CS 5035 (Fall 2016)

### Project 6. Modules (first attempt by Oct 17)

Based on chapter [7 of LYH](http://learnyouahaskell.com/modules). [Videos](https://sites.google.com/a/lclark.edu/drake/courses/pls/lesson-6-haskell-modules).

Write code to solve problem 4 from [here](https://www.shiftedup.com/2015/05/07/five-programming-problems-every-software-engineer-should-be-able-to-solve-in-less-than-1-hour).

Given a list of non negative integers, arrange and concatenate them so that they form the largest possible number. For example, given [50, 2, 1, 9], the largest possible number is 95021. Or, given [5, 54, 56] the result should be 56554. (Getting this test case right is the key.)

Use what you learned from Project 5 to convert the numbers to strings, sort and then concatenate the strings, and convert the result back to an Integer.

Use sortBy in your solution. sortBy has this type.

sortBy :: (a -> a -> Ordering) -> [a] -> [a]

The first argument is the comparison function sortBy uses to compare elements. In our case its type is.

String -> String -> Ordering

What comparison function should sortBy use in this problem?

sortBy is designed to order elements from smallest to largest. If compare x y = LT then x is placed to the left of y. In this problem that is counter-intuitive. In general we want the *larger* element to the left. Suppose your comparison function is called order. It will have to find the following.

For the first example:

order “9” “50” = order “9” “2” = order “9” “1” = LT  
order “50” “2” = order “50” “1” = LT  
order “2” “1” = LT

#### In the second example:

order “56” “5” = order “56” “54” = LT  
order “5” “54” = LT