

DOCUMENTO DE ANÁLISIS – RETO 1

Jose Vicente Rincón Celis – 202013321 – j.rinconc@uniandes.edu.co

Juan Esteban Arboleda Restrepo – 201921578 – j.arboleda@uniandes.edu.co

Análisis de complejidad de los requerimientos

Requerimiento 1 (Trabajo en equipo)

Pasos del requerimiento 1

Paso	Descripción	Complejidad	Observaciones
1.	Input del usuario (categoría, país, número de video)	$O(1)$	
2.	Llamado a la función <code>topVidsCatCountry()</code>	$O(n \log n + O(n))$	Más adelante se discute en detalle la complejidad de esta función
3.	Output de resultados	$O(n)$	El peor caso es en el que los videos del resultado son todos los videos de la base de datos. En ese caso, para imprimir los resultados hay que hacer un ciclo n veces.

Pasos de la función `topVidsCatCountry()`

Paso	Descripción	Complejidad	Observaciones
1.	Creación de estructuras de datos	$O(1)$	
2.	Recorrido por videos para filtrar	$O(n)$	
3.	Ordenamiento de videos (mergesort)	$O(n \log n)$	En el peor de los casos, todos los videos cumplen con los filtros de búsqueda, por lo que hay que ordenar todos los videos

Requerimiento 2 (Jose Vicente Rincón Celis)

Pasos del requerimiento

Paso	Descripción	Complejidad	Observaciones
1.	Input del usuario (categoría)	$O(1)$	
2.	Llamado a la función <code>trendingVidCountry()</code>	$O(n^2 + n \log(n))$	Más adelante se discute en detalle la complejidad de esta función
3.	Output de los resultados	$O(1)$	

Pasos de la función `trendingVidCountry()`

Paso	Descripción	Complejidad	Observaciones
1.	Creación de estructuras de datos	$O(1)$	
2.	Iteración por videos para filtrado	$O(n^2)$	Se debe iterar por cada video y (en el peor de los casos) en cada iteración se debe llamar a la función <code>isPresent</code> que tiene complejidad $O(n)$, por tal motivo, la complejidad total es $O(n^2)$
3.	Ordenamiento de videos filtrados (mergesort)	$O(n \log n)$	En el peor de los casos todos los videos pasan los filtros

Requerimiento 3 (Juan Esteban Arboleda Restrepo)

Pasos del requerimiento

Paso	Descripción	Complejidad	Observaciones
4.	Input del usuario (categoría)	$O(1)$	
5.	Llamado a la función trendingVidCat()	$O(n^2 + n \log(n))$	Más adelante se discute en detalle la complejidad de esta función
6.	Output de los resultados	$O(1)$	

Pasos de la función trendingVidCat()

Paso	Descripción	Complejidad	Observaciones
4.	Creación de estructuras de datos	$O(1)$	
5.	Iteración por videos para filtrado	$O(n^2)$	Se debe iterar por cada video y (en el peor de los casos) en cada iteración se debe llamar a la función isPresent que tiene complejidad $O(n)$, por tal motivo, la complejidad total es $O(n^2)$
6.	Ordenamiento de videos filtrados (mergesort)	$O(n \log n)$	En el peor de los casos todos los videos pasan los filtros

Requerimiento 4 (trabajo en equipo)

Pasos del requerimiento

Paso	Descripción	Complejidad	Observaciones
1.	Input del usuario (tag, país, número de video)	$O(1)$	
2.	Llamado a la función mostComentedVids()	$O(n^2 + n)$	Más adelante se discute en detalle la complejidad de esta función
3.	Output de los resultados	$O(1)$	

Pasos de la función trendingVidCat()

Paso	Descripción	Complejidad	Observaciones
1.	Creación de estructuras de datos	$O(1)$	
2.	Iteración por videos para filtrado	$O(n)$	
3.	Iteración por videos filtrados para excluir repeticiones	$O(n^2)$	En el peor de los casos, todos los videos pasan el filtro. Además se llama a la función isPresent, por lo que la complejidad total (en el peor de los casos) es $O(n^2)$