**Structures and Interpretation of Computer Program**

**Exercise Chapter 2.1 Name:** Wan Huzaifah bin Wan Azhar

**Exercise 2.1.3 What is meant by data**

2. Verify (car (cons x y)) yields x for any obj x and y

(car (cons x y))

(car (lambda (m) (m x y)))

((lambda (m) (m x y)) (lambda (p q) p))

(lambda (p q) (p)) (x y)

x

1. Cdr equiavalent

(define (cdr z)

(z (lambda (p q) q)))



(define (cons a b)

(\* (expt 2 a) (expt 3 b)))

;Return the highest power that can divide the number

(define (highest-pow x powr)

(define (highest-pow-iter z n)

(cond ((= 0 (remainder z powr))

(highest-pow-iter (/ z powr) (+ 1 n)))

(else n)))

(highest-pow-iter x 0))

(define (cdr y)

(highest-pow y 3))

(define (car x)

(highest-pow x 2))

(define test-var (cons 5 4))

(display (car test-var))

(newline)

(display (cdr test-var))



(define zero (lambda (f) (lambda (x) x)))

(define (add-1 n)

(lambda (f) (lambda (x) (f ((n f) x)

))))

(define one

(lambda (f) (lambda (x) (f x)))

)

(define two

(lambda (f) (lambda (x) (f (f x))))

)

(define (add g n)

(lambda (f) (lambda (x) ((g f) ((n f) x)))))