

ANÁLISIS DEL RETO

Xiong yu chen wu, 202116588, x.chen

Marlon David Forero Martinez, 202220738, md.forerom1

Juan Nicolas Suarez Pulido, 202222678, jn.suarez

Requerimiento <<1>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructuras de datos del modelo
Salidas	Actividad económica con mayor total saldo a pagar para cada año
Implementado (Sí/No)	Implementado por Nicolas

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Ordenar por mayor saldo (merge_sort)	$O(n \log n)$
Iterar la lista ordenada	$O(n)$
Paso	$O(\dots)$
TOTAL	$O(n \log n)(n)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)
small	1.88
5 pct	10.02
10 pct	14.45
20 pct	41.21
30 pct	53.02
50 pct	104.00
80 pct	138.15
large	253.33

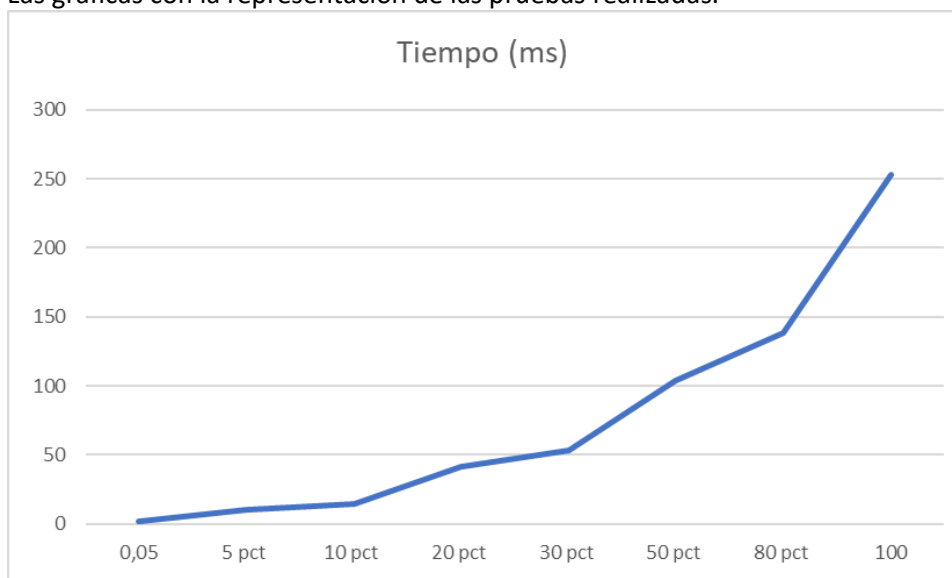
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	1.88
5 pct	Dato2	10.02
10 pct	Dato3	14.45
20 pct	Dato4	41.21
30 pct	Dato5	53.02
50 pct	Dato6	104.00
80 pct	Dato7	138.15
large	Dato8	253.33

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

El proceso consiste en ordenar una lista por el saldo a pagar en orden descendente y, dentro de cada año, ordenarlos de mayor a menor. Luego, se itera a través de la lista y se agrega el primer elemento de cada año a una nueva lista.

Para llevar a cabo este proceso, se debe ordenar la lista primero por año y luego por saldo a pagar en orden descendente. Después, se puede iterar a través de la lista y comparar el año de cada elemento con el año del elemento anterior. Si son diferentes, se agrega el elemento actual a la nueva lista. Si son iguales, se omite el elemento actual y se pasa al siguiente. De esta manera, se obtiene una lista que contiene el primer elemento de cada año, ordenados por mayor saldo a pagar.

Requerimiento <<2>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructura de dato del modelo
Salidas	Actividad económica con mayor total saldo a favor para cada año
Implementado (Sí/No)	Implementado por Nicolas

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Ordenar por mayor saldo (merge_sort)	$O(n \log n)$
Iterar la lista ordenada	$O(n)$
Paso	$O(\dots)$
TOTAL	$O(n \log)(n)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)
small	14.54
5 pct	18.54
10 pct	20.86
20 pct	31.60
30 pct	45.53
50 pct	65.58
80 pct	85.96

