Profesor: Éric Tanter
Auxiliares: Rodrigo Urrea Tomás Vallejos
Ayudantes: R. Iturrieta P. Jaramillo A. Restrepo T. Rivas

CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

Auxiliar 1

Racket y programación funcional

Resumen.

- **Tipos Primitivos**: Números, booleanos, Strings y símbolos
- **Funciones predefinidas**: +, -, equal?, substring, ...
- **Condicionales**:
  - ```
    (if expr
        TRUE_BRANCH
        FALSE_BRANCH)
    ```

    ```
    (cond   [(> 2 3)  (printf "hola")]
            [(> 6 5)  (printf "chao")]
            [else #f])
    ```
- **Definir identificadores**: `(define a 2)`
- **Definir funciones**:

  ```
  (define (double x)
    (+ x x))
  ```
- **Estructuras de datos**:
  - Pares: `(cons 1 2)` o `'( 1 . 2 )`
  - Listas: `(list 1 2 3)` o `'(1 2 3)`]
- **Test** con `test` y `test/exn`

  ```
  (test (func args) result)
  (test/exn (func args) "error substring")]
  ```

Si tiene una duda sobre una función, puede usar el Help Desk (menú Help en DrRacket).
  ```

Recuerde enunciar el contrato y escribir tests.

P1. Conceptos.

(a) ¿Cual es la diferencia entre `(cons 'a 'b)` y `( list 'a 'b)`? ¿Como se representaría el segundo con notación de pares?

(b) ¿Cual es la notación de lista equivalente a `'((a . (b . ())) . (c . ()) ?`

(c) Dado `(define l ( list '(a b c) '(d e f) '(g h i)))`, ¿Como se accedería el elemento `'c` y el `'e` en l? Por ejemplo, `'b` es accesado por `(car (cdr (car l)))`

(d) Usando solo `cons`, la lista vacía y símbolos, muestre como construir las siguientes expresiones: `'(c)`, `'(a b)`, `'((a b) (c))`

P2. Programación.

(a) Defina la función `pair-add1 p` que recibe un par de números y retorna un nuevo par dónde los dos elementos fueron incrementados en `1`.

(b) Usted tiene un monedero. El monedero solo puede contener monedas de 50, 100 y 500 pesos. Defina la función `sums-coins` que recibe 3 enteros representando la cantidad de monedas de 50, 100 y 500 respectivamente Ly retorna la cantidad de dinero total que hay en el monedero

(c) Defina la función `tax`, que recibe como argumento el sueldo bruto y retorna el impuesto a pagar. Para un sueldo menor de $500,000 el impuesto es de 0 %, entre $500,000 y $750,000 el impuesto es de %15 y para más o igual a $750,000 es de %28

(d) Defina la función `netpay` que recibe la cantidad de horas trabajadas en un mes y retorna el sueldo líquido del trabajador, asuma que el pago por hora es de $5,000

(e) Implemente la función quicksort que recibe una lista y retorne la lista ordenada en forma ascendente. Tome siempre como pivote el primer elemento, Ejemplo:

```
> (quicksort '(3 2 9 1))
    '(1 2 3 9)
```

P3. Funcional.

Defina las siguientes funcionalidades utilizando map, foldl, or filter:

(a) Dado una lista de enteros, retorne una lista con los elementos incrementados en uno.

(b) Dado una lista de strings, retorne una lista con la longitud de cada cadena.

(c) Dado una lista de enteros, retorne la suma de todos su elementos.

(d) Dado una lista de string, retorne el string resultante de la concatenacion de todas las cadenas de la lista.

(e) Dado un lista de enteros, retorne la lista de todos los elementos mayores que cero.