

CS355: Programming Paradigms Lab

Lab 6: Haskell

September 30th, 2024

Q1. Try out the `commonWords` program from the class (check with some sample examples). Notice that we had started off by converting each letter in the text to lowercase and then separating out the words in the text. An alternative is to do the other way round: first separate out the words and then convert each letter to lowercase. The first method was expressed as `words . map toLower`. Give an expression for the second method and check that it works.

Q2. A positive integer is *perfect* if it equals the sum of all its factors, excluding the number itself. Using a list comprehension and a function `factors`, define a function `perfects :: Int -> [Int]` that returns the list of all perfect numbers up to a given limit. For example:

```
> perfects 500
[6,28,496]
```

Q3. Define a recursive function `insert :: a -> [a] -> [a]` that takes an element and a sorted list and inserts the element into its right place. For example:

```
> insert 4 [2,5,6]
[2,4,5,6]
```

Using `insert`, define a function `isort :: [a] -> [a]` that implements *insertion sort*, in which the empty list is already sorted, and any non-empty list is sorted by inserting its head into the list that results from sorting its tail.