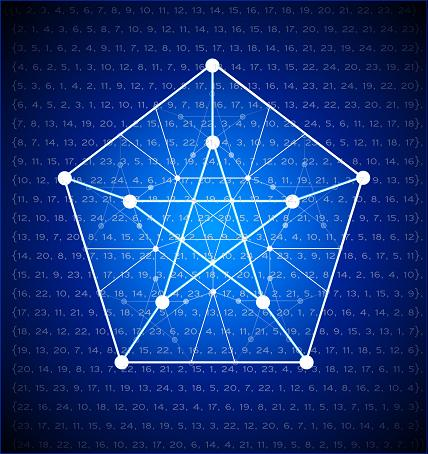
****

**Tecnológico de Monterrey**

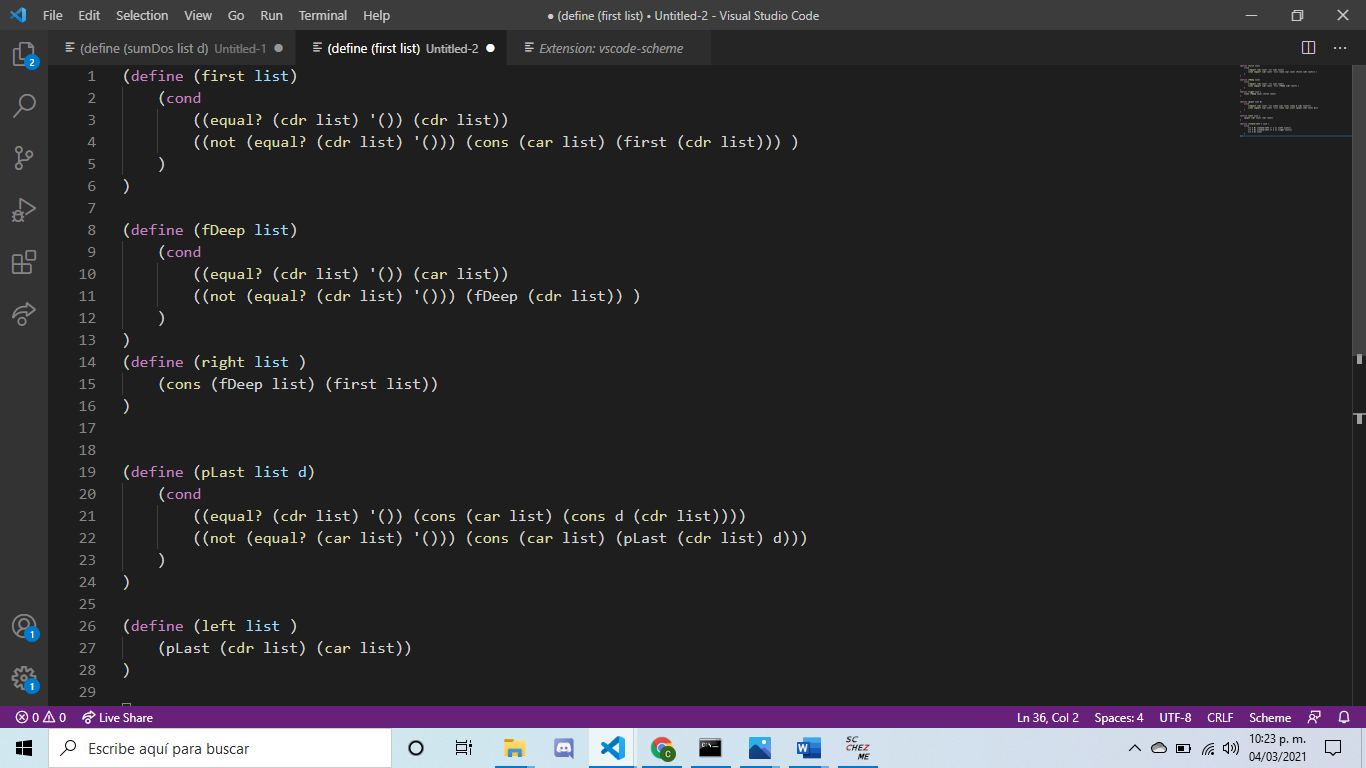
**TC2037: Implementación de métodos computacionales**

