Integrantes:

- Sofia Velasquez Marin - s.velasquezm2@uniandes.edu.co - C.E: 202113334
- Valeria Caro Ramirez - v.caro@uniandes.edu.co - C.E: 202111040

Documento de Análisis

En el siguiente documento, registraremos la complejidad y el tiempo de carga de los requerimientos del Reto 2.

Cargar información del Catalogo

Tiempo de carga

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	187.5
5%	1328
10%	2250.0
20%	4093
30%	5687.5
50%	8875
80%	14093.75
Large	16125

Req 1.

Para el requerimiento 1 se usó una función: Se recorren las llaves del mapa Begin Dates utilizando el método keySet () de complejidad O (# Llaves). Se organizan estas llaves por la fecha utilizando Merge Sort de complejidad O (# Llaves *log (# Llaves)). Luego se iteran las llaves, esto tiene una complejidadad de O (# Llaves). Posteriormente se compara el key con beginDate y endDate, dos parámetros que recibe la función. Después se obtienen los valores del año utilizando el método get () y se obtienen los valores de ese año con el método getValue (), ambos métodos tienen de complejidad de O(1). Luego se itera la lista de obras de ese año y se agrega a una lista. Haciendo la suma de complejidades, la complejidad total fue: O (# Llaves + #Llaves * log(#Llaves) + #Artistas) donde #Llaves es el numero de fechas de nacimiento que existen el archivo de artists y #Artistas es el numero de artistas que nacieron del rango dado por parámetro.

Tiempo de carga

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	15.625
5%	15.625
10%	15.625
20%	15.625
30%	15.625
50%	15.625
80%	31.25
Large	31.25

Complejidad: O (# Llaves + #Llaves * log(#Llaves) + #Artistas).

Req 2.

Tiene el mismo funcionamiento del Req 1, pero se diferencian en que este requerimiento se recorre el mapa de DatesAcquired. Haciendo la suma de complejidades, la complejidad total fue: O (# Llaves + #Llaves * log(#Llaves) + #Obras) donde #Llaves es el número de fechas de adquisición que existen en el archivo de artworks y #Obras es el número de obras que fueron adquiridas en el rango dado por parámetro.

Tiempo de carga:

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	46.875
5%	187.5
10%	234.375
20%	343.75
30%	546.87
50%	671.87
80%	687.5
Large	718.75

Complejidad: O (# Llaves + #Llaves * log(#Llaves) + #Obras).

Req 3.

Para el requerimiento se usaron 2 funciones: la primera se encarga de buscar el nombre que ingresa por parámetro en el map ArtistsNames, esta búsqueda se hace con el método get() el cual tiene una complejidad O(1), se saca su id y se busca en el map ArtistsWorks y retorna su valor (el TAD map con los medios), esta función tiene una complejidad temporal total de O(1); La segunda función recorre las llaves del mapa retornado en la primera función usando el método keySet() el cual tiene una complejidad de O(#Llaves), y por cada llave se calcula el número de obras asociadas y se suman a un contador y a la par se calcula cual es el medio (llave) con mayor número de obras asociadas, la complejidad de esta función es de O(#Llaves). Haciendo la suma de complejidades, O (1 + #Llaves), la complejidad del requerimiento es O(#Llaves) donde el #Llaves es el número de medios utilizados por el artista que se quiere consultar.

Tiempo de carga:

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	0.0
5%	0.0
10%	15.625
20%	15.625
30%	15.625
50%	15.625
80%	15.625
Large	15.625

Complejidad: O(#Llaves)

Req 4.

