

## Quiz 2 Solutions

### Problem 1

```
(1 2)
(3 1 2)
(4 2)
(3 4 2)
(5 2)
(3 5 2)
```

### Problem 2

```
7
2
27
27
```

### Problem 3

```
false false true
true true true
true true true
true true true
```

### Problem 4

```
(define (amount s tree)
  (accumulate-tree tree
    (lambda (x) (if (eqv? x s) 1 0))
    +
    0))
```

### Problem 5

```
(define (make-toggle)
  (let ((answer 'no))
    (lambda ()
      (if (eq? answer 'no)
          (begin (set! answer 'yes) answer)
          (begin (set! answer 'no) answer)))))
```

*Problem 6*

The missing code:

```
((eq? action 'reset)
  (set! current (car old))
  (set! old (cdr old))
  current)
```

*Problem 7*

```
Learned by student : More fun with scheme: object oriented code
Learned by opl-student : metacircular evaluator
Learned by opl-student : More fun with scheme: generic operators
Learned by student : I know More fun with scheme: streams
```

*Problem 8*

```
<1> P2
<2> L1
<3> E1
<4> GE
<5> E1
<6> x
<7> E3
<8> 0
<9> 10
<10> E1
<11> 10
<12> GE
<13> P1
<14> P3
```