Федеральное государственное бюджетное образовательное учреждение

высшего образования

Уфимский государственный авиационный технический университет

Кафедра вычислительной математики и кибернетики

**Отчет по лабораторной работе №2**

**по дисциплине "** **Функционально-логическое программирование"**

Выполнила:

студентка группы ПРО-211

Кабирова Л.Р.

Проверила:

Усманова А.Р.

**Уфа 2017**

Вариант 1.

1. В задании описаны правила составления некоторого пароля. Написать функцию, проверяющую удовлетворяет ли заданная строка данным правилам.

а)первый символ не может быть цифрой;

б)пароль содержит не более двух заглавных букв;

в)сумма встречающихся цифр должна быть четна.

import Data.Char

fooB ab = zoo ab

zoo (x:xs) = isAlpha x

fooA xx = hoo xx 0

hoo \_ 3 = False

hoo [] 2 = True

hoo [] 1 = True

hoo [] 0 = True

hoo (x:xs) m | (x `elem` ['A'..'Z']) = hoo xs (m+1)

| otherwise = hoo xs m

ee = filter (`notElem` ['0'..'9'])

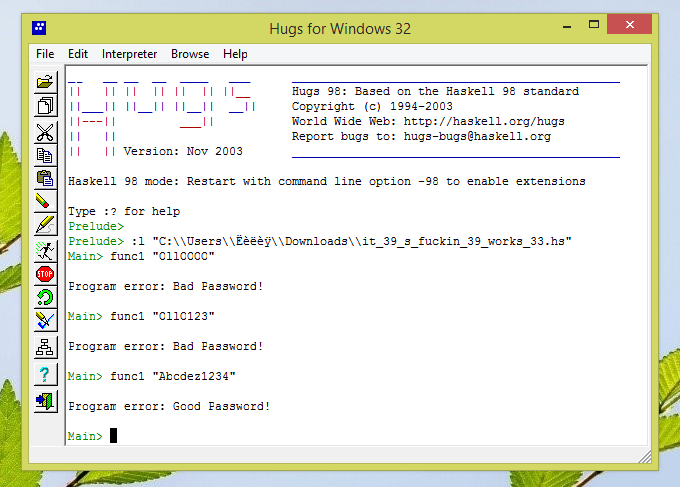
fooC ee = if (even (length ee) == True) then True

else False

func1 aa = if ((fooA aa == True) && (fooB aa == True) && (fooC aa == True)) then error "Good

Password!"

else error "Bad Password!"



2. Зашифруйте текст данного задания с помощью шифра Цезаря с ключом, соответствующим номеру варианта. Напишите функцию для расшифровки (можно одну функцию с дополнительным аргументом, задающим режим шифровки или расшифровки).

import Char

fooS [] \_ = []

fooS (x:xs) a | ((x `elem` [',','.','/','?']) && (a == 1)) = x:(fooS xs 1)

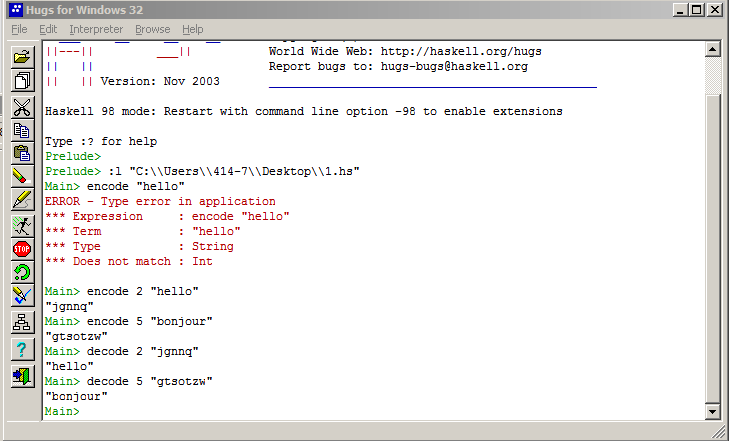
| ((x `elem` [',','.','/','?']) && (a == 2)) = x:(fooS xs 2)

| ((a == 1) && (isUpper x)) = (chr (((2 + (ord x) - (ord 'A')) `mod` 26) + (ord 'A'))):(fooS xs 1)

| ((a == 2) && (isUpper x)) = (chr (((2 - (ord x) - (ord 'A')) `mod` 26) + (ord 'A'))):(fooS xs 2)

| ((a == 1) && (isLower x)) = (chr (((2 + (ord x) - (ord 'a')) `mod` 26) + (ord 'a'))):(fooS xs 1)

| ((a == 2) && (isLower x)) = (chr (((2 - (ord x) - (ord 'a')) `mod` 26) + (ord 'a'))):(fooS xs 2)



3. Написать функцию, вычисляющую значение выражения и выводящую ответ в десятичном представлении, функция должна иметь не более двух аргументов строкового типа.

