Midterm test 1 CIS280, Spring 2001, Saunders

Name: Solutions

1 21 points

21 = `(- (* 8  3) (/ 12 4))`

24 = `(+ (* (- 9 7) 2 5) 4)`

error `(* (- (+ (4 5) 3) 2))`

```
(define a 3)
(define b (- a 1))
```

#t = `(> a b)`

#f = `(and (= a b) (/ b (- a 3)))`

2 = `(or ( = a b) (+ 1 1) (< b a))`

3 = `(if (and (< b a) (< a (* b b))) a b)`

2 12 points

15 = `(fun1 2 3)`

15 = `(fun1 x 3)`

17 = `(fun2 3 4)`

3 15 points

Part A.

```
(define smoother
  (lambda (f d) ; the 2 arguments to smoother are a function and a distance.
        (lambda (x) (/ (+ (f (- x d)) (f x) (f (+ x d)))
                               3 ))
  )
)
```

Part B. ((smoother sin 0.01) 2)

4 20 points

Part A. 3 = `(f 4)`, 3 = `(f 6)`, 4 = `(f 8)`, 101 = `(f (expt 2 100))`, 100 = `(f (- (expt 2 100) 1))`.

Part B.

```
(define f (lambda (n) (f+ n 0)))
(define f+
  (lambda (n s)
          (if (< n 1)
              s
              (f+ (quotient n 2) (+ s 1))
              )))
```

5 12 points

Part A.

```
(define sums
  (lambda (n) (sums-with-max n n)) )
(define sums-with-max
  (lambda (n m)
          (if (or (zero? n) (= m 1))
              1
              (+ (sums-with-max n (- m 1))
                 (sums-with-max (- n m) (min m (- n m)))
                 ))))
```

Part B. tree recursive.

6 20 points

Part A. 2 sec: expect `(expt-v1 1.01 2048)` to take?
1024 sec : expect `(expt-v1 1.01 1048576)` to take?
1.1 sec : expect `(expt-v2 1.001 2048)` to take?
2 sec : expect `(expt-v2 1.001 1048576)` to take?
0.01 sec : expect `(expt-v2 1.01 1024)` to take on computer X? [about 1000 steps for expt-v1, only 10 steps for expt-v2. each step of expt-v2 is a little longer so maybe a little more than 0.01, at most 0.02.]

Part B.

```
(define (expt-v1 b e)
  (cond ((zero? e) 1)
        (else (* b (expt-v1 b (- e 1)))) ) )
```

Part C.

```
(define (expt-v2 b e)
  (if (zero? e) 1
      (if (even? e)
          (square (expt-v2 b (quotient e 2)))
          (* b (expt-v2 b (- e 1))) ) ))
```