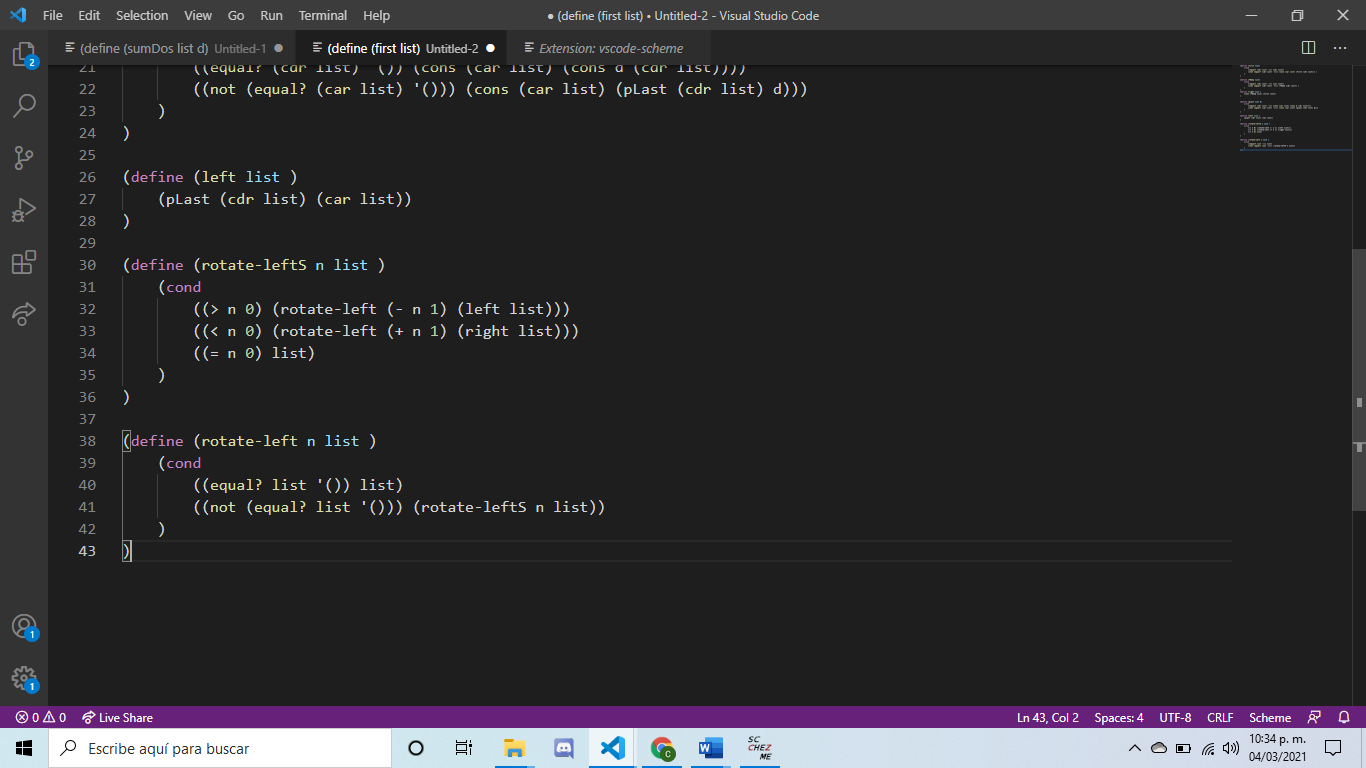
Actividad 2.1: Programación funcional – Parte 2

****

* Carlos Estrada Ceballos A01638214
* Abigail Velasco García A01638095
* Natalia Velasco García A01638047
* **2**

La función ‘rotate-left’ toma dos entradas: un numero entero n y una lista lst. Devuelve la lista que resulta de rotar ‘lst’ un total de n elementos a la izquierda. Si n es negativo, rota hacia la derecha.





(define (first list)

(cond

((equal? (cdr list) '()) (cdr list))

((not (equal? (cdr list) '())) (cons (car list) (first (cdr list))) )

)

)

(define (fDeep list)

(cond

((equal? (cdr list) '()) (car list))

((not (equal? (cdr list) '())) (fDeep (cdr list)) )

)

)

(define (right list )

(cons (fDeep list) (first list))

)

(define (pLast list d)

(cond

((equal? (cdr list) '()) (cons (car list) (cons d (cdr list))))

((not (equal? (car list) '())) (cons (car list) (pLast (cdr list) d)))

)

)

(define (left list )

(pLast (cdr list) (car list))

)

(define (rotate-left n list )

(cond

((> n 0) (rotate-left (- n 1) (left list)))

((< n 0) (rotate-left (+ n 1) (right list)))

((= n 0) list)

)

)

