საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ნათელ არჩვაძე

დრო: 2 სთ

ქულათა ჯამი: 25=4+4+5+0+4

ქულათა გადანაწილების ინსტრუქცია: 5 საკითხი, თითოეული 5 ქულა.

ბილეთი =17

1. განსაზღვრეთ შემდეგი ფუნქცია ორი ხერხით: პირობითი ოპერატორისა და დაცული განტოლებების გამოყენებით:  მნიშვნელობის გამოსათვლელად. ფუნქციის ტიპი განსაზღვრეთ ცხადად. როგორ მიმართავთ ამ ფუნქციას?
2. განსაზღვრეთ შემდეგი ფუნქციები λ ნოტაციის გამოყენებით და მიუთითეთ შესაბამისი ტიპები:

* ფუნქცია, რომელსაც გადაეცემა ორი არგუმენტი და ითვლის ნამდვილ ხარისხს.
* ფუნქცია, რომელიც ორჯერ ითვლის მნიშვნელობას არგუმენტზე.
* ფუნქცია, რომელიც გადაცემული რიცხვების სიისთვის ამოწმებს თითოეული ელემენტის ლუწ-კენტობას და გვიბრუნებს ლოგიკური მნიშვნელობების სიას.

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:

* ხუთნიშნა რიცხვების სია, სადაც ყველა ციფრი ერთნაირია;
* 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.
* ფუნქცია, რომელსაც გადაეცემა ორი სია და დააბრუნებს მათ გაერთიანებას.
* ფუნქცია isRectangular, რომელიც პარამეტრად ღებულობს სიბრტყეზე სამი წერტილის კოორდინატებს და აბრუნებს True–ს, თუ მათ მიერ შედგენილი სამკუთხედი არის მართკუთხა სამკუთხედი.

1

func x = if x == 0.4 then 4 \* x \*\* 3 +(100 / 3 \* x) – 5 +4

else if x < -2 then 7 \* (3.5 + 2 \* x) - 1 / (2.5 \* (1 - x))

else 3 \* x \* x - 7 \* x - 3 / (5 \* x - 2)

func :: (Floating a, Ord a) => a -> a

func1 x

|x == 0.4 = 4 \* x \*\* 3 +(100 / 3 \* x) – 5

|x < -2 = 7 \* (3.5 + 2 \* x) - 1 / (2.5 \* (1 - x)) |

otherwise = 3 \* x \* x - 7 \* x - 3 / (5 \* x - 2)

func1 :: (Floating a, Ord a) => a -> a

2

1)f1 = (\x y -> x \*\* y)

f1 :: Double -> Double -> Double

3) f2 = \xs ->[if even x then True else False| x <- xs]

f2 :: [Integer] -> [Bool]

2)

3

fs1 True True = True

fs1 False False = True

fs1 \_ \_ = False

fs1 :: Bool -> Bool -> Bool

fs2 a b = if a == b then True else False

fs2 :: Eq a => a -> a -> Bool

5

1) f1 xs = [if even x then x \* x else x \* x \* x| x <- xs]

f1 :: Integral a => [a] -> [a]

\*Midterm> f1 [1,2,3,4]

[1,4,27,16]

let f1 xs = [if even x then x \* x else x \* x \* x| x <- xs]

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

\*Midterm> f [2,3,4,5,6]

[4,27,16,125,36]

2) f2 xs ys = xs ++ ys

f2 :: [a] -> [a] -> [a]

\*Midterm> f2 [1,2] [3,4,5]

[1,2,3,4,5]

let f xs ys = xs ++ ys

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

\*Midterm> f [3,4,5] [1,2]

[3,4,5,1,2]

3)-0