CS2205: Theory of Computation
Problem Class III

1. Evaluate the following list expressions. They make use of the primitive operations, cons, isEmpty, head and tail, covered in the lectures. Write 'undefined' if you think an expression is illegal.

   (a) head(cons(a, [ ]))
   (b) tail(cons(a, [ ]))
   (c) head(tail([a, b, c]))
   (d) head(head([a, b, c]))
   (e) cons(a, cons(b, cons(c, cons(d, [ ]))))
   (f) cons([a], cons([b], cons([c], cons([d], [ ]))))
   (g) cons([a], cons([b], cons([c], [d])))
   (h) isEmpty(tail([a]))
   (i) isEmpty(tail([a, [ ]]))
   (j) head(tail([a, [ ]]))
   (k) cons([ ], [ ])
   (l) tail(tail(tail(tail([a, b, c]))))
   (m) cons(isEmpty([a, b, c]), [**false**])
   (n) cons(tail([b, c, d]), [b, c, d])

2. Write *recursive* DECAFF algorithms using cons, isEmpty, head, tail and, where necessary, length & append to solve the following problems.

   Don't just guess! Think about the recursive structure of lists. Think about your base case or base cases. For the recursive case(s), ensure the parameters are getting closer to the base case. In the recursive case, think hard about how to stick together the different parts of the answer. Desk-check your solutions by 'dry-running' them on a well-chosen set of different test cases.

   (a)

   Problem III.1.
   Parameters: *A finite-length list of integers, L.*
   Returns: *The sum of the integers in L.*

   (b)

   Problem III.2.
   Parameters: *A non-empty finite-length lists, L, and an integer i, $0 \le i < \text{length}(L)$.*
   Returns: *The ith element of L (where the head of L is taken to be the 0th element).*

   (c)

   Problem III.3.
   Parameters: *A finite-length list of integers, L.*
   Returns: *A list of all the even integers in L.*

   (d)

   Problem III.4.
   Parameters: *A non-empty finite-length list, L.*
   Returns: *The last element of L.*

   (e)

   Problem III.5.
   Parameters: *A non-empty finite-length list, L.*
   Returns: *All but the last element of L.*

   (f)

   Problem III.6.
   Parameters: *A finite-length list, L.*
   Returns: *The reverse of L.*

   (g)

   Problem III.7.
   Parameters: *Two finite- and equal-length lists of integers, L_1 and L_2.*
   Returns: *A list containing the sum of corresponding elements from L_1 and L_2.*

   (h)

   Problem III.8.

Parameters: *Two finite- but possibly different length lists of integers, L_1 and L_2.*

Returns: *A list containing the sum of corresponding elements from L_1 and L_2. If one of the lists is longer, then its extra elements are included as the tail of the answer.*

(i)

Problem III.9.

Parameters: *A finite-length list, L, and an integer, n, $n \geq 0$.*

Returns: *The first n elements of L (or the whole of L if n is greater than the length of L).*

(j)

Problem III.10.

Parameters: *Two finite- but possibly different length lists of integers, L_1 and L_2, already sorted in ascending order.*

Returns: *A merged list, containing all the elements of L_1 and all the elements of L_2, in ascending order.*