

Documento de análisis Reto 1 G10

Juan Camilo Falla 201922219 - j.fallag@uniandes.edu.co

Nicolas Klopstock 202021352 - n.klopstock@uniandes.edu.co

Análisis de complejidad:

1. Requerimiento 1:

$O(n)$ en el proceso más largo (n siendo la cantidad de videos en el catálogo)

En el requerimiento 1 se deben encontrar los videos más vistos en un país, de una categoría determinada, cuyos parámetros son elegidos por parte del usuario. En nuestro código se inicia filtrando todos los datos que coincidan con el país que es elegido por el usuario, y los videos que se encuentren en un mismo país serán agregados a una lista de retorno. Debido a que es necesario recorrer todo el catálogo de videos (con una cantidad n de datos), el orden de crecimiento de este proceso es $O(n)$. El segundo proceso realizado por el requerimiento busca el ID de la categoría que ingresa el usuario, el cual, en el peor de los casos, tiene una complejidad $O(z)$ (siendo z la cantidad de categorías que hay). La lista de retorno entregada por el primer proceso se considerará con una cantidad m de datos. Esta lista nuevamente es filtrada para poder encontrar los videos que coincidan con la categoría que ingresa el usuario y debido a que es necesario recorrer toda la lista, el filtrado por categoría tiene un orden de complejidad de $O(m)$ y retorna una lista con una cantidad p de datos. Por último, dicha lista debe ser organizada teniendo en cuenta las vistas de cada video y este proceso se hace con un “Merge sort”, lo que implica que su orden de complejidad es $O(p \log(p))$.

2. Requerimiento 2 (Nicolas Klopstock).

$O(n)$ en el proceso más largo (n siendo la cantidad de videos en el catálogo)

En el segundo requerimiento nuevamente se hace el filtrado por país del catálogo y una vez más se observa un orden de crecimiento de $O(n)$ y se retorna una lista de tamaño m . Posteriormente se realiza un ordenamiento por el ID de los videos con un “Shell sort”, debido a que el uso de un “Merge” ocuparía demasiada memoria RAM, y este tiene un orden de crecimiento de $O(m^{\frac{3}{2}})$, en el peor de los casos. Posteriormente se da un recorrido de la lista para encontrar el video con más días como tendencia, lo que implica que el proceso tiene un orden de crecimiento de $O(m)$.

3. Requerimiento 3 (Juan Camilo Falla)

$O(n)$ en el proceso más largo (n siendo la cantidad de videos en el catálogo)

El tercer requerimiento, en términos de complejidad de operaciones es bastante similar al segundo. El primer proceso encuentra el ID de la categoría que se desea encontrar, cuyo orden de complejidad es lineal con respecto a la cantidad de categorías que se encuentren. En el segundo

proceso se filtran los videos que estén dentro de una misma categoría y este proceso tiene una complejidad de $O(n)$, teniendo en cuenta que n equivale a la cantidad de videos en el catálogo, el retorno de este proceso es una lista con m elementos. En este requerimiento se recorre nuevamente la lista dos veces para encontrar el video con más días como tendencia, lo que significa que el orden de complejidad de este tercer proceso es de $O(m)$. Por último, se imprime el resultado que tiene una complejidad constante.

4. Requerimiento 4

$O(n)$ en el proceso más largo (n siendo la cantidad de videos en el catálogo)

El último requerimiento nuevamente realiza un filtro de la información por los países que se desean ver, antes de proveer los resultados. Dicho filtro nuevamente tiene un orden de complejidad de $O(n)$ (teniendo en cuenta que n es la cantidad de videos que se encuentran en el catálogo) y retorna una lista con m datos (que contiene todos los videos tendencia en un país). El segundo proceso es un filtro por la etiqueta que ingresa por parámetro, y su complejidad es $O(m)$ y se retorna una lista p . Por último, se hace una organización de esta lista con p datos mediante un Quicksort, el cual tiene una complejidad de $O(p \log(p))$ y se imprimen los primeros tres videos de esta lista organizada.