

Auxiliar 1

Agarrando vuelo

P1. *Listas y Pares*

- (a) ¿Cuál es la diferencia entre (`cons` 'a 'b) y (`list` 'a 'b)?
- (b) Escriba el código que genere la lista: (`list` 1 '(2 (3 . 4)) #'f)
- (c) Dado (`define` l (`list` '(a (b c)) '(d e f))), ¿Cómo accedería al elemento b y el f en l? Hint: Para acceder al elemento d es: (`car` (`car` (`cdr` l))).
Hint: Usar una estructura mutable de Racket
- (d) Usando solo cons y la lista vacía, genere las siguientes expresiones: '(a . b) '(a b)' (a (b . c) d)

P2. *Funciones*

- (a) Usted tiene un monedero. El monedero solo puede contener monedas de 50, 100 y 500 pesos. Defina la función `sums-coins` que representa al monedero y retorna la cantidad de dinero total que hay en el monedero.
- (b) Defina una función llamada `sumar-multiplos` que reciba una lista de números y devuelva la suma de aquellos que sean múltiplos de 3 o 5.
- (c) Defina la función `pitatoria`, la cual recibe dos números naturales, el inicio y el final de la pitatoria, y realiza el cálculo correspondiente. Un ejemplo sería:

```
(pitatoria 2 4) ; <-> (* 2 3 4)
<< 24
```

- (d) Defina la función `lista_impares`, la cual recibe un número `n` natural que representa el largo de la lista de impares que se tendrá que crear.
Por ejemplo, si `n = 4`, la función debe retornar (`list` 1 3 5 7)