Advanced Functional Programming 2017–2018

EXAM

10 April, 13:30 – 16:30

Please read the following instructions carefully:

- Fill in your name and student number on all you work. Be prepared to identify yourself with your student card when you submit your exam.
- This is an open-book exam – you may consult any notes you wish, but you are forbidden from accessing any electronic material or online resources.
- A maximum of 90 points can be obtained by the questions of this exam, to be divided by 10 and incremented to yield the mark for the exam.

Please do not write in the space below.

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Bonus Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lenses</td>
<td>20</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>20</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Lenses

A lens between two types \(a\) and \(b\) is given access to an element of type \(b\) in some (larger) type \(a\):

\[
data a \rightarrow b = \text{Lens}\{\text{view} :: a \rightarrow b, \\
\quad \text{update} :: b \rightarrow a \rightarrow a\}
\]

Lenses are should adhere to the following two \textit{view}/\textit{update} laws:

\[
\forall (f :: a \rightarrow b) \quad (y :: a). \quad \text{update } f (\text{view } f \ y) \ y \equiv y
\]

\[
\forall (f :: a \rightarrow b) \quad (x :: b) \quad (y :: a). \quad \text{view } f (\text{update } f \ x \ y) \equiv x
\]

(a) (3 points) Define the ‘identity’ lens

\[
idLens :: a \rightarrow a
\]

and prove that it obeys both the laws above.

(b) (4 points) Define three lenses that access the address component of a person record, the street component of an address, and the street of a person:

\[
data \text{Person} = \text{Person}\{\text{name} :: \text{String}, \text{address} :: \text{Address}\}
\]

\[
data \text{Address} = \text{Address}\{\text{street} :: \text{Street}, \text{zipcode} :: \text{String}, \text{city} :: \text{String}\}
\]

\[
type \text{Street} = \text{String}
\]

\[
\text{addrOfPers} :: \text{Person} \rightarrow \text{Address}
\]

\[
\text{streetOfAddr} :: \text{Address} \rightarrow \text{Street}
\]

\[
\text{streetOfPers} :: \text{Person} \rightarrow \text{Street}
\]

The latter \text{streetOfPers} can be defined using the first two \text{addrOfPers} and \text{streetOfAddr}. Give the implementation for \text{streetOfPers} both using only the records selectors respectively using only \text{addrOfPers} and \text{streetOfAddr}.

(c) (3 points) Define a function that updates the substructure accessed by a lens according to the given function:

\[
\text{modify} :: (a \rightarrow b) \rightarrow (b \rightarrow b) \rightarrow (a \rightarrow a)
\]
(d) (4 points) Define a lens composition operator of the following type:

\[ \text{compose} :: (b \rightarrow c) \rightarrow (a \rightarrow b) \rightarrow (a \rightarrow c) \]

(e) (6 points) Prove using equational reasoning that if the two view/update laws stated above hold for both \( f \) and \( g \), then they also hold for \( \text{compose} \ f \ g \).