Daniel Frey

CS 3160-001

Assignment 1

9/5/18

1. The biggest difference between a compiled language and an interpreted language is that compiled can be directly executed by the computer, while the interpreted must be translated at runtime. For compiled, the translation time is separate from the run time. For interpreted, the translation time is included in the run time.
2. Which of the following defines or correctly categorizes the halting problem?  
   B. The halting problem is not solvable (i.e., not computable by computers).
3. Identify as partial or total function on the integers. If partial, indicate where defined and undefined.  
   A. f(x) = if (x + 2) > 5 then x + 2 else x/0
   1. This function is a partial function. It is defined for integers where x > 3 (4+). It is not defined for integers where x ≤ 3, since dividing by zero is undefined.

B. f(x) = if x < 0 then 1 else f(x – 2)

* 1. This function is a total function, it covers all values for x.

1. Function that takes in a list and duplicates the first element.  
   (defun duplicate-first-element (lst)  
    “This function takes in a list and duplicates the first element.”

(cons (car lst) lst)

)

(print (duplicate-first-element ‘(1 2 3)) 🡪 (1 1 2 3)

1. Function that takes in arbitrary parameter x, and returns the product of all numeric values in x.  
   (defun product (x)  
    "This function takes in an arbitrary parameter x, and returns the product of all numeric values contained within x."  
    (cond  
    ((consp x) (\* (product (car x)) (product (cdr x))))  
    ((numberp x) (\* x 1))  
    (t 1)  
    )  
   )  
     
   (print (product ‘x)) 🡪 1  
   (print (product ‘(x 5))) 🡪 5  
   (print (product ‘(2 2) 3)) 🡪 12  
   (print (product ‘((a 3) (2 1)))) 🡪 6
2. Function that takes in a list input and returns the length of the list.  
   (defun list-len (lst)  
    "This function takes in a list and returns the length."  
    (if lst   
    (1+ (list-len (cdr lst)))   
    0  
    )  
   )   
     
   (print (list-len '(1 2 3))) 🡪 3  
   (print (list-len '((1 2 3)(4 5)))) 🡪 2  
   (print (list-len '((1 2 3)(4 5)(a (b) c)))) 🡪 3
3. Function to get the last atom in a list.  
   (defun last-atom (lst)  
    "This function returns the last atom in a list."  
    (setq x (list-length lst))  
    (loop while (> x 1) do  
    (setq lst (cdr lst))  
    (setq x (list-length lst))  
    )  
    (car lst)  
   )  
     
   (print (last-atom '(a b c d e f))) 🡪 F