Робота № 4

**Розробка прикладних програм мовою Haskell**

Варіант № 12

**Мета роботи**

Ознайомитись з модульною органiзацiєю програм та засобами введення-

виведення. Набути досвiду компiляцiї Haskell-програм.

**Завдання**

Реалiзувати та скомпiлювати одну з програм, розроблених у лабораторнiй роботi No 2 для Вашого варiанта з введенням даних: а) з клавiатури, б) з файлу та виведенням результатiв: в) на екран, г) у файл. Запущена програма запитує користувача про спосiб введення та виведення, продовження чи припинення сеансу роботи.

Роздiлити список на двi частини при заданiй довжинi першої n, напр. при n=3: "abcdefghik"⇒ ("abc "defghik")

**Код програми**

module Main (main) where

import System.IO

import Control.Monad

import System.Directory

import Data.List.Split

import Data.List

createName name = "data/" ++ name ++ ".txt"

readFromFile = do

return readFile $ createName "input"

writeToFile inputData = do

writeFile (createName "output") inputData

promptLine = do

x <- System.IO.getLine

return x

splitN n xs = (take n xs, drop n xs)

splitLine inputData = splitOn " " inputData

convertToInt index = read index :: Int

toStr ls = "(" ++ fst ls ++ ", " ++ snd ls ++ ")"

splitData inputData = do

let ls = splitLine inputData

let ind\_ = ls !! 1

let str\_ = ls !! 0

let index = convertToInt ind\_

let ls = splitN index str\_

let kek = show ls

return kek

writeResult inputData = do

let res = splitData inputData

let result = res >>= show

writeToFile result

printToScreen inputData = do

let result = splitData inputData

result >>= putStrLn

readInput n = read n :: String

main :: IO ()

main = do

putStrLn "Choose input method: [1] Keyboard, [2] File"

inputMethod <- getLine

inputData <- case inputMethod of

"1" -> getLine

"2" -> readFile "data/input.txt"

\_ -> error "Invalid input method"

putStrLn "Choose output method: [1] Screen, [2] File"

outputMethod <- getLine

case outputMethod of

"1" -> printToScreen $ readInput inputData

"2" -> writeResult $ readInput inputData

-- "2" -> writeFile "data/output.txt" inputData

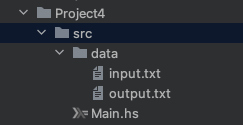
\_ -> error "Invalid input method"

putStrLn "Enter [1] to continue or [2] to exit"

continue <- getLine

when (continue /= "2") main

**Структура проекту**

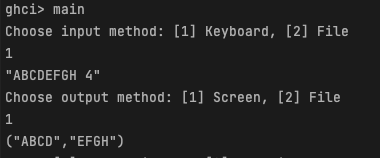
****

**Приклад роботи**

Головне меню

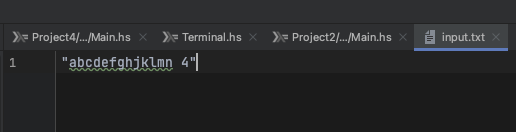
**image1.png**

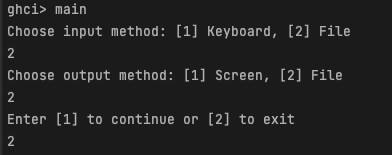
Читаємо та виводимо з консолі



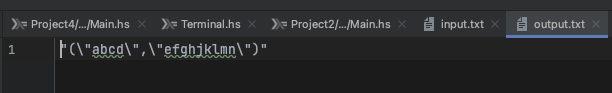
Читаємо з файлу та пишемо в файл

Файл для читання

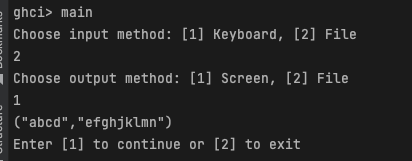




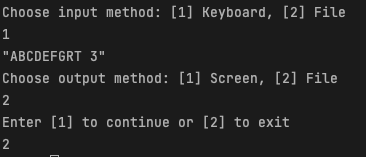
Результат – Файл для запису



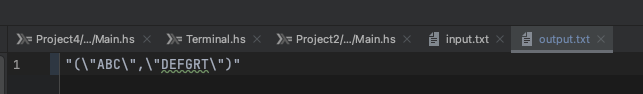
Читаємо з файлу та виводимо в консоль



Читаємо з консолі – ишемо в файл



Результат – Файл для запису



Вихід

image3.png