

Робота № 2
Визначення рекурсивних функцій
Варіант № 12

Мета роботи

Набути досвіду визначення та використання функцій вищого порядку

Завдання

Визначте вказані функції в кожному з завдань:

- а) без застосування,
- б) з застосуванням вбудованих функцій вищого порядку.

1.12. Розділити список на дві частини при заданій довжині першої n , напр.
при $n=3$: "abcdefghik" \Rightarrow ("abc "defghik").

2.12. Перевірити гіпотезу Гольдбаха у вказаному діапазоні.

Код програми

```
module Main (main) where

import Data.List (unfoldr)
import Data.Maybe (listToMaybe)

main :: IO ()
main = do
    putStrLn "hello world"

{-
Визначте вказані функції в кожному з завдань: а) без застосування, б) з
застосуванням вбудованих функцій вищого порядку.

1.12 Розділити список на дві частини при заданій довжині першої n, напр.
при n=3: "abcdefghik" ⇒ ("abc "defghik")

2.12 Перевірити гіпотезу Гольдбаха у вказаному діапазоні.

-}

-- 1.12 -----

-- | Функція для розбиття списку на дві частини при заданому індексу розбиття -- n
-- | без використання ФВП
-- >> splitN 3 "abcdefghij"
-- > ("abc","defghij")
--
-- >> splitN 3 [0..9]
-- > ([0,1,2],[3,4,5,6,7,8,9])
splitN :: Int -> [a] -> ([a], [a])
splitN n list = splitAt n list

-- | Функція для розбиття списку на дві частини при заданому індексу розбиття -- n
-- | з використанням вбудованих функцій take та drop
-- | без використання ФВП
-- >> spN 3 "abcdefghij"
-- > ("abc","defghij")
--
-- >> spN 3 [0..9]
```

```
-- > ([0,1,2],[3,4,5,6,7,8,9])
spN n xs = (take n xs, drop n xs)

-- | Функція для індексування списку
-- | з використанням вбудованих функцій take та drop
-- >> index "abcdefghj"
-- > [('a',1),('b',2),('c',3),('d',4),('e',5),('f',6),('g',7),('h',8),('j',9)]
index :: [a] -> [(a, Int)]
index ls = zip ls [1..length ls]

-- | Функція для деіндексування списку
-- | з використанням вбудованих функцій take та drop
-- >> deindex [('a',1),('b',2),('c',3),('d',4),('e',5),('f',6),('g',7),('h',8),('j',9)]
-- > "abcdefghj"
deindex :: [(a, Int)] -> [a]
deindex [] = []
deindex ((i,x):xs) = i : deindex xs

-- | Функція для створення списку з перших n елементів заданого списку
-- >> take_ 4 "abcdefghj"
-- > "abcd"
take_ :: Int -> [a] -> [a]
take_ k _
  | k <= 0 = []
take_ _ [] = []
take_ n list = deindex (takeWhile (\ (x,y) -> y <= n) (index list))

-- | Функція для створення списку з усіх після n перших елементів заданого списку
-- >> drop_ 4 "abcdefghj"
-- > "efghj"
drop_ :: Int -> [a] -> [a]
drop_ k _
  | k <= 0 = []
drop_ _ [] = []
drop_ n list = deindex (dropWhile (\ (x,y) -> y <= n) (index list))

-- | Функція для розбиття списку на дві частини при заданому індексу розбиття -- n
-- | з використання ФВП
-- >> spN2 3 "abcdefghj"
-- > ("abc","defghj")
--
-- >> spN2 3 [0..9]
```

```
-- > ([0,1,2],[3,4,5,6,7,8,9])
spN2 n xs = (take_ n xs, drop_ n xs)

-- 2.12 -----

-- | Функція для отримання дільників числа
-- | з використання ФВП filter
-- >> factors 42
-- > [2,3,6,7,14,21]
factors a = filter (isFactor a) [2..a-1]

-- | Функція для перевірки на дільник цілого числа
-- >> isPr 11
-- > True
isFactor a b = a `mod` b == 0

-- | Функція для перевірки на просте число
-- >> isPr 11
-- > True
isPr 1 = False
isPr a = null $ factors a

-- | Функція для перевірки гіпотези Гольдбаха
-- | з використання ФВП filter та map
-- >> gb2 14
-- > (3,11)
gb2 :: Int -> (Int, Int)
gb2 a = head $ filter (\(x,y) -> isPr x && isPr y) $
    map (\c -> (c, a - c)) [3,5..a `div` 2]

-- | Функція для перевірки гіпотези Гольдбаха
-- | без використання ФВП
-- >> checkTheory 14 [2,3,5,7,11,13] [2,3,5,7,11,13]
-- > (3,11)
checkTheory :: Int -> [Int] -> [Int] -> (Int, Int)
checkTheory n ls (x:xs)
    | any_ ls n x /= (0, 0) = any_ ls n x
    | otherwise = checkTheory n ls xs
where
```

```
any_ [] n k = (0, 0)
any_ (x:xs) n k
  | x + k == n = (k, x)
  | otherwise = any_ xs n k

-- | Функція для перевірки коректності введеного числа,
-- | Якщо число задовольняє умови теорії, то повертає функції перевірки checkTheory
-- >> chTh 14 [2,3,5,7,11,13]
-- > (3,11)
chTh :: Int -> [Int] -> (Int, Int)
chTh n ls
  | n < 2 = (0, 0)
  | odd n = (0, 0)
  | otherwise = checkTheory n ls ls

-- | Функція для перевірки гіпотези Гольдбаха
-- | без використання ФВП
-- >> gb 14
-- > (3,11)
gb :: Int -> (Int, Int)
gb n = chTh n $ genPrimes n
where
  genPrimes n = [i | i <- [2..n], isPr i]
```

Приклад роботи

```
*Main> splitN 3 "abcdefghj"
("abc","defghj")
*Main> splitN 3 [0..9]
([0,1,2],[3,4,5,6,7,8,9])
*Main> spN 3 "abcdefghj"
("abc","defghj")
*Main> spN 3 [0..9]
([0,1,2],[3,4,5,6,7,8,9])
*Main> index "abcdefghj"
[('a',1),('b',2),('c',3),('d',4),('e',5),('f',6),('g',7),('h',8),('j',9)]
*Main> deindex [('a',1),('b',2),('c',3),('d',4),('e',5),('f',6),('g',7),('h',8),('j',9)]
"abcdefghj"
*Main> take_ 4 "abcdefghj"
"abcd"
*Main> drop_ 4 "abcdefghj"
"efghj"
*Main> spN2 3 "abcdefghj"
("abc","defghj")
*Main> spN2 3 [0..9]
([0,1,2],[3,4,5,6,7,8,9])
```

```
*Main> isFactor 12 4
True
*Main> factors 42
[2,3,6,7,14,21]
*Main> gb 14
(3,11)
*Main> gb2 14
(3,11)
*Main> ch
chTh      checkTheory
*Main> chTh 14 [2,3,5,7,11,13]
(3,11)
*Main> checkTheory 14 [2,3,5,7,11,13] [2,3,5,7,11,13]
(3,11)
```

Круть Катерина

ТІ-01

Лабораторна робота №2

Висновок: під час виконання цієї лабораторної роботи було набуто знання про ФВП та досвід роботи з ними. В результаті виконання роботи було завершено Роботу №2 Комп'ютерного парктикуму.