საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი:ნათელ არჩვაძე

ქულათა ჯამი:15

ქულათა გადანაწილების ინსტრუქცია: 1 საკ -3ქ., 2 საკ -2ქ., 3 საკ -3ქ.,4 საკ-2 ქ., 5 საკ -5 ქ.

ბილეთი N=9

1. მოიყვანეთ არატრივიალური გამოსახულებების მაგალითები, რომლებიც ეკუთვნის ტიპებს:

a)(([Bool],[Double]),[Integer])

b)[([Integer],[Double],[(Bool,Char)])]

c)([Integer],[Double],[(Bool,Char)])

მოთხოვნა გამოსახულებების არატრივიალურობის შესახებ ნიშნავს, რომ გამოსახულებებში მონაწილე სიები უნდა შეიცავდნენ ერთ ელემენტზე მეტს.

a) (([True, False],[1.4,3.5,4.1]),[1,2,3])

b) [([1,2,3,4],[1.2,2.4,5.6],[(False,’a’,’b’,’c’)])] მეორე წევრი -1

c) ([3,4,5,6],[2.3,3.1,6.5],[(True,’g’,’h’)])

1. როგორია შემდეგი მნიშვნელობების ტიპები: -2

[(False,’0’),(True,’1’)]

(’a’,’b’,’c’)

[“a”,”b”,”c”]

<interactive>:1:1:

The function `[(False, '0'), (True, '1')]'

is applied to two arguments,

but its type `[(Bool, Char)]' has none

In the expression:

[(False, '0'), (True, '1')] ('a', 'b', 'c') ["a", "b", "c"]

1. განსაზღვრეთ შემდეგი ფუნქციების მხოლოდტიპებიპოლიმორფიზმის გამოყენებით:
2. swap (x,y) = (y,x)
3. fun1 xs=(sum xs)/(product xs)
4. last
5. [tail,init]

თითოეული ფუნქცია გამოიძახეთ კონკრეტული მონაცემებისთვის. -2

Prelude> let swap ( x,y ) = ( y,x )

Prelude> :type swap

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

Prelude> let fun1 xs = ( sum xs ) / (product xs )

Prelude> :type fun1

Fun1 : : fractional a => [a] -> a

1. მოცემულია ფუნქციის ტიპები:

a)fun :: Fractional a => a -> a -> a

b)fun :: (Ord a, Num a) => a -> a -> Bool

თითოეული ტიპისთვის მოიყვანეთ მაგალითი ფუნქციებისა, რომლებსაც ექნებათ მოცემული ტიპი.

1. ტოლობის ძირითადი კლასი.

კლასი არის მეთოდებად წოდებული გარკვეული გადატვირთული ოპერაციესის მხარდამჭერი ტიპების კოლექცია

ჰასკელში აქვს ტიპთა ბევრი კლასი მათშორის არის Eq ტოლობის ტიპებიც

ეს არის კლასი ,რომელიც შეიცავს ტიპებს ,რომელთა მნიშვნელობების შედარება შეიძლება მათი ტოლობის ან უტოლობის დასადგენად

(*==*) :: *a → a → Bool*

(*≠*) :: *a → a → Bool*

ყველა ძირითადი ტიპი bool ,string, integer ,char and float არის Eq კლასის ეგზემპლარი როგორც სიის და კკორტეჟის ტიპი. გარდა??? -1

თუ წარმოადგენს კლასის ეგზემპლარებს მაგ:

*> False ==False*

*True*

*>*'a' *==* 'b'

*False*

*>*"abc" *==* "abc"

*True*

*>*[1*,* 2] *==* [1*,* 2*,* 3]

*False*

*>*('a'*, False*) *==* ('a'*, False*)

*True*