arsDigita University

Structure and Interpretation of Computer Programs
October 2000

Quiz 1
Scheme Basics

Before starting, please write your name in the first blank below, and *optionally* a guess as to how well you think you will do in the second. Start the quiz only when instructed to do so. You may use any written resources you wish, but you may not consult another student, nor use a computer, nor a calculator. You will have one hour to finish this quiz, at which point please close the document, and *optionally* re-assess your anticipated grade in the third blank below. Please give your tests to the staff as you leave. Your actual grade will not be affected by your self-assessment, nor by opting out of self-assessment.

Name: ______________________________

Optional, expected grade (percent correct) before taking quiz: __________

Optional, expected grade (percent correct) after taking quiz: __________

P1: ______ P2: ______ P3: ______ P4: ______ Total: ______

Problem 1 What will Scheme print in response to the following statements? Assume that they are each evaluated in order in a single Scheme buffer. Write your answer below each statement. You may write "procedure" if a procedure would be returned, or "error" if an error message would be returned. (This problem spans two pages.)

```
(define x 2)
```

```
x
```

```
(x)
```

```
(define (y) (* x 2))
```

```
y
```

```
(y)
```

```
(define (a) (lambda (x) (+ x 1)))
```

```
a
```

```
(a)
```

```
(let ((x 1)
      (y 2)
      (z (+ x 4)))
   (+ x y z))
```

```
(define (if a b c) (+ a b c))
```

```
(if 2 3 4)
```

Problem 2 Write a function called `new-add` which returns the sum of two numbers. Do not use the internal functions `+` and `-`, but instead, use `inc` (an already defined Scheme procedure which takes one argument and returns the sum of that argument and 1) and `dec` (a similar procedure which returns the sum of its argument and -1).

Is your procedure recursive? ____________________

Is your procedure tail-recursive? ____________________

Is the process it generates recursive or iterative? ____________________

Problem 3 Assume that we have defined `sum` and `square` as follows:

```
(define (sum term a next b)
   (if (> a b)
       0
       (+ (term a)
          (sum term (next a) next b))))

(define (square x) (* x x))
```

Determine the order of growth in time for the following functions using Θ notation. (Hint: you will only need to use one or more of the following for your answers: $\Theta(1)$, $\Theta(\log n)$, $\Theta(n)$, $\Theta(n^2)$ and $\Theta(2^n)$. All classes might not be used.) Write your answer in the blanks to the right of each function.

```
(define (integrate a b f dx)
   (sum (lambda (x) (* (f x) dx))
        a
        (lambda (x) (+ x dx))
        b))
```

```
(define (number-of-bits-in n)
   (if (< n 2)
       1
       (+ 1 (number-of-bits-in (/ n 2)))))
```

```
(define (times-5 x)
   (* x 5))
```

```
(define (exp a b)
   (cond ((< b 0) (error "Oops! b cannot be negative"))
         ((= b 0) 1)
         (else (if (odd? b)
                   (* a (exp a (- b 1)))
                   (square (exp a (/ b 2)))))))
```

```
(define (sum-of-squares x y)
   (+ (square x)
      (square y)))
```

```
(define (triangle-sum n)
   (sum (lambda (m) (sum (lambda (x) x) 1 inc m))
        1
        inc
        n))
```

Problem 4 Write a procedure `power-close-to` that takes two non-zero positive integers (`b` and `n`) as arguments and returns the smallest power of `b` that is greater than `n`. That is, it should return the smallest integer i such that $b^i > n$. You may use the Scheme procedure `(expt b i)` which raises `b` to the power `i`.

Does your procedure generate an iterative process or a recursive process? ____________________