Programming Paradigms 2023
Session 7 : Declaring types and type classes

Preparing for the session

Hans Hüttel

31 October 2023

Where nothing else is mentioned, chapters and page numbers refer to *Programming in Haskell*.

The video podcast

You can watch the podcast on YouTube via the course page on Moodle.

Tuesday 31 October 2023 – Declaring types and type classes

The text is Chapter 8 of *Programming in Haskell*.

Learning goals for the session

- To understand how `type`, `newtype` and `data` declarations and how they differ.

- To be able to define functions over recursively defined data types using pattern matching.

- To understand how recursively defined data types can be used to implement the formation rules of a language.

- To understand the notion of term constructors and how they are used.

- To understand the principles of and be able to declare new instances of type classes.

- To understand how the above notions can be applied in the setting of a larger program.

How you should prepare before we meet on Tuesday

Before we meet, watch the podcast and read the text. You can do this in any order you like. Also see if you can solve the following two small discussion problems. We will talk about them in class.

1. Unary numerals consist of a finite sequence of `I`'s followed by a `Z`. The natural number n can be represented as n successive `I`'s and a `Z`, so e.g. 4 is represented in unary notation as `IIIIZ`. The natural number 0 is represented as `Z`.

   Define a recursive datatype Unary for unary numerals and use your type definition to define a function unary2int of type unary2int :: Unary −> Integer that finds the natural number represented by a given number. As an example, unary IIIIZ should give us 4

2. Use the declaration of the type Tree on page 97 to define a function least that finds the least element in a given binary tree.

   What should the type of least be?