Functional Programming WS 2021 LVA 703025

Exercise Sheet 1, 5 points Deadline: Wednesday, October 13, 2021, 6am

- Mark your completed exercises in the OLAT course of the PS.
- You can use a template .hs-file that is provided on the proseminar page.
- Upload your modified .hs-file of Exercise 2 in OLAT.
- Your .hs-file should be compilable with ghci.

Exercise 1 *Haskell setup, no points*

Setup a working Haskell environment on your computer and get familiar with `ghci`. To do this follow these steps:

1. Install Haskell, e.g., via `ghcup`.[1]
2. Run `ghci` in a terminal and evaluate the expression `(5 + 2) * 3`.
3. Find and install a suitable text editor for your system to write and edit .hs files.[2] You can try one of the following free editors:
   - Atom[3] (Windows, macOS, Linux)
   - Notepad++[4] (Windows)
   - Gedit[5] (Windows, macOS, Linux)
4. Copy or enter the following code in your text editor and save it to a file called `myProgram.hs`. Be aware to use standard double-quotes (`"`), but neither two single-quotes (`''`) nor fancy-looking double-quotes (“ or ”).

```
hello :: String -> String
hello xs = "Hello " ++ xs
```

5. Load the file in `ghci` with the command `ghci myProgram.hs`
6. Evaluate the expression `hello "World"`
7. Make yourself familiar with `ghci`, in particular try the following commands:
   - `:?` – help
   - `:load name.hs` or `:l name.hs` – load Haskell script `name.hs`
   - `:reload` or `:r` – reload current Haskell script
   - `:edit` or `:e` – edit current Haskell script
   - `:set editor someEditor` – set `someEditor` as preferred editor

   Further investigate what happens if you type `h` and then the tabulator key, or `hel` and then the tabulator key.

You can find links to introductory material about `ghci`, the command line, etc. on the lecture homepage.[6]

[1] https://www.haskell.org/ghcup/

[2] Word processors like Microsoft Word, Apple pages,. . . are not text editors.

[3] https://atom.io/

[4] https://notepad-plus-plus.org/

[5] https://wiki.gnome.org/Apps/Gedit

[6] http://cl-informatik.uibk.ac.at/teaching/ws21/fp/ghc_setup.php

Solution 1

After the proseminar everyone should have access to a working Haskell-environment and be able to run ghci.

Exercise 2 *Writing simple functions* **5 p.**

1. Define a function `milesToKilometers m = ...` to convert miles into kilometers. (1 point)
2. Define a function `volume r = ...` to compute the volume of a sphere with radius `r`. (1 point)
3. Define a function `average x y = ...` that computes the average of two numbers `x` and `y`. (1 point)
4. Is `average (average x y) z` the average of three numbers `x`, `y` and `z`? (1 point)
5. Define a function `averageVolume r1 r2 = ...` that computes the average volume of two spheres having radius `r1` and `r2`, respectively. (1 point)

Solution 2

```haskell
milesToKilometers m = 1.609344 * m

-- use predefined "pi", or replace "pi" by numerical value 3.141592653589793
volume r = 4 / 3 * pi * r^3

average x y = (x + y) / 2

{-
 average (average x y) z =
 ((x+y) / 2 + z) / 2 =
 x/4 + y/4 + z/2
 which is not the same as the correct average
 x/3 + y/3 + z/3
-}

-- just reuse the previous functions
averageVolume r1 r2 = average (volume r1) (volume r2)
```