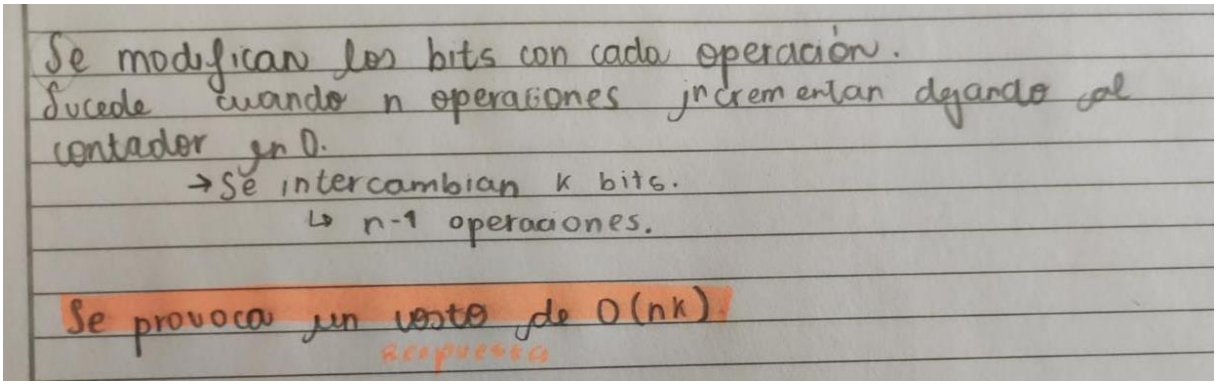
Ma. Isabel Ortiz Naranjo

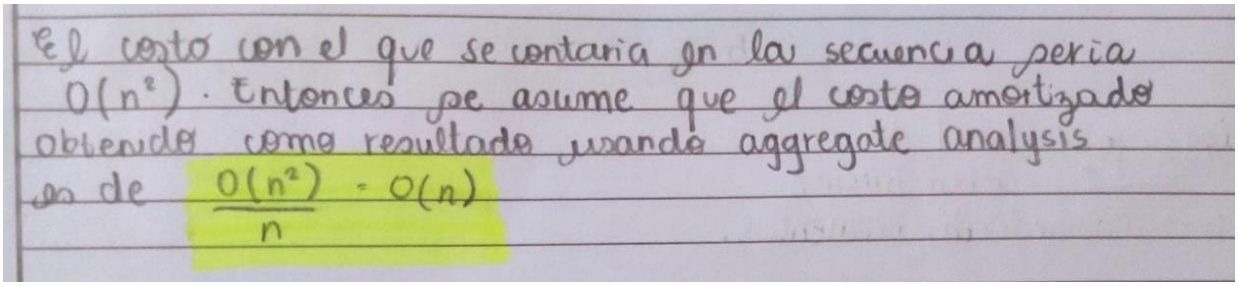
Carné: 18176

Tarea 5 – Análisis de algoritmos

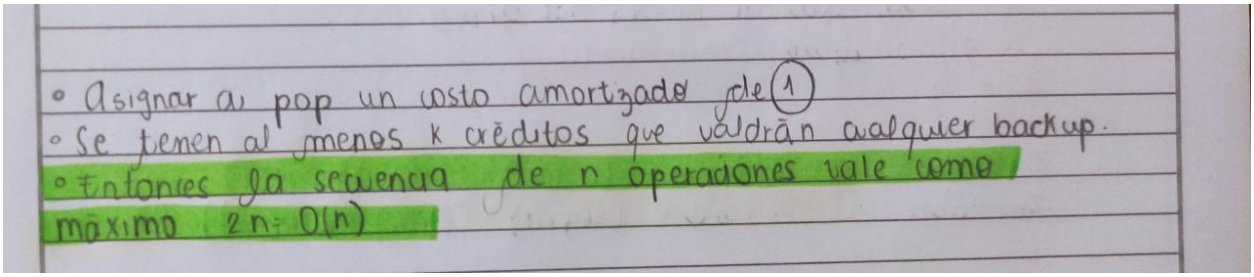
1. Agregamos al contador binario una operación de decremento: *Decrement*. Identifique el *worst-case scenario* realista para una secuencia de n operaciones *Increment* y *Decrement* combinadas, y provea una cota superior para su tiempo de ejecución. Suponga que el contador trabaja con una lista de k bits.



