Habib University

CS 200 Functional Data Structures
Spring 2016

Prelab 3: Introduction to Haskell
Due: 10 Feb, 2016

Problem 1 of 1. Introduction to Haskell

You have already read the Haskell book up to Chapter 5. Read up to Chapter 8. You may skip the following sections in Chapter 6: "Curried functions", "Only folds and horses", and "Function application with $".

Answer the following questions.

(a) Give an example signature of a function which receives another function as argument.

doubleInt :: (Int -> Int) -> Int -> Int

(b) Provide an example in which receiving a function as argument is beneficial for us.

Take the map example for example. If you have to modify the elements of the set, instead of taking ordinary arguments and modifying your map function accordingly, you can call function as an argument and modify the list.

(c) What does the `map` function do?

map function takes a list and a function as arguments and apply that function to every element of the list ; it ultimately produces a new list.

(d) What does the `filter` function do?

filter function takes a list and a predicate (returning boolean value) function as arguments and returns a new list containing those elements that satisfy the predicate.

(e) Write the output of the following Haskell statement.

```
map (max 5) [1..10]
```

[5,5,5,5,5,6,7,8,9,10]

(f) What is a module in Haskell?

A module contains relevant functions , types and typeclasses. As the name suggests, it is quite similar to the modular programming in imperative languages. You can import modules.

(g) Write the command to load a module in Haskell.

import <module name>

(h) How can we avoid name conflicts, i.e. a function with same name defined in two different modules?

There are two ways to do that. First : we can use to import <module> "hide" <hiding function>. Second:

import qualified <module> so for example you can call related function as Module.function instead of just function

(i) List the names of any two Haskell modules other than Data.List, Data.Char, Data.Map, and Data.Set.

Geometry.Sphere , Geometry. Cuboid

(j) What is an algebraic data type?

An algebraic data type is a composite data type that can be constructed using other types.

(k) Which keyword do we use to define our own data type?

data

(l) What is a value constructor?

= is a value constructor. It using | (or) can refer to how many different types can the function obtain.

(m) How can we associate different type classes with our defined types?

using |