Para el requerimiento 4 se usó solo una función: Esta se encarga de recorrer las llaves del índice de nacionalidad, usando el método keySet () que tiene una complejidad O (#Llaves). Se itera por las llaves del mapa, luego se utiliza get () de complejidad O (1) para obtener las nacionalidades. Posteriormente si la nacionalidad existe, utiliza la función getValue de complejidad O (1) para obtener el valor de las obras de la nacionalidad. Luego se agregan a una lista, la longitud de la nacionalidad utilizando size(value) de complejidad O (1), la nacionalidad(key) y las obras de la nacionalidad(value). Luego se organiza la lista por la longitud, utilizando Merge Sort que tiene una complejidad O (# Llaves * log(#Llaves)). Se crea una sublista de la lista y se toma desde la posición 1 a la 10. A esa sublista, se le aplica el método getElement () de complejidad O (1) para obtener las obras de la nacionalidad de mayor longitud. Por último, a la variable donde se almacenan las obras de la nacionalidad de mayor longitud se le hace getElement (), para devolver los tres primeros y tres últimas obras.

Tiempo de carga:

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	0.0 o 15.625
5%	0.0 o 15.625
10%	0.0 o 15.625
20%	0.0 o 15.625
30%	0.0 o 15.625
50%	0.0 o 15.625
80%	0.0 o 15.625
Large	0.0 o 15.625

Complejidad: O (#Llaves + (# Llaves * log(#Llaves)).

Req 5.

Para el requerimiento 5 se usó una función: Se busca un departamento dado por parámetro, utilizando el método get () y se obtienen los valores de ese departamento utilizando el método getValue (), ambos métodos tienen una complejidad de O (1). Luego se sacan los valores de las obras que hay ese departamento y estos se organizan por fecha y costo, para esto se utiliza Merge Sort de complejidad O (n * log(n)). Haciendo la suma de complejidades, como se hacen dos ordenamientos, seria O(n*log(n) + n*log(n) + 1), en total la complejidad del requerimiento seria O(n*log(n)), n siendo el numero de obras pertenecientes al departamento.

Tiempo de carga:

Tamaño del archivo	*Tiempo de Ejecución [ms]*
Small	31.25
5%	203.125
10%	421.875
20%	890.625
30%	1468.75
50%	2421.875
80%	4031.25
Large	5281.25

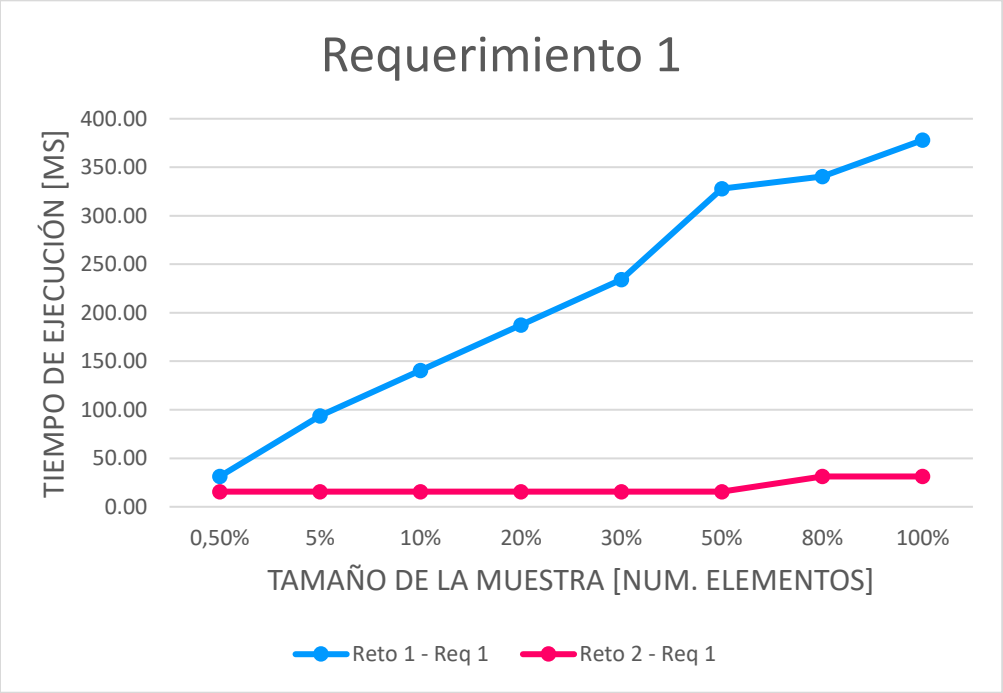
Complejidad: O(n*log(n))

