Program Design and Data Structures, 20.0 c

Course code: 1DL201, Report code: DL201, 67%, DAG, NML,
week: 44 - 02 Semester: Autumn 2017
week: 03 - 11 Semester: Spring 2018

LAB ASSIGNMENT 8

This information is not available in English. Now showing the Swedish version.

After this lab you should be able to ...

- Work with polymorphic expressions.
- Work with anonymous functions.
- Work with higher-order functions.

Instructions

- Read the slides for Lecture 17 (Polymorphism, Higher-Order Functions).
- Remember to write function specifications for all (non-anonymous) functions that you write. Do this *before* you write your Haskell code. Also remember to follow the other parts of our coding convention (for identifiers and indentation).

The task

PROBLEM 1

What is the type of the following expressions? (Try to remember/figure out the type yourself. Use GHCi's :t only to check your answer, or if you are stuck.)

1. head
2. tail
3. \x -> x
4. (,)
5. (:)
6. [[]]
7. tail [[]]
8. id : []
9. id id
10. head [id] "foo"

Which of these expressions are polymorphic?

PROBLEM 2

Write a function flipArguments that, when given a function f that takes two arguments, returns a function flipArguments f that takes the arguments in opposite order (and otherwise behaves as f).

Examples:

- flipArguments take "higher-order" 6 == "higher"
- flipArguments (-) 2 3 == 1

What is the type of flipArguments?

Try to write a function specification for flipArguments, as described in the coding convention.

PROBLEM 3

In mathematics, given a function $f : \mathbb{R} \to \mathbb{R}$, the difference quotient for the two points with x-coordinates x and $x + h$ is defined as $\frac{f(x+h)-f(x)}{h}$.

Write a function differenceQuotient that, when given a function f :: Double -> Double and two numbers x :: Double and h :: Double, computes the corresponding difference quotient. (You may assume that h is non-zero.)

Examples:

- differenceQuotient (\x -> x * x) 10.0 1.0 == 21.0
- differenceQuotient sin 0.0 0.1 == 0.998...

What is the type of differenceQuotient?

Try to write a function specification for differenceQuotient, as described in the coding convention.

PROBLEM 4

Recall that anonymous functions (lambda expressions) are expressions of the form \... -> ...

Write an anonymous function that behaves like the function flipArguments from Problem 2. In other words, find a lambda expression \... -> ... such that, e.g.,

- (\... -> ...) take "higher-order" 6 == "higher"
- (\... -> ...) (-) 2 3 == 1

(Do not declare any auxiliary functions to solve this problem. In particular, do *not* use your function flipArguments from Problem 2.)

WHEN YOU ARE DONE

When you are done with all problems (the explorations below are optional), raise your hand or approach an assistant to have your solution graded.

If you pass this lab at least 30 minutes early and other groups are still working on it, we ask you to **help one other group**. Do not simply share your solution with them, but try to understand the (partial) solutions that they have developed so far (which may be different from yours) and the difficulties that they have. Assist them in coming up with their own solutions. Once the group that you are helping completes the lab assignment, you may leave. (If it is still early, the group that received your help should then stay to help another group.)

EXPLORATIONS

If you have finished the lab exercises and are waiting to be graded, or if you wish to explore programming in Haskell further at any time, or want to get some extra practice, have a look at the Explorations page.

Exploration problems are optional and should only be attempted after other problems are completed.

You don't have to show these answers to the lab assistants for grading.