Ejercicios de tercer parcialito
Taller 23/05

Del tercer parcialito del primer cuatrimestre de 2012

1. Escribir una función *reemplazar* que tome una Pila, un valor_nuevo y un valor_viejo y reemplace en la Pila todas las ocurrencias de valor_viejo por valor_nuevo. Considerar que la Pila tiene las primitivas apilar(dato), desapilar() y esta_vacia().

Del recuperatorio del tercer parcialito del primer cuatrimestre de 2012

2. Escribir un método que invierta una ListaEnlazada utilizando una Pila como estructura auxiliar y considerando que lista solo tiene una referencia al primer nodo.

Del tercer parcialito del primer cuatrimestre de 2013

3. Escribir una función que reciba una pila de números y elimine de la misma los elementos consecutivos que están repetidos. Se pueden usar estructuras auxiliares. La función no devuelve nada, simplemente modifica los elementos de la pila que recibe por parámetro.
Por ejemplo: remover_duplicados_consecutivos(Pila([2, 8, 8, 8, 3, 3, 2, 3, 3, 3, 1, 7])) Genera: Pila([2, 8, 3, 2, 3, 1, 7]).

Del segundo recuperatorio del tercer parcialito del primer cuatrimestre de 2013

4. Para una lista simplemente enlazada de números (que solo mantiene una referencia al primer nodo) implementar la primitiva suma_acumulativa() que devuelva una nueva lista (del mismo largo) tal que el nodo i de la nueva lista contenga la suma acumulativa de los elementos de la lista origina hasta el nodo i.

Del segundo recuperatorio del tercer parcialito del primer cuatrimestre de 2014

5. Escribir una función *reducir* que reciba por parámetro una cola y una función f de dos parámetros, y aplique sucesivamente la función f a los dos primeros elementos de la cola (luego de desencolarlos) y encole el resultado, hasta que solo quede un elemento. La función *reducir* debe devolver el único elemento restante en la cola.

Del primer recuperatorio del tercer parcialito del primer cuatrimestre de 2014

6. Escribir una función que reciba una cola y la cantidad de elementos en la misma, y devuelva True si los elementos forman un palíndromo o False si no.
Por ejemplo:
`es_palindromo([n, e, u, q, u, e,n], 7)` $->$ True.