**Structures and Interpretation of Computer Program**

**Exercise Chapter 1.2 Name:** Wan Huzaifah bin Wan Azhar

**Exercise 1.2.2 Tree Recursion**

EX 1.11a Recursive f(n)

(define (f n)  
 (cond ((< n 3) n)  
 ((> n 3)   
 (+ (f (- n 1))  
 (\* 2 (f (- n 2)))  
 (\* 3 (f (- n 3)))))  
 (else 0)))

Else 0 is important here because author did not stated what value is returned if n = 3. I could have leave n = 3 and make the algorithm produce an error if it tried to compute it but I think 0 if n = 3 is more correct.

EX1.11b Iterative f(n)

(define (f n)

(cond ((= n 3) 0)

((< n 3) n)

(else (f-iter 0 2 1 n))

)

)

(define (f-iter a b c n)

(if (= n 3)

a

(f-iter (+ a (\* 2 b) (\* 3 c)) a b (- n 1))

)

)

EX1.12 Finding element in Pascal’s triangle

; r stands for row and c stands for column

(define (pasc r c)

(cond

((or (= r 0)(= r 1)) 1)

((= c 0) 1)

((= r c) 1)

((< c r)

(+ (pasc (- r 1) (- c 1)) (pasc (- r 1) c))

)

)

)