საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ნათელ არჩვაძე

დრო: 2 სთ

ქულათა ჯამი: 25=3+4+3+0+4

ქულათა გადანაწილების ინსტრუქცია: 5 საკითხი, თითოეული 5 ქულა.

ბილეთი=14

1. განსაზღვრეთ შემდეგი ფუნქცია ორი ხერხით: პირობითი ოპერატორისა და დაცული განტოლებების გამოყენებით:  მნიშვნელობის გამოსათვლელად. როგორ მიმართავთ ამ ფუნქციას? ფუნქციის ტიპი განსაზღვრეთ ცხადად.

**პირობითი ოპერატორებით:**

fun x = if x == (-8) then 7\*x + (1/(1+x))/(x\*x – 1) + 1/(2\*x)

else if x <= 5 && x/= (-8) then (3\*x\*x) – ((5)/2\*(x+8)) + 10

else (4\*x – 5\*x\*x + 2)/(2+2\*x) – (1)/(3\*(x-1)

fun :: (floating a, Ord a) => a -> a +3

**დაცული განტოლებებით:**

fun x | (x <= 5 && x/= (-8)) = (3\*x\*x) – ((5)/2\*(x+8)) + 10 არაა სწორე

| (x == (-8)) = 7\*x + (1/(1+x))/(x\*x – 1) + 1/(2\*x)

| otherwise = (4\*x – 5\*x\*x + 2)/(2+2\*x) – (1)/(3\*(x-1)

fun :: (floating a, Ord a) => a -> a

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

* ფუნქცია, რომელიც აბრუნებს გაორმაგებულ არგუმენტს.

fun = (\a -> a\*a) -1

fun :: Integer -> Integer

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

fun = (\a b -> a ^ b)

fun :: Integer -> Integer -> Integer

* ფუნქცია, რომელსაც გადაეცემა ორი წყვილი და დააბრუნებს შესაბამისი ელემენტების ჯამების წყვილს. მაგალითად, არგუმენტებია: (1,2), (3,4), შედეგია: (4,6).

add = (\(x, y) (u, v) -> (x+u, y+v))

add :: (Integer, Integer) -> (Integer, Integer) -> (Integer, Integer)

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

|  |  |  |
| --- | --- | --- |
| a | b | a $$ b |
| False | False | True |
| False | True | True |
| True | False | False |
| True | True | True |

განსაზღვრეთ ფუნქციის ტიპი.

fun1 True False = False +3

fun1 \_ \_ = True

fun1 :: Bool -> Bool -> Bool

1. სიის კონსტრუქტორის გამოყენებით შეადგინეთ: -0

* იმ სამნიშნა რიცხვების სია, რომელთა ციფრთა ჯამი არის 3-ის ჯერადი რიცხვი;
* მოცემულ n რიცხვამდე ისეთი რიცხვების სია, რომელიც ორი მარტივი რიცხვის ჯამით; წარმოდგება.
* გამოსახულება , რომელიც პირველი 100 რიცხვის 12 + 22 + . . . + 1002 ჯამს ანგარიშობს.

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

* ფუნქცია, რომელიც მოცემულ წყვილში ადგილებს უცვლის ელემენტებს.

fun (a,b) = (b,a) +1

**\*Main>** fun (1,2)

(2,1)

it :: (Integer, Integer)

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

fun = \xs -> [if even x then True else False | x <- xs] +2

**\*Main>** fun [1,2,3,4,5]

[False,True,False,True,False]

* ფუნქცია isTriangle, რომელიც განსაზღვრავს, შეიძლება თუ არა მოცემულ x, y და z სიგრძის მონაკვეთებზე აიგოს სამკუთხედი.

triangles = [(x,y,z) | x <- [1..10], y <- [1..10], z <- [1..10], z < x + y]