საგნის დასახელება: ფუნქციონალური პროგრამირება

სპეციალობა: კომპიუტერული მეცნიერება

ლექტორი: ასოც.პროფ. ნათელ არჩვაძე

ქულათა ჯამი: 20

ქულათა გადანაწილების ინსტრუქცია: 1-4 საკითხი -2ქ., 5-8 საკითხი -3ქ. =15

1. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს მთელ რიცხვს n-ს და აბრუნებს 1-დან n -მდე ნატურალური რიცხვის ფაქტორიალების სიას.

**factorial :: (Eq p, Num p) => p -> p**

**factorial 0 = 1 +2**

**factorial n = n \* factorial(n-1)**

**factlist :: (Eq a, Num a) => a -> [a]**

**factlist 0 = [1]**

**factlist n = factorial n : factlist(n-1)**

**rfactlist :: (Eq a, Num a) => a -> [a]**

**rfactlist n = reverse (factlist n)**

**facttlist :: (Eq a, Num a) => [a] -> [a]**

**facttlist [] = []**

**facttlist (x:xs) = factorial x : facttlist xs**

1. სტანდარტული prelude ფაილის გამოუყენებლად რეკურსიის საშუალებით განსაზღვრეთ ფუნქცია, რომელიც დაადგენს, არის თუ არა მოცემული მნიშვნელობა სიის ელემენტი: elem::Eq a=>a->[a]->Bool.

**myelem::Eq a=>a->[a]->Bool**

**myelem n [] = False +2**

**myelem n (x:xs)**

**| n == x = True**

**| otherwise = chemielem n xs  
  
-- let myelem n xs = or[x == n | x <- xs]**

1. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს მთელ რიცხვებს a-ს და b-ს და აბრუნებს a-დან b -მდე ლუწი 11-ის ჯერადი რიცხვების სიას.   
    **let eveneleven a b +2  
   | a == b = []   
   | mod a 11==0 && even a= a:eveneleven(a+1) b  
   | otherwise = eveneleven (a+1) b**
2. განსაზღვრეთ ფუნქცია, რომელიც მოცემული სიიდან n-ურ წევრს შეცვლის მისი კვადრატით.
3. განსაზღვრეთ ფუნქცია, რომელიც შესასვლელზე ღებულობს სიას და აბრუნებს სიის ლუწი რიცხვების სიას გაერთიანებულს კენტი რიცხვების სიასთან თავდაპირველ სიაში მათი თანმიმდევრობის შენარჩუნებით. მაგალითად, fun [1,2,3,4,5,6,7,8,9] = [2,4,6,8, 1,3,5,7,9].

**swap [] = [] -**

**swap (x:[]) = []**

**swap (x:y:xs)**

**| even x && odd y = y:x : swap xs**

**| odd x && even y = y:x : swap xs**

**| otherwise = swap xs**

1. ფუნქცია removeEmpty, რომელიც ამოაგდებს ცარიელ სტრიქონებს სტრიქონების მოცემული სიიდან. მაგალითად, removeEmpty ["", "Hello", "", "", "World!"] უნდა დააბრუნოს ["Hello","World!"].

**removeEmpties :: [[Char]] -> [[Char]] +3**

**removeEmpties [] = []**

**removeEmpties (x:xs)**

**| x == "" = removeEmpties xs**

**|otherwise = x : removeEmpties xs**

1. განსაზღვრეთ ფუნქცია makePositive, რომელიც უცვლის ნიშანს რიცხვების სიის ყველა უარყოფით ელემენტს. მაგალითად, makePositive [-1, 0, 5, -10, -20] გვაძლევს [1,0,5,10,20].

**makePositive :: (Ord a, Num a) => [a] -> [a] +3**

**makePositive [] = []**

**makePositive (x:xs)**

**| x < 0 = negate x : makePositive xs**

**|otherwise = x : makePositive xs**

1. განსაზღვრეთ ფუნქცია, რომელიც მოცემული არგუმენტიდან (სიიდან) ამოაგდებს მხოლოდ ორნიშნა რიცხვებს.

**twodigit :: (Ord a, Num a) => [a] -> [a] +3**

**twodigit [] = []**

**twodigit (x:xs)**

**| x > 9 && x < 100 = twodigit xs**

**|otherwise = x : twodigit xs**