

Exercise 1.1 *GHC setup***0 p.**

Setup a working Haskell environment on your computer. To do this follow these steps:

1. Install the Haskell platform¹
2. Run `ghci` in a terminal and evaluate the expression `5 + 2 - 3`
3. Find a suitable text editor for your system to write and edit `.hs` files² You can use one of the following free editors:
 - Atom³ (Windows, macOS, Linux)
 - Notepad++⁴ (Windows)
 - Gedit⁵ (Windows, macOS, Linux)
4. Copy the following code to your text editor and save it to a file called `Example.hs`:

```
hello :: String -> String
hello xs = "Hello " ++ xs
```
5. Load the file in `ghci` with the command `ghci Example.hs`
6. Run the statement `hello "World"`

You can find links to introductory material about `ghci`, the command line, etc. on the lecture homepage⁶.

Exercise 1.2 *Evaluation Strategies***4 p.**

1. The function `subTwoY x y = x - (2 * y)` subtracts two times the value of `y` from `x`, and the function `double x = x + x` doubles the value of `x`. Stepwise evaluate with pen and paper the expression `subTwoY (double 5) 4` according to call-by-value, call-by-name and the lazy strategy until a normal form is reached, cf. part 2, 13–14. (2 points)
2. Consider the functions `foo x y = xPowYPowZ x y x` and `xPowYPowZ x y z = x ^ y ^ z`. Stepwise evaluate with pen and paper the expression `foo (1+1) 3` according to call-by-value/strict, call-by-name/non-strict and call-by-need/lazy until a normal form is reached. (2 points)

Exercise 1.3 *Evaluation Strategies***4 p.**

Haskell-functions are pure. Consider an extension of Haskell by a single non-pure constant `next_nat :: Integer`, which on first evaluation results in 0, then results in 1, in 2, etc. Define a single pure Haskell function `f` in a script, i.e., in the script `next_nat` must not appear, such that the evaluation of the expression `f next_nat` evaluates to at least three different values, depending on whether one uses call-by-value, call-by-name, left-to-right argument evaluation or right-to-left argument evaluation.

¹<https://www.haskell.org/platform/>

²Word 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>

⁶http://c1-informatik.uibk.ac.at/teaching/ws19/fp/ghc_setup.php