1)

a) `10` → `Value: 10`

b)`(+ 5 3 4)` → `Value: 12`

c)`(- 9 1)` → `Value: 8`

d)`(/ 6 2)` → `Value: 3`

e)`(+ (* 2 4) (- 4 6))` → `Value: 6`

f)`(define a 3)` → `Value: 3`

g)`(define b (+ a 1))` → `Value: 3`

h)`(+ a b (* a b))` → `Value: 19`

i)`(= a b)` → `Value: f`

j)`(if (and (> b a) (< b (* a b))) b a)` → `Value: 4`

k)`(cond ((= a 4) 6) ((= b 4) (+ 6 7 a)) (else 25))` → `Value: 16`

l)`(+ 2 (if (> b a) b a))` → `Value: 6`

m)`(* (cond ((> a b) a) ((< a b) b) (else -1)) (+ a 1))` → `Value: 16`

2) Translate $\frac{5+4+(2-(3-(6+\frac{4}{5})))}{3\cdot(6-2)(2-7)}$ in to prefix form.

$5+4+(2-(3-(6+\frac{4}{5})))$ → `(+ 5 (+ 4 (- 2 (- 3 (+ 6 4/5)))))`

$3\cdot(6-2)(2-7)$ → `(* 3 (* (- 6 2) (- 2 7)))`

$\therefore \frac{5+4+(2-(3-(6+\frac{4}{5})))}{3\cdot(6-2)(2-7)}$ ⇒ `(/ (+ 5 (+ 4 (- 2 (- 3 (+ 6 4/5))))) (* 3 (* (- 6 2) (- 2 7))))`

3) Define a procedure that takes three numbers as arguments and returns the sum of the squares of the two larger numbers.

(define (gr-three a b c) (cond ((= a b c) (sq-sum a b)) ((and (= a b) = a c)) (if (= b c) (sq-sum a b) (sq-sum a c))) ((and (= b c) (= b a)) (if (= c a) (sq-sum b c) (sq-sum b a))) ((and (= c a) (= c b)) (if (= a b) (sq-sum c a) (sq-sum c b)))

))`(define (sq-sum x y) (+ (* x x) (* y y)))`

`(define (gr-three a b c) (cond ((= a b c) (sq-sum a b)) ((and (= a b) = a c)) (if (= b c) (sq-sum a b) (sq-sum a c))) ((and (= b c) (= b a)) (if (= c a) (sq-sum b c) (sq-sum b a))) ((and (= c a) (= c b)) (if (= a b) (sq-sum c a) (sq-sum c b)))`

`) )`

`4) Observe that our model of evaluation allows for combinations whose operators are compound expressions. Use this observation to describe the behavior of the following procedure:`

`texttt(define (a-plus-abs-b a b) ((if (> b 0) + -) a b))`