1 COMP305 Week 4 Exercises

Assignment 2

Things to note:

- Code is now worth 60%
- Using Custom Data Types -¿ Explained next week
- Using an infinite list
- New Commenting Standard — Your comments should enable any student who has followed the lecture content only to fully understand what your code does and howit works.

Previous Week

1.1 Ex 2 - Hard

Write a function to return the maximum value of a list using recursion. Hint: you might need to keep things in the list

$max :: [Int] \rightarrow Int$

1.2 Ex 3

Write a function to return the maximum value of a list using higher order functions, using a fold method is highly recommended

$maxFO :: [Int] \rightarrow Int$

2 This Week

2.1 Ex 1 - Very Hard

Write a function which returns the **greatest difference** between the value of two consecutive elements in a list

$maxDiff :: [Int] \rightarrow Int$

Hint: This will use a lot of more complex higher order functions, using things such as

$zip :: [a] \rightarrow [b] \rightarrow [(a, b)]$

and

$foldl :: (b \rightarrow a \rightarrow b) \rightarrow b \rightarrow [a] \rightarrow b$

2.2 Ex 2

Using an infinite list and the `map` function, create a function that returns an array where each of the elements are incremented by one.

$addOne :: [Int] \to [Int]$

Hint: Use a zip

2.3 Ex 3 - Hard

Create a function that returns the reverse of an inputted list

type $prev = [Int]$

$rev :: prev \to [Int] \to [Int]$