TEL AVIV UNIVERSITY
Department of Computer Science
0368.1105 – Extended Introduction to Computer Science
Fall Semester, 2000/2001

**Homework 1, October 29, 2000**

This homework is due by 18:00 on Sunday, Nov 5, at the hand-in boxes near room 005 of the Schreiber building. Write your name, id and the **Tirgul group number** on top of the first page. Keep a copy of everything your turn in!!

## Finger exercises

What would a Scheme evaluator print in response to the following expressions? First, try to find out the answers *without* using the evaluator. Then, check your answers using the evaluator and make sure you understand the results. Assume that the expressions are evaluated in order.

```
⟹ (+ (* 3 7) 6)
;value:
⟹ (+ (*4 9) 5)
;value:
⟹ (- 7)
;value:
⟹ (/ 50 2)
;value:
⟹ (/ 60 3 2)
;value:
⟹ (/ 25)
;value:
⟹ (define twelve 10)
;value:  twelve
⟹ twelve
;value:
⟹ (define +$*+ (- 24 8))
;value: +$*+
⟹ +$*+
;value:
⟹ (define seven (lambda () 8))
;value: seven
⟹ seven
;value:
⟹ (+ seven 1)
;value:
```

```
==> (+ (seven) 1)
;value:
==> (define (my-six) 6)
```
What is another way to write this?
```
;value: my-six
==> my-six
;value:
==> (+ my-six 2)
;value:
==> (+ (my-six) 1)
;value:
==> (+ twelve 3)
;value:
==> (+ (twelve) 4)
;value:
==> (define b (* 2 twelve))
;value: b
==> b
;value:
==> (define n 8)
;value: n
==> (define (mul2 x) (* 2 x))
;value: mul2
```
*What is the equivalent de-sugared definition?*

```
==> (mul2 n)
;value:
==> n
;value:
==> (define - *)
;value: -
```
*Note - Pcscheme (but not MIT Scheme) gives a warning, and does not act consistently. You should continue as scheme* **should** *act*
```
==> (- 2 4)
;value:
==> (define (cube x) (* x x x))
;value: cube
==> (cube 3)
;value:
==> x
;value:
==> ((lambda (x) (* x x x)) 3)
;value:
==> ((lambda (x y) (+ (* 3 x) (* 4 y))) (+ 3 4) (+ 5 -6))
;value:
==> ((lambda (op) (op 5 6 7)) +)
;value:
==> ((lambda (op) (op 6 7 8)) (lambda (x y z) (+ x (* z 4))))
;value:
```