**Primera Entrega Proyecto Análisis de Algoritmos**

**Camilo Rocha**

**Miguel Romero**

**Pontificia Universidad Javeriana Cali**

**Ingeniería de Sistemas y Computación**

**2019-2**

**Sebastián Toro Franco**

**Especificación del problema**

**Entrada:** El algoritmo recibirá en total 5 parámetros, donde 4 son enteros: la longitud K de la secuencia o cadena donde 2 ≤ K ≤ 10⁶; los segundos M donde 0 ≤ M ≤ 10⁴ durante los cuales la secuencia cambia su estado; y los limites del intervalo principal a manipular ai y bi siendo ai ≤ bi. Finalmente, el 5 parámetro será la secuencia de 1´s y 0´s (unos y ceros) sobre la cual se trabajará usando los anteriores 4 parámetros.

**Salida:** Una cadena hexadecimal que describa el estado de la secuencia de luces en el segundo M

**Algoritmos y estructuras de datos a usar**

Actualmente no se usa ningún tipo de algoritmo específico, ni tampoco se utilizan estructuras específicas como clases. La solución inicial que se ha pensado es de tipo inocente al usar la fuerza bruta.

**Estrategia de solución**

Paso 1: transformar la cadena hexadecimal recibida a binario, rellenándola con ceros hasta alcanzar longitud K.

Paso 2: Ubicar el elemento ai y bi.

Paso 3: Verificar si alguno de los elementos anteriores en 0 (cero).

Paso 5: En caso de que alguno de los elementos ai cero, recorrer la cadena desde ai hasta li donde li es el primer 1 (uno) encontrado a la izquierda de ai. En caso de que bi sea cero, recorrer la cadena desde bi hasta ri donde ri es el primer 1 (uno) encontrado a la derecha de bi.

Paso 6: Realizar un cambio del estado en el intervalo [ai,bi] (cambiar 1’s a 0’s y viceversa).

Repetir los pasos 1 a 6 hasta que hayan pasado M segundos.

**Análisis (alto nivel) de complejidad temporal y espacial**

Este algoritmo al ser inocente y hacer uso de la estrategia de fuerza bruta, a nivel temporal el algoritmo estará dentro de la complejidad O(K2\*M). Siendo K la longitud de la cadena, es decir que en el peor de los casos el algoritmo tendrá que recorrer la cadena para encontrar los límites ai, bi y luego cambiar todos los elementos comprendidos por el intervalo [ai,bi] incluidos los límites. M siendo el número de iteraciones en las que se cambiará la cadena.

A nivel espacial el algoritmo entra dentro de la complejidad O(K\*M). Esto debido a que se guardarán M filas de K columnas.

**Citas bibliográficas y posibles soluciones futuras al proyecto:**

* Z. Akbari. Segment Tree. Enero, 2014. Recuperado de: <http://cs.yazd.ac.ir/farshi/Teaching/CG3921/Slides/chapter10%20-section3.pdf>