Diagrama

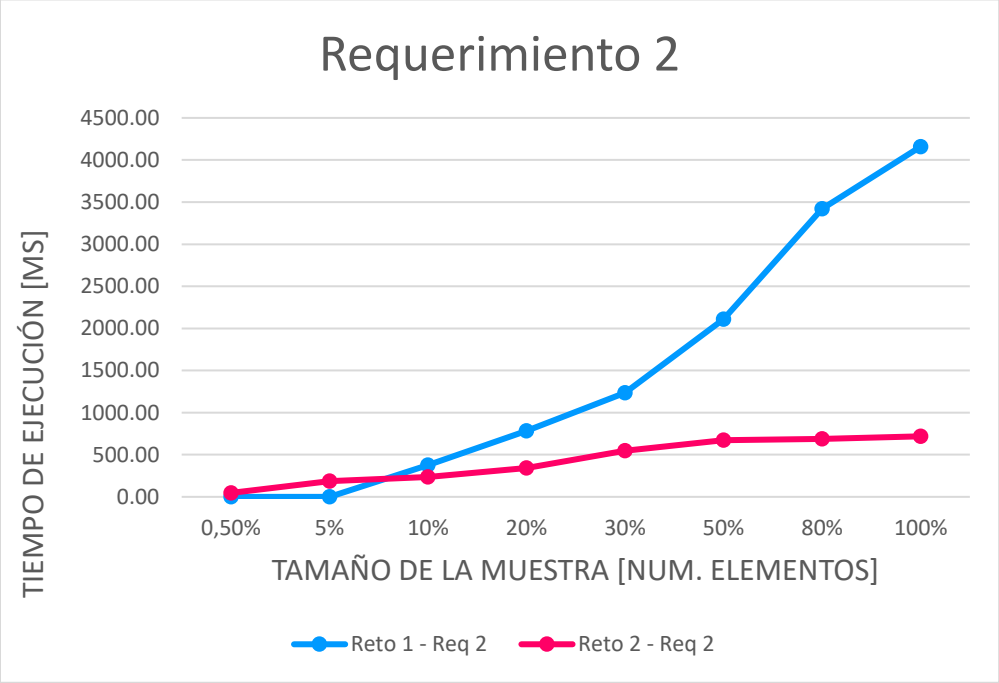
Para visualizar el diagrama, acceder a la carpeta Docs y ahí se encuentra una imagen con el Diagrama.

