1.

```
(/  ( tan (- x  (* y (cos y))))
    (sin (+ x y)))
```

+10

2.

a) `(define (big2 a b) (if (>= a b) a b))`

b)
```
(define (big3 a b c)
    (big2 (big2 a b) c))
```

c)
```
(define (big4 a b c d)
    (big2 (big3 a b c) (d)))
```

+15

3.

a)
```
(( lambda (x y z)
    (* (- x y) z))
  (* y 5) (+ x z) (+ x y))
```

b) $(x-y)z = (10-5)5 = 25$

4.
```
(define (f n)
    (cond ((= n 0) 8)
          ((= n 1) 1)
          ((= n 2) 9)
          ((odd? n) (+ (f (- n 1)) (- (f (- n 2))) (f (- n 3))))
          (else     (+ (f (- n 1)) (f (- n 2)) (- (f (- n 3)))))
```

```
(define (f n)
    (define (iter i n1 n2 n3 n4)
        (cond
```

4. b) (define (f n)
      (define (iter i n1 n2 n3 res)
        (if (> i n)
            res
            (if (odd? i) (iter (+ i 1) res n1 n2
                               (+ n1 (- n2) n3))
                (iter (+ i 1) res n1 n2
                      (+ n1 n2 (- n3))))))
      (cond ((= n 1) 8)
            ((= n 2) 1)
            ((= n 3) 4)
            (else (iter 4 8 1 4 0)))))

5. a) $(a <*> b) <*> c = (a*b + a+b) <*> c$

      $= a*b*c + ac + bc + a*b + a+b + c$

   $a <*> (b <*> c) = a <*> (b*c + b + c)$

      $= a*b*c + ab + ac + b*c + a+b+c$

   these

   ⇒ Since two expression are equivalent, the operator is associative

   b) (define (fast-bn b n) ;; $a <*> a <*> a ... <*> a = (a+1)^n - 1$
        ;; (- (expt (+ b 1) n) 1))
        (- (expt (+ b 1) n) 1))

   (define (expt b n)
     (define (iter base c res)
       (if (= c 0) res
           (if (odd? c) (iter (* base base) (- c 1)
                             (* res base
                         (iter (* base base) (/ c 2)
                               res)))))

   (iter b n 1)) 第二頁

6. a)
```scheme
(define (filtered-sum filter a b)
    (define (iter i res)
        (if (> i b)
            res
            (if (filter i)
                (iter (+ i 1) (+ res i))
                (iter (+ i 1) res))))
    (iter a 0))
```

+15

b)
```scheme
(define (modsum a b m)
    (filter-sum a
                b
                (lambda (x) (= (modulo x m) 0))))
```

第一頁 (+ lower