Homework 4 (25pts)

Solution

1. Translate the following Scheme code to equivalent function definitions in ML.

   (a)
   ```
   (define sum1 (lambda (x)
                  (cond ((null? x) 0)
                        ((cons? x) (+ (car x) (sum1 (cdr x)))))))
   ```

   Solution:

   ```
   fun sum1(nil) = 0
     | sum1(x1::x2) = x1 + sum1(x2);
   ```

   (b)
   ```
   (define sum2 (lambda (x)
                  (cond ((null? x) 0)
                        ((number? x) x)
                        ((cons? x) (+ (sum2 (car x)) (sum2 (cdr x))))
                        (else 0))))
   ```

   Solution:

   ```
   datatype ValType = Nil | Number of int | Cons of ValType * ValType
   fun sum2(Nil) = 0
     |  sum2(Number(x)) = x
     | sum2(Cons(x1,x2)) = sum2(x1) + sum2(x2);
   ```

2. Use loops and assignments to implement a ML function which takes a single list of integers and then return the number of positive integers (i.e., integers that are > 0) inside the list.

   ```
   fun count (x) = let  val px = ref x; val res=ref 0 in
         while (not (null(!px))) do (px := tl(!px); res:= (!res)+1); !res end;
   ```