A1016 - DCCLIII

import Char

rometodec [x] = rtd x

rometodec (x:y:xs) = if (rtd x >= rtd y) then rometodec (y:xs) + rtd x

else rometodec (y:xs) - rtd x

hex2Dec [] = 0

hex2Dec (x:xs) = (hex2DecSymb x)\*(16^(length xs)) + hex2Dec xs

hex2DecSymb ss | (ss `elem` ['0'..'9']) = digitToInt ss

| (ss == 'A') = 10

| (ss == 'B') = 11

| (ss == 'C') = 12

| (ss == 'D') = 13

| (ss == 'E') = 14

| (ss == 'F') = 15

rtd r | (r == 'M') = 1000

| (r == 'D') = 500

| (r == 'C') = 100

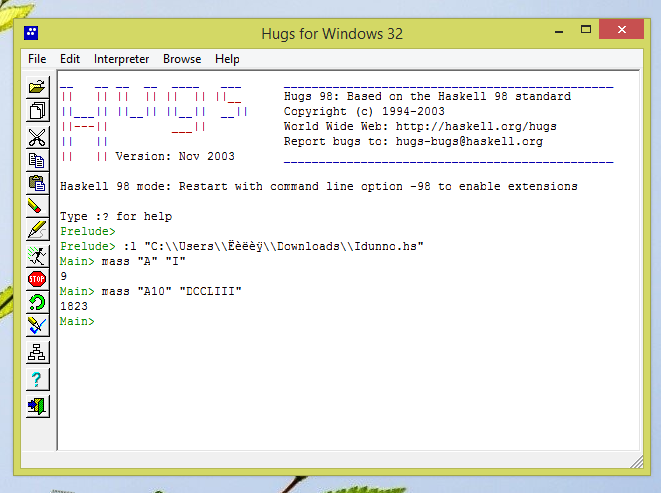
| (r == 'L') = 50

| (r == 'X') = 10

| (r == 'V') = 5

| (r == 'I') = 1

mass x y = hex2Dec x - rometodec y



4. Дано универсальное множество U={1,2,…,10} и некоторые его подмножества A,B,C. Найти множество D, заданное в варианте. (Реализовать множества списками).



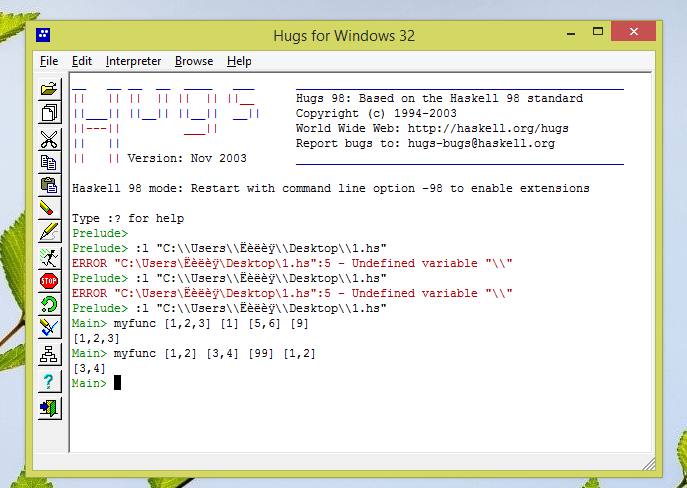
import List

mynot \_ [] = []

mynot l (d:ds) | (d `elem` l) = mynot l ds

| otherwise = d:(mynot l ds)

myfunc a b c d = (union a b ) \\ mynot c d



5. По введенному возрасту печатает статус человека: дошкольник(0-6), школьник(7-17), студент(18-22), работник(23-55), пенсионер(55-90). При значении возраста вне данных интервалов выводит соответствующие сообщения об ошибках.

myfun n | (n > 0)&&(n < 6) = error ("You are doshkol'nik")

| (n > 7)&&(n < 17) = error ("You are shkol'nik")

| (n > 18)&&(n < 22) = error ("You are student")

| (n > 23)&&(n < 55) = error ("You are worker")

| (n > 90)&&(n < 90) = error ("You are pensioner")

| otherwise = error ("I don't know who are you...")

