CSE1729 - Introduction to Programming | October 20, 2020

Laboratory Assignment 6

Objectives

More work with lists

Activities

1. The median value of a set of n numbers is the value that separates the half of higher values from the half of lower values in the set. The median can be found by arranging the values in the set in order and choosing the "middle" value. See the lecture slides for some sorting functions we will talk about tomorrow that will get any list in order. If there are an even number of values in the set, the median is described as the mean of the two middle values.
   (a) Define a SCHEME function named `(list-at l i)` which, given a list `l` and an index `i` returns the element in the list at index `i`. You should consider the first element of the original list to be at index 0.
   (b) Write a SCHEME function, named `list-median`, that takes a list of numbers as a parameter and returns the median value in the list.

2. Masahiko Fujiwara showed that 1458 is one of four positive integers (with the trivial case 1) which, when its digits are added together, produces a sum which, when multiplied by its reversal, yields the original number:

$$\begin{aligned} 1+4+5+8 &= 18 \\ 18\times 81 &= 1458 \end{aligned}$$

   (a) Define a function named `(explode x)` which converts any positive integer x to a list of single-digit integers (use division by 10 and the floor function). For instance, the expression `(explode 12345)` would return `'(1 2 3 4 5)`. You may only use `cons`, `car`, `cdr`, functions you wrote in lab or are contained in the lecture slides. You may not use string or char functions.
   (b) Define a SCHEME function named `(implode l)` which takes a list of digits in base 10 and converts them to an integer (the inverse operation of `explode`). You may only use `cons`, `car`, `cdr`, functions you wrote in lab or are contained in the lecture slides. You may not use string or char functions.
   (c) Write a function named `(has-property x)` which accepts an integer x and returns `#t` if x has this property and `#f` otherwise. Test your function with the integer 1458. Feel free to use the `sum-list` and `reverse` functions from lecture.
   (d) We are looking for a certain quantity of integers which have a particular property. It would be helpful if we wrote a higher order function which could take a function which generated a sequence (in this case, the sequence of positive integers) and a function which tested for the property in which we are interested as well as the integer n and returned the n^{th} value in that

sequence which satisfied the property (i.e. the test function returns true when passed that value). Write a function, named `find`, which takes three parameters:

- `sequence`, a function that takes a positive integer as the index and returns the value at that index
- `test`, a function that returns true or false
- `n`, a positive integer

`find` returns the n^{th} value in sequence for which the test function returns true (`#t`) when passed a value in sequence. (sequence k) should return the k^{th} element of the sequence (where k is a positive integer); `(test x)` should return `#t` if the property holds for `x`, `#f` otherwise.

(e) Use the `find` function to identify two other integers with this property. To do this, define a SCHEME function, named `(fujiwara n)`, which takes an integer n and returns the n^{th} number with the property he described. *Hint: 1 is the first integer with this property and 1458 is the third. You are looking for the second and fourth integers with this property.*