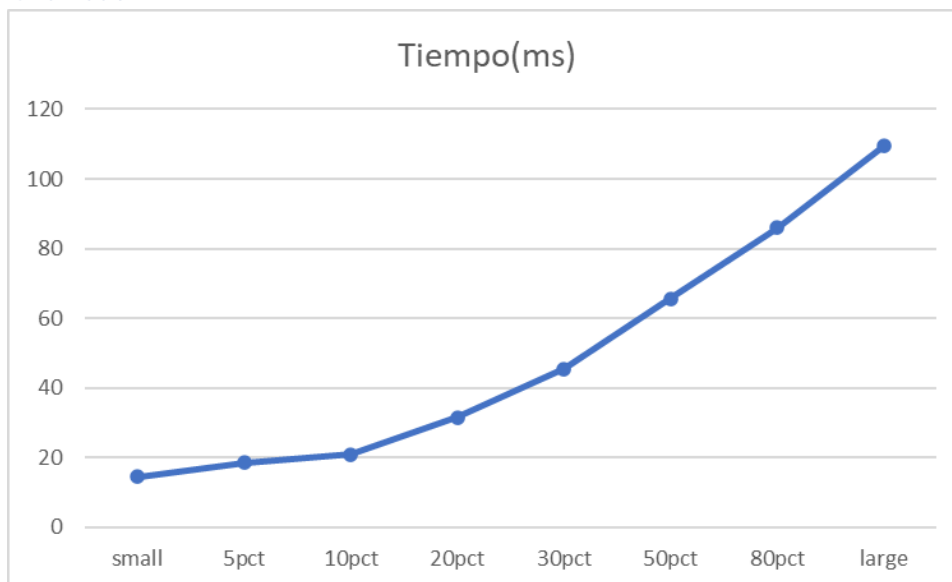
large	109.4
-------	-------

Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	0.05
5 pct	Dato2	0.33
10 pct	Dato3	1.28
20 pct	Dato4	2.54
30 pct	Dato5	4.98
50 pct	Dato6	7.51
80 pct	Dato7	13.81
large	Dato8	25.97

Graficas



Análisis

Organizar por mayor saldo a favor y por año, luego de eso se itera y se agrega una lista el primero de cada año.

Requerimiento <<3>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	No recibe entrada
Salidas	Listas de diccionario
Implementado (Sí/No)	xiong

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Paso 1: ordenar	$O(n \log n)$
Paso 2: iterar	$O(n)$
Paso 3: ordenar	$O(n \log n)$
TOTAL	$O(n) * (n \log n)^2$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)
small	48.86
5 pct	252.72
10 pct	528.72
20 pct	887.05
30 pct	1339.73
50 pct	2210.9
80 pct	3663
large	4451

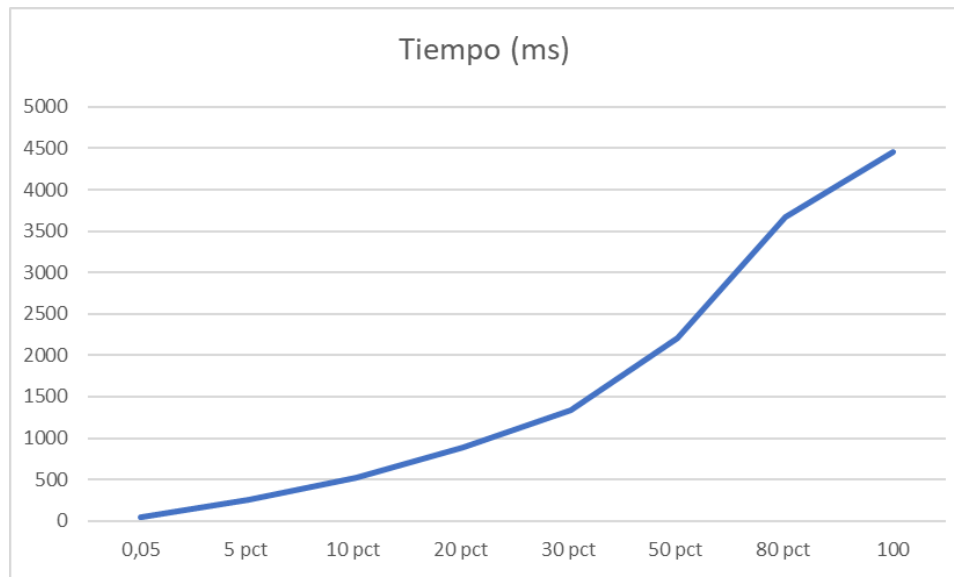
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	48.86
5 pct	Dato2	252.72
10 pct	Dato3	528.72
20 pct	Dato4	887.05