Graficas de comparación del tiempo de ejecución del Reto 1 vs Reto 2

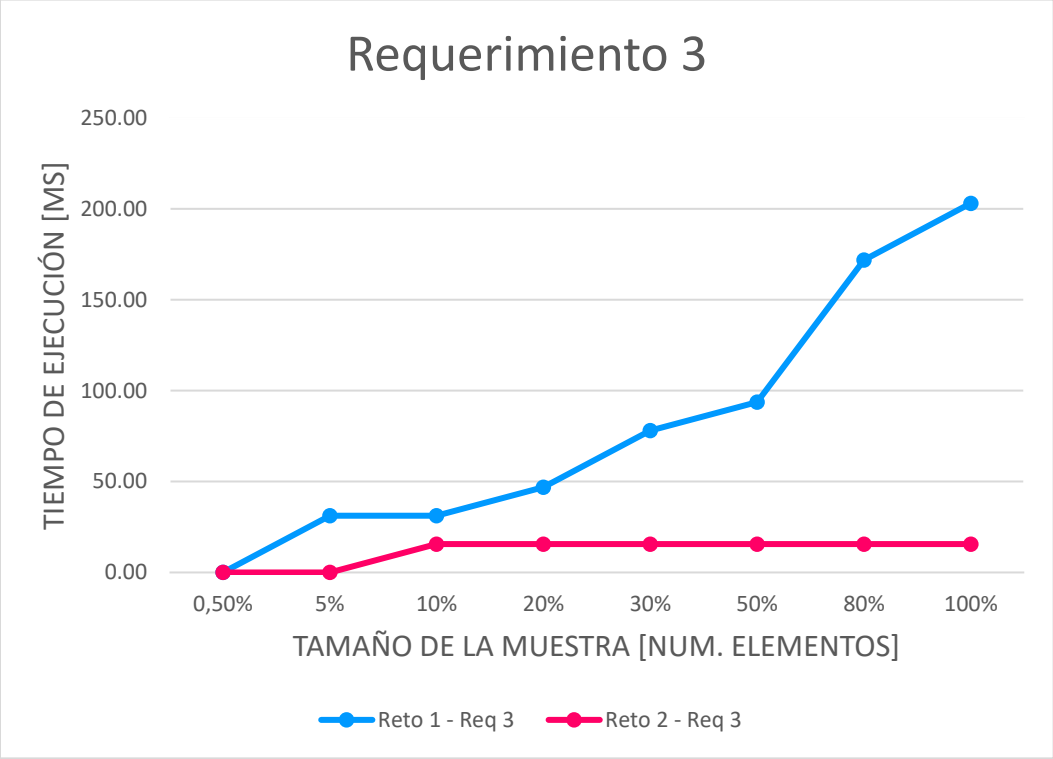
Req 1



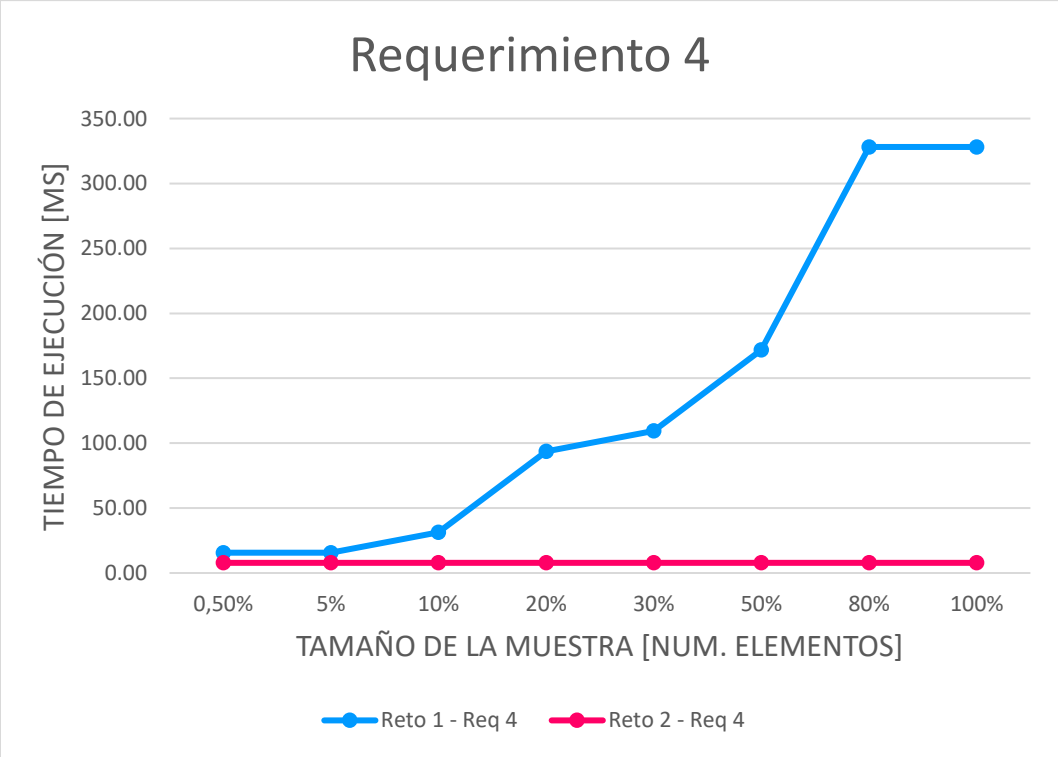
Req 2



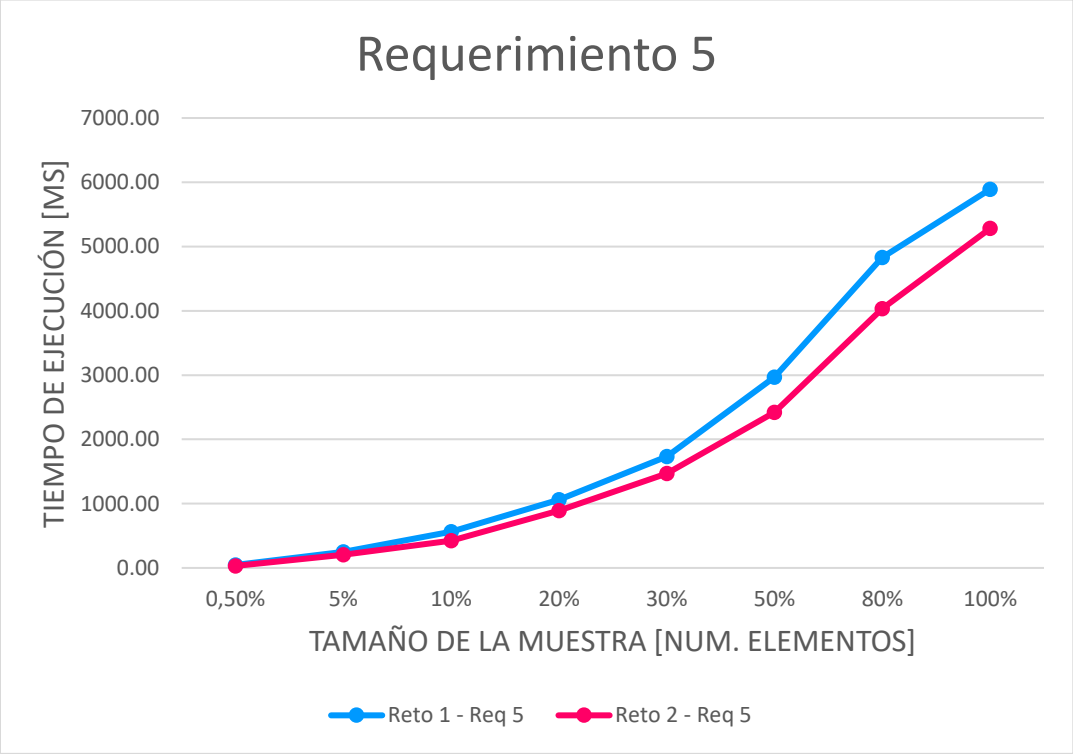
Req 3


Requerimiento 3
250.00
200.00
150.00
100.00
50.00
0.00
TIEMPO DE EJECUCIÓN [MS]
0,50%
5%
10%
20%
30%
50%
80%
100%
TAMAÑO DE LA MUESTRA [NUM. ELEMENTOS]
Reto 1 - Req 3
Reto 2 - Req 3

Req 4


Requerimiento 4
350.00
300.00
250.00
200.00
150.00
100.00
50.00
0.00
TIEMPO DE EJECUCIÓN [MS]
0,50%
5%
10%
20%
30%
50%
80%
100%
TAMAÑO DE LA MUESTRA [NUM. ELEMENTOS]
Reto 1 - Req 4
Reto 2 - Req 4

Req 5


Requerimiento 5
7000.00
6000.00
5000.00
4000.00
3000.00
2000.00
1000.00
0.00
TIEMPO DE EJECUCIÓN [MS]
0,50%
5%
10%
20%
30%
50%
80%
100%
TAMAÑO DE LA MUESTRA [NUM. ELEMENTOS]
Reto 1 - Req 5
Reto 2 - Req 5

Comparación Requerimientos Reto 2