* ფუნქცია, რომელიც ორჯერ ითვლის მნიშვნელობას არგუმენტზე.

let f2 = map( \x-> if x/3 than x\*x else x\*x\*) sxvaa

F2 : : [Integer] ->

* ფუნქცია, რომელიც გადაცემული რიცხვების სიისთვის ამოწმებს თითოეული ელემენტის ლუწ-კენტობას და გვიბრუნებს ლოგიკური მნიშვნელობების სიას.

f = \(xs) -> \(ys) -> ((xs)++(ys)) sxvaa

Prelude> f = \(xs) -> \(ys) -> ((xs)++(ys))

f :: [a] -> [a] -> [a]

Prelude> f [2..5] [5..8]

[2,3,4,5,6,7,8]

it :: [Integer]

1. განსაზღვრეთ ცხრილით მოცემული ფუნქცია **რამდენიმე** შაბლონის გამოყენებით:

|  |  |  |
| --- | --- | --- |
| a | b | a⇒b |
| False | False | True |
| False | True | False |
| True | False | False |
| True | True | True |

განსაზღვრეთ ფუნქციის ტიპი.

1. სიის კონსტრუქტორის გამოყენებით შეადგინეთ: -0

* ხუთნიშნა რიცხვების სია, სადაც ყველა ციფრი ერთნაირია;

**Prelude>** [7,7,7,7,7] sxvaa

[7,7,7,7,7]

it :: [Integer]

* გამოსახულება, რომლითაც დაითვლით 1000–ზე ნაკლები 3–ის ან 7–ის ჯერადი ნატურალური რიცხვების ნამრავლი ?
* სია [22, 44,...2020].

1. განსაზღვრეთ ფუნქცია ორი ხერხით: REPL გარემოში და საწყისი კოდის სახით. ფუნქცია გამოიძახეთ კონკრეტული მონაცემებისთვის:

* ფუნქცია, რომელიც გადაცემული სიისთვის აბრუნებს ლუწი ელემენტების კვადრატებს, კენტების -კუბებს.

let fun = \xs -> map (\x -> if rem x 2==0 then x^2 else x^3)xs

* ფუნქცია, რომელსაც გადაეცემა ორი სია და დააბრუნებს მათ გაერთიანებას.

let fun = \(xs) -> \(ys) -> ((xs)++(ys))

* ფუნქცია isRectangular, რომელიც პარამეტრად ღებულობს სიბრტყეზე სამი წერტილის კოორდინატებს და აბრუნებს True–ს, თუ მათ მიერ შედგენილი სამკუთხედი არის მართკუთხა სამკუთხედი.

ნომერი 3

let Istrue a b = if a == b then True

else False

ტიპი : istrue :: eq a => a -> a -> Bool

შაბლონების გამოყენებით:

function False False = True

function True True = True

function \_ \_ = False

function :: Bool -> Bool -> Bool