**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**КАФЕДРА ІНТЕЛЕКТУАЛЬНИХ ТЕХНОЛОГІЙ**

**Лабораторна робота № 2**

з дисципліни «Непроцедурне програмування»

Тема роботи «Визначення рекурсивних функцiй»

**Варіант №14**

Виконала студентка

групи КН-32

Козелько С.Є.

Перевірив:

Миколайчук Р.А.

**Київ – 2021**

**Мета роботи:** Набути досвiду визначення рекурсивних функцiй, використання механiзму зiставлення зi зразком i роботи з кортежами та списками.

**Завдання** Визначте вказанi функцiї в кожному з завдань: а) без застосування, б) з застосуванням вбудованих функцiй.

**1.14** Замiнити кожен n-й елемент списку вказаним значенням, напр. при n=2 та значеннi ’z’: "1234590"⇒ "1z3z5z0".

replaceAtN :: Int -> a -> [a] -> [a]

replaceAtN n c xs = countdown n xs where

countdown 1 (x:xs) = c:countdown n xs

countdown \_ [] = []

countdown m (x:xs) = x:countdown (m-1) xs

**Тести**

ghci> deleteAtN 3 'z' "aaa"

"aaz"

ghci> deleteAtN 2 'z' "aaaaaaaa"

"azazazaz"

ghci> deleteAtN 5 'b' "aaaa"

"aaaa"

ghci>

safeTail :: [a] -> [a]

safeTail [] = []

safeTail (\_:xs) = xs

replaceAtN2 :: Int -> Char -> String -> String

replaceAtN2 index replacement str = strHead ++ replacement : safeTail strAfter

where (strHead, strAfter) = splitAt index str

**Тести**

ghci> replaceAtN2 5 'b' "aaaaaaaa"

"aaaaabaa"

ghci> replaceAtN2 5 'b' "123456"

"12345b"

ghci> replaceAtN2 3 'z' "123456"

"123z56"

ghci>

**2.14** Знайти два простi числа, сума яких дорiвнює заданому парному N ≥ 2 (бiнарна проблема/гiпотеза Ґольдбаха).

dm :: Int -> [ Int ] -> [ Int ]

dm x xs = [ y | y <- xs , y `mod ` x /= 0]

da :: [ Int ] -> [ Int ]

da ( x : xs ) = x : da ( dm x xs )

primes :: [ Int ]

primes = da [2 ..]

goldbach n

| n `mod` 2 == 0 = [(a, b) | a <- takeWhile (<n) primes, b <- takeWhile (<n) primes, n == a + b]

| otherwise = []

**Тести**

ghci> goldbach 8

[(3,5),(5,3)]

ghci> goldbach 12

[(5,7),(7,5)]

ghci> goldbach 16

[(3,13),(5,11),(11,5),(13,3)]

ghci> goldbach 24

[(5,19),(7,17),(11,13),(13,11),(17,7),(19,5)]

ghci>

goldbach2 a = head $

filter (\(x,y) -> isPrime x && isPrime y) $

map (\e -> (e, a - e)) [1,3..a `div` 2]

where

factors a = filter (isFactor a) [2..a-1]

isFactor a b = a `mod` b == 0

isPrime a = null $ factors a

**Тести**

ghci> goldbach2 24

(1,23)

ghci> goldbach2 18

(1,17)

ghci> goldbach2 19

\*\*\* Exception: Prelude.head: empty list

ghci> goldbach2 16

(3,13)

ghci>

**Висновок:** В результаті виконання лабораторної роботи, ми набули досвіду визначення рекурсивних функцій, навчилися використовувати механізм зіставлення зі зразком та попрацювали з кортежами та списками.