Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Уфимский университет науки и технологий»

Факультет информатики и робототехники

Кафедра вычислительной математики и кибернетики

Отчет по лабораторной работе №3

По дисциплине «Функциональное программирование»

Выполнил:

МО-321 Шемануев А.Е.

Проверил:

Ризванов Д. А.

Уфа, 2022

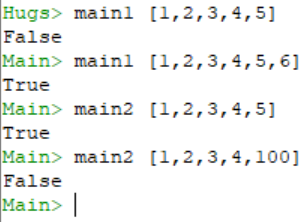
**Задание 1**

Используя функции any и all проверьте условия в интерпретаторе.

а) в списке есть числа, кратные 6,

б) все числа в списке меньше 100

|  |
| --- |
| a = any (\x -> mod x 6 == 0) a  main2 a = all (\x -> x < 100) a |

****

**Задание 2**

Используя функции map, filter, foldl, foldr, takeWhile, dropWhile, break, span, splitAt напишите функцию, решающую задачу.

а) по заданному списку строк все начальные прописные символы преобразовать в строчные

б) полученный список отсортировать по длине слов

|  |
| --- |
| import Debug.Trace  import Data.Char(digitToInt)  import Data.Char  import Data.List  fwToLower :: [String] -> [String]  sortStrings :: [String] -> [String]  filterB str s = filter (\a -> length a >= length s) str  filterS str s = filter (\a -> length a < length s) str  sortStrings [] = []  sortStrings (h:ts) = (sortStrings (filterS ts h)) ++ [h] ++ (sortStrings (filterB ts h))  fwToLower str = map (\(h:st) -> (toLower h):st) str  doThing :: [String] -> [String]  doThing a = sortStrings (fwToLower a)  main a = doThing a |

****

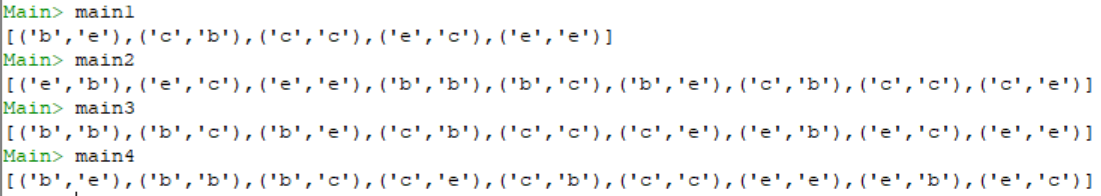
**Задание 3**

Для заданного отображения Р найти 

(можно использовать функции zip, unzip, zipWith в дополнение к предыдущим, также допускается использование генератора списков).



|  |
| --- |
| import Debug.Trace  import Data.List  p1 :: [(Char, Char)] -> [(Char, Char)]  p2 :: [(Char, Char)] -> [(Char, Char)]  p3 :: [(Char, Char)] -> [(Char, Char)]  p4 :: [(Char, Char)] -> [(Char, Char)]  p1 x = zip ((snd.unzip) x) ((fst.unzip) x)  p2 x= nub [(a,c) | (a,b) <- x, (b,c) <- x]  p3 x= nub [(a,c) | (a,b) <- p1 x, (b,c) <- x]  p4 x = nub [(a,b) | a <- (snd (unzip (p3 x))),  b <- (fst (unzip (p2 x)))]  main1 = p1 a  main2 = p2 a  main3 = p3 a  main4 = p4 a  a = [('e','b'), ('b','c'), ('c','c'), ('c','e'), ('e','e')] |

****

**Задание 4**

Дан список предикатов двух переменных:

Р1(х,у)=”x+y – четное число”,

P2 (х,у)=”x>y”,

P3 (х,у)=”x и y имеют одинаковые остатки от деления на 4”,

P4 (х,у)=”x+2y<8”,

P5 (х,у)=”max{x,y} – нечетное число”,

и список кортежей [(x,y)]. Написать функцию, имеющую аргументами эти два списка и решающую задачу. Рекомендуется использовать функции map, foldl, foldr, and, or.

Возвращает список логических значений выражения для каждого из предикатов.

|  |
| --- |
| p1 :: (Int, Int) -> Bool  p2 :: (Int, Int) -> Bool  p3 :: (Int, Int) -> Bool  p4 :: (Int, Int) -> Bool  p5 :: (Int, Int) -> Bool  p1 (x,y) = mod (x+y) 2 == 0  p2 (x,y) = x>y  p3 (x,y) = (mod x 4) == (mod y 4)  p4 (x,y) = x + 2 \* y < 8  p5 (x,y) = mod (max x y) 2 == 1  doThing :: [(Int, Int) -> Bool] -> [(Int, Int)] -> [Bool]  doThing [] h = []  doThing (f:tf) h = (and (map f h)) : (doThing tf h)  doThing1 :: [(Int, Int) -> Bool] -> [(Int, Int)] -> [Bool]  doThing1 t h = map (\a -> (and (map a h))) t  main x = doThing1 [p1, p2, p3, p4, p5] x |

****

**Задание 5**

Используя ранее изученные функции, написать функцию(-ии), решающую задачу.

Заданную строку разделить на список слов, считая разделителями пробелы или знаки пунктуации, их в список не включать.

|  |
| --- |
| import Data.List  main x = filter (\x -> not (x == "")) (splitOneOf " ,.!?" x) |

****

**Задание 6**

|  |
| --- |
| p a = [ [x,y] | x <- a, y <- a, y >= x, not(x==y)]  main = p [1,3,5,10,15] |

****