

Auxiliar Extra - Preparación para control 1

Profesores: Jérémy Barbay
Patricio Poblete, Nelson Baloian
Auxiliares: Felipe Lizama, Franco Sanguinetti,
Matías Ramírez, Sven Reisenegger.

P1. P2 control 1 otoño 2018

Considere la siguiente implementación para nodos de una lista enlazada:

```
class Nodo:  
    def __init__(self, val, sig):  
        self.valor = val  
        self.sigte = sig
```

En esta implementación, `val` corresponde a un entero y `sig` es una referencia a otro `Nodo`. Adicionalmente, suponga que los valores en una lista están **ordenados ascendentemente**. El objetivo es de implementar **de manera iterativa**¹ una función que reciba dos listas enlazadas y devuelva el número de elementos distintos entre ellas, es decir, el total de elementos que sólo aparece en una de las listas.

Por ejemplo, para las listas que contienen los valores $A = \{12, 22, 45, 67, 81, 99\}$ y $B = \{8, 17, 22, 60, 81\}$ la función debe devolver 7, dado que los valores $\{8, 12, 17, 45, 60, 67, 99\}$ (siete valores en total) son los que aparecen en sólo una de las listas.

- Describa el invariante de su programa con un dibujo (¡claro!) y/o una frase formal en castellano.
- Escriba el pseudo-código (o el código Python) que calcule lo pedido. Si las listas tienen tamaño n y m respectivamente, su función debe tomar tiempo $O(n + m)$.
- Muestre que su algoritmo cumple con la cota temporal exigida.

P2. P3 control 1 otoño 2017

Definamos el peso de un árbol binario como el número de nodos internos que pertenecen a él, y digamos que un nodo interno es equilibrado si los pesos de sus subárboles izquierdo y derecho son exactamente iguales. Escriba un método que reciba un puntero a la raíz de un subárbol y que recorra ese subárbol imprimiendo el contenido de los campos `info` de todos sus nodos equilibrados. Indicación: Aproveche la misma función para hacer que, además de imprimir lo que se pide, retorne el peso del árbol que acaba de recorrer. Piense recursivamente.

¹Una solución recursiva recibirá solamente puntos parciales.

P3. P3 control 1 otoño 2019

Una persona desea ir en auto desde un punto 0 hasta un punto N , siguiendo una línea recta. En todos los puntos $0, 1, \dots, N$ hay puestos de arriendo de autos. Debido a las tarifas, es posible que para ir desde el punto i hasta un punto j sea más económico contratar un auto desde i hasta un punto intermedio k y otro desde k hasta j . Suponiendo que se conoce el costo $a(i, j)$ de contratar un auto retirándolo en el punto i y entregándolo en el punto j , calcule el costo óptimo $C(i, j)$ para ir desde i hasta j para todo i, j . Para esto, encuentre una fórmula recursiva y luego aplique tabulación (programación dinámica) para encontrar un algoritmo eficiente. Analice cuánto demora el cálculo de $C(0, N)$, en función de N (solo orden de magnitud).