CS5001 Assignment 10

Programming Language: ISL with Lambda

Due Date: Thursday 4/5 @ 10:00pm

Problem 1.
Design a function called `list->chunks` that will consume a non-empty list and a positive integer `n` and returns a list of lists of size `n`, each list of size `n` is a sub-sequence of the input. You may assume that the length of the input list is divisible by `n`

```
> (list->chunks (list "a" "b" "c" "d" "e" "f") 3)
  (list (list "a" "b" "c")
        (list "d" "e" "f"))
```

Problem 2.
DNA is often modelled by strings of the characters A, C, G and T. They are very long and so, often need to be compressed. A simple way to do this is to replace all substrings of identical consecutive characters with the character followed by the length of the substring. These substrings must be as long as possible. For example, the run-length encoding of the string:

`"AAGCCCCTTAAAAAAAAAA"` is the string `"A2G1C4T2A10"`.

This is the unique run-length encoding – something like `"A2G1C4T2A4A6"` is not valid. Use generative recursion to write a function called `dna-encode` that consumes a DNA string and produces its run-length encoding.

Here are a couple of examples

```
> (dna-encode "AAGCCCTTAAAAAAAAAA")
    "A2G1C3T2A10"

 > (dna-encode "")
    ""
```

You may assume that the input string consists of capital letters only.

Also design the function `dna-decode` that will perform the opposite operation, i.e., given a run-length encoding return the full string.

```
> (dna-decode "A2G1C3T2A10")
    "AAGCCCTTAAAAAAAAAA"

> (dna-decode "")
    ""
```

Problem 3.
Design a function that implements Bubble_sort. Your function should consume a list of numbers and produce the same list of numbers but in sorted order, from smallest to largest, using the algorithm for Bubble sort.