საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ნათელ არჩვაძე

დრო: 2 სთ

ქულათა ჯამი: 25=4+2+2+0+0

ქულათა გადანაწილების ინსტრუქცია: 5 საკითხი, თითოეული 5 ქულა.

სახელი და გვარი: ილკინ გოჯაევი

ბილეთი =8

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

* ფუნქცია, რომელიც აბრუნებს სიის მეორე ელემენტს.
* ფუნქცია, რომელიც გადაცემული სიისთვის აბრუნებს ლუწი ელემენტების კვადრატებს, კენტების -კუბებს.
* ფუნქცია, რომელსაც გადაეცემა ელემენტი და სია და დააბრუნებს ახალ სიას, სადაც ბოლო ელემენტად ჩასმულია პირველი არგუმენტი.

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.

1. სიის კონსტრუქტორის გამოყენებით შეადგინეთ:

* იმ სამნიშნა რიცხვების სია, რომელთა ციფრთა ჯამი არის კენტი რიცხვი.
* შეადგინეთ ისეთი (x,y,z) სამეულების სია, სადაც თითოეული ცვლადი არის მარტივი, მნიშვნელობით 1–დან n-მდე და ჭეშმარიტია პირობა x+y==z.
* სია [22, 33, ... 2020].

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

* ფუნქცია, რომელიც აწყვილებს თავის ორ არგუმენტს.
* ფუნქცია, რომლის არგუმენტია სტრიქონი და რომელიც ამოწმებს, არის თუ არა არგუმენტი პალინდრომი.
* ფუნქცია isSorted, რომელიც შესასვლელზე ღებულობს სამ რიცხვს და აბრუნებს True, თუ ეს რიცხვები დალაგებულია ზრდადობით ან კლებადობით.

დავალება 1) ტიპი -

Handlefunc n = if n >= -3 && n<=7 && not (n == -2.5) then ((1-n\*n +8.2\*n)/(n+2.5))-4 else

if n == (-2.5) then ((3.75 \* n \* n) + (10 / ((2 - n)(1 + 4\*n)))) else ((5\*n-7)+((3.5\*n\*n\*n)/(4(1.5+n))))

Handlefunc n | n >= -3 && n<=7 && not (n == -2.5) = ((1-n\*n +8.2\*n)/(n+2.5))-4

| n == (-2.5) = ((3.75 \* n \* n) + (10 / ((2 - n)\*(1 + 4\*n))))

| otherwise = ((5\*n-7)+((3.5\*n\*n\*n)/(4\*(1.5+n))))

დავალება 2)

ა) list = [1, 2, 3]

head :: [a] -> a +1

head (x:\_) = x

head list -- 2

ბ) squares low high = map (^2) . filter even $ [low..high] +1

დავაელბა 3)

Handlefunc :: Bool -> Bool -> Bool +1

Handlefunc False False = True +1

Handlefunc False True = True

Handlefunc True False = False

Handlefunc True True = True