Computer Science C24
University of Toronto Scarborough

Homework Exercise # 3

Handing in and marking

For all exercises/assignments in this course, you need to submit your solutions to the pencil-and-paper questions on crowdmark and your solutions to the programming questions on MarkUs. Your pencil-and-paper solutions will be marked with respect to correctness, clarity, brevity, and readability. Your code will be marked with respect to correctness, efficiency, program design and coding style, clarity, and readability. This exercise counts for 5% of the course grade.

Question 1. Basic Haskell

Implement all functions in the module `E3`. Make sure your implementation passes the sanity-check tester (this is not a full tester!) `TestE3.hs` as well as HLint.

Question 2. Exploring Programming Languages – Python

In this question we implement Python alternatives of all Haskell functions in E3.hs. While doing so, we will familiarise ourselves with (or review) Python list comprehensions, as well as the functions map and reduce.

Your task is to implement every function in `e3.py`. Make sure your code passes flake8 and pylint checkers.

Question 3. Exploring Programming Languages – Java

In this question we implement the Object-Oriented, polymorphic equivalent of the Haskell functions `interleave` and `toPairs` in Java. Study the starter file `demo.Poly.java`, and implement the required classes `Pair` and `Interleaver` accordingly.

To make sure we learn something about Java in the process, we will not use constructs such as loops or recursion in this exercise. We will use Java **streams** instead.

Make sure all your classes/interfaces are **public**, all classes/interfaces and methods are named **exactly as specified**, and are in the **package poly**.

Finally, make sure your code passes the `checkstyle` tool, linked from the course website.

As always, make sure you are using the **correct versions of all tools**.

As always, make sure to fully test your code. The starter code provides **example** usage of the code you will develop, but **not proper testing** of the code — it is your responsibility to thoroughly test all code you submit.