

# Effective Functional Programming

## *Prelude*

### Assignment 0

## Hello, World!

Paul Downen

### 1 All About You (10 points)

**Exercise 1.1** (10 points). Write a short description on your previous experience with functional programming, and what you hope to get out of this class. Have you written a program in a functional language (like Haskell, OCaml, SML, Scheme, Clojure) before? If so, which one and what program? Or have you used a functional programming technique (such as MapReduce) in some other setting? How comfortable are you with functional programming basics (like recursion, higher-order functions, generics (a.k.a. parametric polymorphism), immutability, etc.)? At the end of the term, what do you hope to learn that you didn't know before?

### 2 Let's Get Hacking (10 points)

**Exercise 2.1** (10 points). Download and install the Haskell Platform (links can be found in the Resources page of the class site <https://classes.cs.uoregon.edu/18S/cis410afp/resources.html#code>). Get the Haskell Stack set up by running the command

```
> stack setup
```

When that completes successfully, open up the Haskell interpreter GHCi with the command

```
> stack repl
```

Now at the interpreter prompt, type in the following expression exactly as shown and press enter:

```
let motor sounds = sounds "Vroom! " in motor cycle
```

Describe in words what happens.

*Hint* 2.1. Holding the “control” key and pressing “c” will halt GHCi.