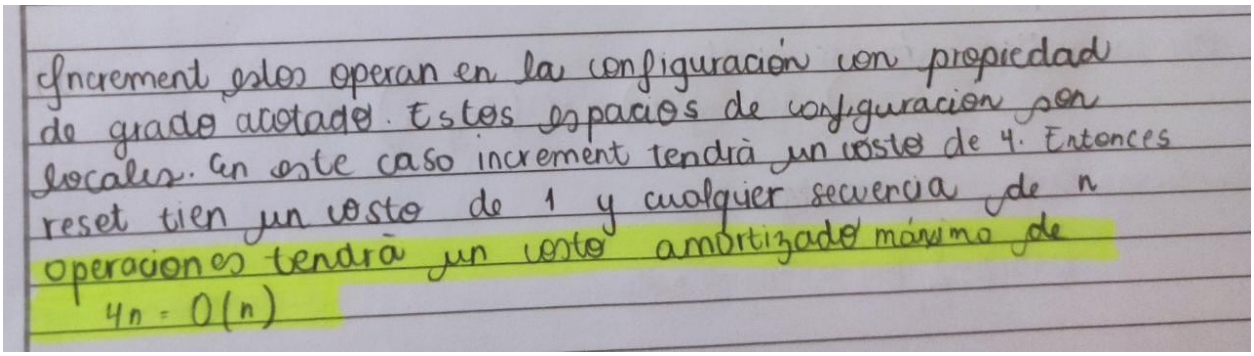
2. Supongamos que a una pila con *multipop* agregamos al operación *multipush(k, A)* que hace una secuencia de operaciones *push* con los primeros k elementos en el arreglo A. En este caso, ¿se mantiene el costo amortizado de $O(1)$ por operación? Explique.



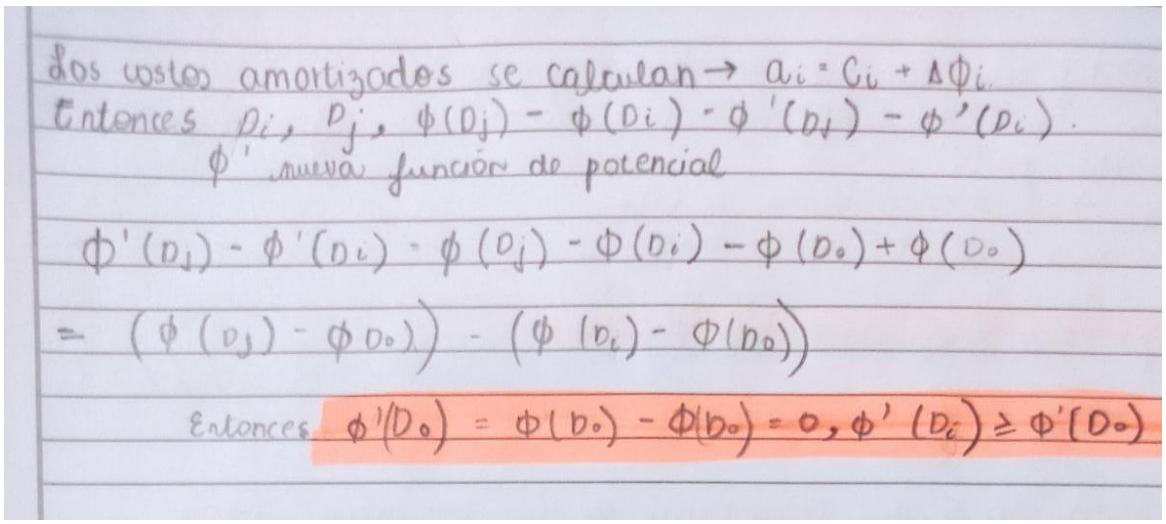
3. Considere una pila común y corriente cuyo tamaño nunca excede k. Luego de k operaciones *push* y/o *pop* se efectúa un *backup* automático, copiando toda la pila. El costo de copiar m elementos es m. Demuestre con el *accounting method* que cualquier secuencia de n operaciones entre *push* y *pop* (con *backups* cada k operaciones) costará $O(n)$. ***Hint***: considere los costos asignados a estas operaciones en el ejemplo *multipop*.



