

1. Evalúa **paso a paso** las siguientes expresiones: **(5 puntos)**

- i. `(equal? '((b . a) (a b))
 (append (list (cons b a)) '((a b).())))`
- ii. `(map (lambda(x y z) (z x y))
 (list list cons) (list list cons) (list cons list))`
- iii. `((lambda(x . y)(list x y)) 2 3 4)`
- iv. `(let ((x list) (y 2))
 (let* ((y 8) (z (+ y 5))) (x y z)))`
- v. `(drop-until positive? '(1 2 -3 -4))`

2. Responde **justificadamente** a las siguientes cuestiones: **(5 puntos)**

- i. ¿Cuándo una función se considera de orden superior?
- ii. ¿Analiza y justifica la siguiente afirmación: “la FOS filter de Scheme puede considerarse un *patrón de programación*”.
- iii. ¿Es cierto que las S-expresiones o bien son listas o bien son átomos?
- iv. ¿Qué es la evaluación perezosa y que ventaja tiene frente a la evaluación ansiosa? Pon un ejemplo.
- v. ¿En qué consiste la técnica de Currying? Pon un ejemplo.