# Лабораторная работа 4

## Задание 2. Функции для работы с типом List. Для введенного ранее типа List определите следующие функции:

### 3) removeNegative, которая из списка целых (тип List Integer) удаляет отрицательные элементы.

data List a = Nil | Cons a (List a) deriving Show

headList (Cons x \_) = x

headList Nil = error "headList: empty list"

tailList (Cons \_ y) = y

tailList Nil = error "tailList: empty list"

removeNegative Nil = Nil

removeNegative (Cons x y) | (x < 0) = tailList (Cons x y)

| otherwise = (Cons x (removeNegative y))

Вывод:

ghci> removeNegative (Cons 1 (Cons (-3) (Cons 4 Nil)))

Cons 1 (Cons 4 Nil)

## Задание 8. (по выбору) Иерархия должностей в некоторой организации образует древовидную структуру. Каждый работник, однозначно характеризующийся уникальным именем, имеет несколько подчиненных. Определите тип данных, представляющий такую иерархию и опишите следующие функции:

### 1) getSubordinate, возвращающую список подчиненных указанного работника.

data Person = PersonName (String) [Person] deriving (Eq, Show)

getName (PersonName n \_) = n

getNames = map getName

getSubordinatePersons (PersonName \_ p) = p

finder ((PersonName n p) :xs) name | (n == name) = getSubordinatePersons (PersonName n p)

| otherwise = finder xs name

getSubordinate (PersonName n p) name = getNames (finder [(PersonName n p)] name)

Вывод:

ghci> getSubordinate (PersonName "a" [PersonName "b" [PersonName "c" [PersonName "d" [], PersonName "e" []], PersonName "f" []], PersonName "g" []]) "a"

["b","g"]

## Общий вывод результатов

