1) Consider the following Scheme definition:

```
>> (define (test x y) (if (= x 3) 1 y))
```

Write the output of the following call to this function

```
>> (test 3 (/ 5 0))
```

a) assuming normal-order evaluation is used (5 pts):

1

b) assuming applicative-order evaluation is used (5 pts):

Error: Division by zero!

2) Write the output of the following programs:

a) >>> (let ((a 3) (b 4))

(let ((a 5) (b a))

(let ((a b) (c a))

(+ a b c)) ))

Output (5 pts):

11 ;(3+3+5)

b) >>> (let ((a 3) (b 4))

(let* ((a 5) (b a))

(let* ((a b) (c a))

(+ a b c)) ))

Output (5 pts):

15 ;(5+5+5)