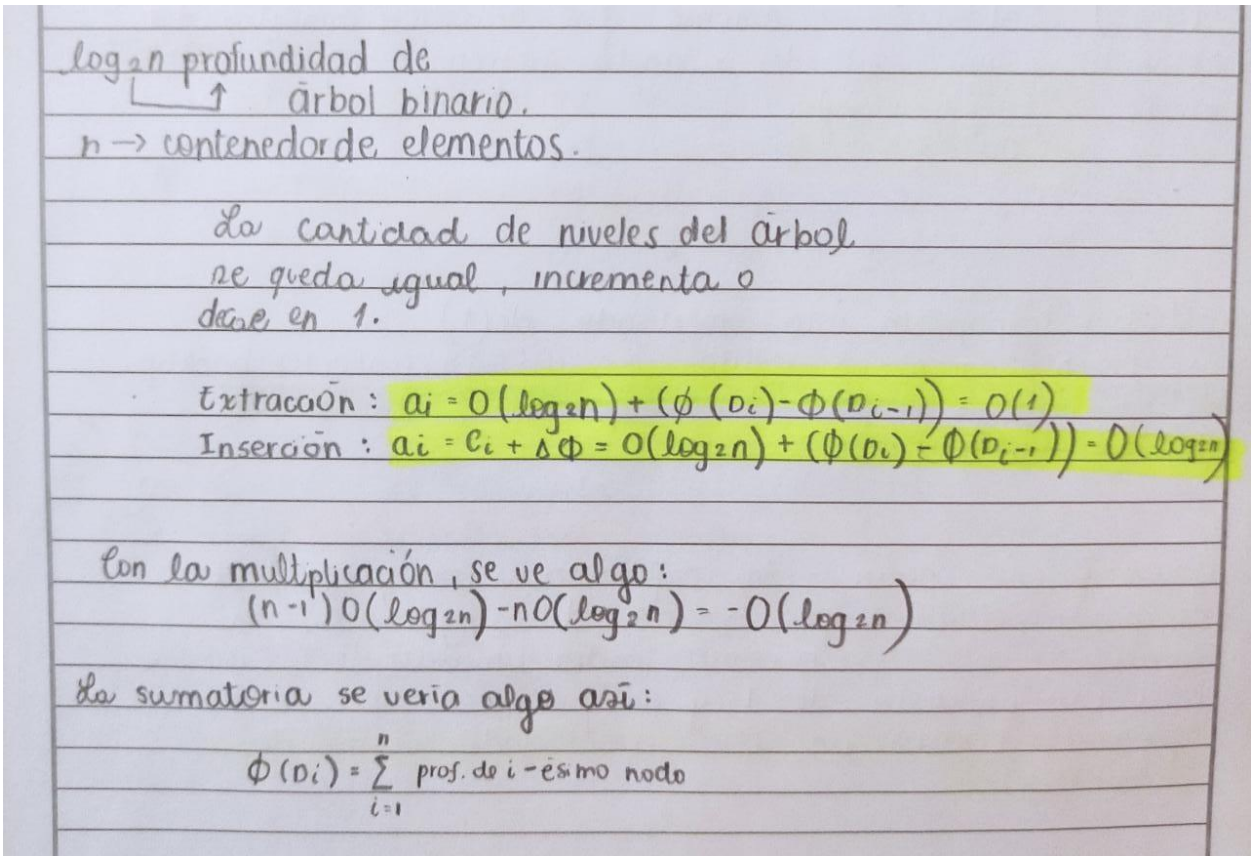
4. Suponga que al contador binario de los ejemplos se le agrega la operación *reset* que busca y convierte todos los 1 en 0, uno por uno a partir del bit menos significativo. Demuestre con el *accounting method* que cualquier secuencia de n operaciones entre *increment* y *reset* toma un tiempo de ejecución de $O(n)$. Considere que el contador inicia desde 0 y que cada revisión y cada modificación de un bit toma $\Theta(1)$. ***Hint***: ¿hasta qué bit del número binario debe llegar *reset* en cualquier momento, y cómo podemos asegurar que todos los bits que *reset* modifique tengan crédito para pagar por su *resetteo*?



5. Tenemos una función de potencial Φ tal que $\Phi(D_i) \geq \Phi(D_0)$ para todo i, pero $\Phi(D_0) \neq 0$. Defina una función Φ' tal que $\Phi'(D_0) = 0, \Phi'(D_i) \geq \Phi'(D_0), \forall i \geq 1$, y que los costos amortizados obtenidos con Φ' sean los mismos que con Φ.



6. Un *min-heap* binario es un árbol binario completo (todos sus niveles están llenos y sus hojas se llenan de izquierda a derecha) en donde cada nodo es menor que todos sus hijos. El *min-heap* tiene una operación de inserción llamada *insert,* pero consideremos además la función *extract-min,* que reemplaza la raíz por el último nodo del árbol y luego la intercambia con el menor de sus hijos. Supongamos que ambas operaciones tienen un tiempo de ejecución real $O(\log_2 n)$. Provea una función de potencial según la cual el costo amortizado de *insert* sea $O(\log_2 n)$ y el de *extract-min* sea $O(1)$. No olvide que el potencial siempre debe ser positivo (no necesariamente el cambio de potencial), y que el potencial inicial es idealmente cero. Demuestre que su función de potencial funciona.
Hint: primero considere la forma que tendrá el costo real de cada operación. Con esto en mente, escriba la fórmula de costo amortizado, usando valores en el cambio de potencial de forma que al sumarlo con el costo real se obtengan los costos amortizados deseados (claramente en uno de los casos el cambio de potencial debe resultar menor a $O(\lg n)$; y en el otro caso debe ser negativo). ¿Qué característica(s) de un árbol binario completo se miden con un $\log_2 \blacksquare$? ¿Cómo cambia(n) esta(s) característica(s) luego de un *insert* o un *extract-min*? Recuerde que el potencial debe extraerse de alguna propiedad de la estructura de datos.



$$= \sum_{i=1}^{n-1} \text{prof. del } i\text{-ésimo nodo} + O(\log_2 n)$$

costo amortizado de una extracción es:

$$a_i = O(\log_2 n) - O(\log_2 n) = O(1)$$

Costo amortizado de una inserción es:

$$a_i = O(\log_2 n) + \left(\sum_{i=1}^{n} \text{profun. del } i\text{-ésimo nodo} + O(\log_2 n) - \sum_{i=1}^{n} \text{prof. del } i\text{-ésimo nodo} \right) = O(\log_2 n)$$