

Activity 13

Haskell Practice on Lists Functions

References:

<code>++</code>	List concatenation operator
<code>:</code>	Append-head operator (“cons”)
<code>!!</code>	Indexing operator
<code>..</code>	Range-specifier for lists
<code>\\</code>	List-difference operator
<code><-</code>	List comprehension generator
	Single assignment operator in do-constr.

Data.List Prelude package: <http://hackage.haskell.org/package/base-4.12.0.0/docs/Data-List.html>

Implement the following functions in Haskell.

- a) write function `myLast` that takes a list and returns its last element
- b) write function `secondTwoLast` that takes a list and returns its second to last element.
- c) write function `elementAt` that takes a list and an index `k` and returns the `K`'th element of the list. Consider that the first element in the list is at index number 1.
- d) write function `myLength` that takes a list and returns its number of elements.
- e) write function `myReverse` that takes a list and returns its reverse.
- f) write function `middle` that takes a list and returns the list stripped of its first and last elements, or empty if the given list has less than 3 elements
- g) write function `isPalindrome` that takes a list and returns true/false whether the list is a palindrome or not. A palindrome can be read forward or backward; e.g. (x a m a x).