```
$Id: 2015q4-soln2,v 1.2 2015-11-18 11:12:58-08 - - $
Answers to cmps112-2015q4-exam1, page 1

_____________________________________________________________________
Question 1. [2]

Examples are multitudinous, so many other than these are correct.

parametric               template <typename T> class stack
                 -or-    class stack<T>

inclusion                class B extends A
or inheritance

_____________________________________________________________________
Question 2. [2]

conversion       void f(double);    f(6)

overloading      void f(double); void f(int);

_____________________________________________________________________
Question 3. [2]

(define (eval expr)
   (if (number? expr) expr
       (apply (car expr) (map eval (cdr expr)))))

_____________________________________________________________________
Question 4. [1]

sum := [:array|
   |sum|
   sum := 0.
   1 to: array size do: [:n| sum := sum + n].
   sum.
]

_____________________________________________________________________
Question 5. [2]


let ip a b =
    let rec ip' a b m = match a, b with
        | [], [] -> m
        | x::xs, y::ys -> ip' xs ys (m +. x *. y)
        | _, _ -> raise (Invalid_argument "ip")
    in ip' a b 0.0;;

```

```
Answers to cmps112-2015q4-exam1, page 2

_____________________________________________________________________
Question 6. [2]

let rec zip x y = match x, y with
    | [], [] -> []
    | x::xs, y::ys -> (x,y)::zip xs ys
    | _, _ -> failwith "zip";;

_____________________________________________________________________
Question 7. [4]

Object subclass: List [ |i_car i_cdr|
   List class >> new [ ^ nil ]
   List class >> car: a_car cdr: a_cdr [ |result|
      result := super new.
      result car: a_car cdr: a_cdr.
      ^ result
   ]
   car: a_car cdr: a_cdr [
      i_car := a_car.
      i_cdr := a_cdr.
   ]
   car [ ^ i_car ]
   cdr [ ^ i_cdr ]
]

_____________________________________________________________________
Question 8. [4]

Object subclass: Num [ |val|
   Num class >> new: n [ |r|
      r := super new.
      r set: n.
      ^ r.
   ]
   set: n [ val := n. ]
   value [ ^ val. ]
]
Object subclass: Mul [ |left right|
   Mul class >> left: n right: m [ |r|
      r := super new.
      r left: n right: m.
      ^ r.
   ]
   left: n right: m [ left := n.  right := m.  ]
   value [ ^ left value * right value. ]
]
```

```
101:
102: Answers to cmps112-2015q4-exam1, page 3
103:
104:  1.    (A) ALGOL
105:
106:  2.    (B) 2 sqrt
107:
108:  3.    (D) 5
109:
110:  4.    (B) M but not D
111:
112:  5.    (C) $ O ( 2 sup n ) $
113:
114:  6.    (B) duck typing
115:
116:  7.    (B) fold left $ O ( 1 ) $ and fold right $ O ( n ) $
117:
118:  8.    (A) ((foo bar) + foo) set: (3 + (4 next))
119:
120:  9.    (C) [3+4] value.
121:
122: 10.    (B) fold_left
123:
124: 11.    (D) throw
125:
126: 12.    (D) sum value: 3 value: 4
127:
```