30 pct	Dato5	1339.73
50 pct	Dato6	2210.9
80 pct	Dato7	3663
large	Dato8	4451

Graficas



Análisis

En este requerimiento ordenamos la lista inicial donde contiene toda la información, luego se le itera para acceder a la lista ordenada, después cada iterado pasa por varios filtros y cuando coincidan se les suma lo anterior para agrupar los resultados de los mismos subsectores económicos del mismo año, y al final se ordena nuevamente

Requerimiento <<4>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructura de datos del modelo
Salidas	Tupla con dos ARRAY_LIST
Implementado (Sí/No)	Implementado por Marlon David

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Ordenar por año y código	$O(n \log n)$
Iterar la lista	$O(n)$
En la iteración un ordenado por costos y gastos de nómina.	$O(n \log n)$
TOTAL	$O(n) * (n \log n)^2$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)
small	53.53
5 pct	68.78
10 pct	70.34
20 pct	88.20
30 pct	101.35
50 pct	123.45
80 pct	161.65
large	195.47

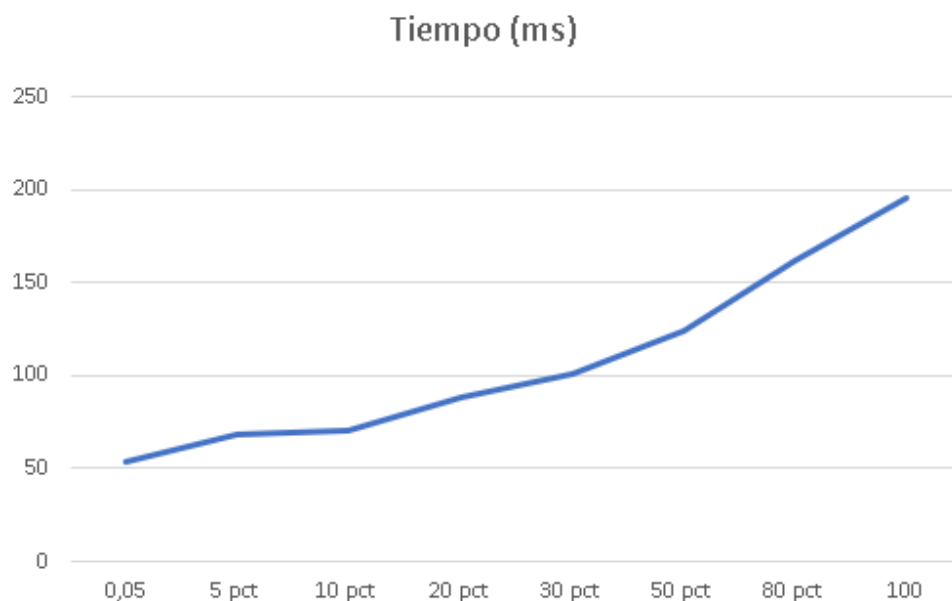
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	53.53
5 pct	Dato2	68.78
10 pct	Dato3	70.34
20 pct	Dato4	88.20
30 pct	Dato5	101.35
50 pct	Dato6	123.45
80 pct	Dato7	161.65
large	Dato8	195.47

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Este proceso implica ordenar una lista por año y código de subsector económico. Luego, se itera a través de la lista y se compara el año y el código de subsector en cada iteración. Si son iguales, se suma el descuento tributario correspondiente a ese subsector económico. Si el código de subsector cambia, se compara la suma de costos y gastos nómina del subsector anterior con la del subsector actual, y se conserva el subsector con la mayor suma. Para volver a comparar códigos en la siguiente iteración, se actualiza la variable de comparación de código.

Requerimiento <<5>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructura de datos del modelo
Salidas	Tupla con dos ARRAY_LIST
Implementado (Sí/No)	Implementado por Nicolas

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
-------	-------------

Ordenar por año y código	$O(n \log n)$
Iterar la lista	$O(n)$
En la iteración un ordenado por el descuento	$O(n \log n)$
TOTAL	$O(n) * (n \log n)^2$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

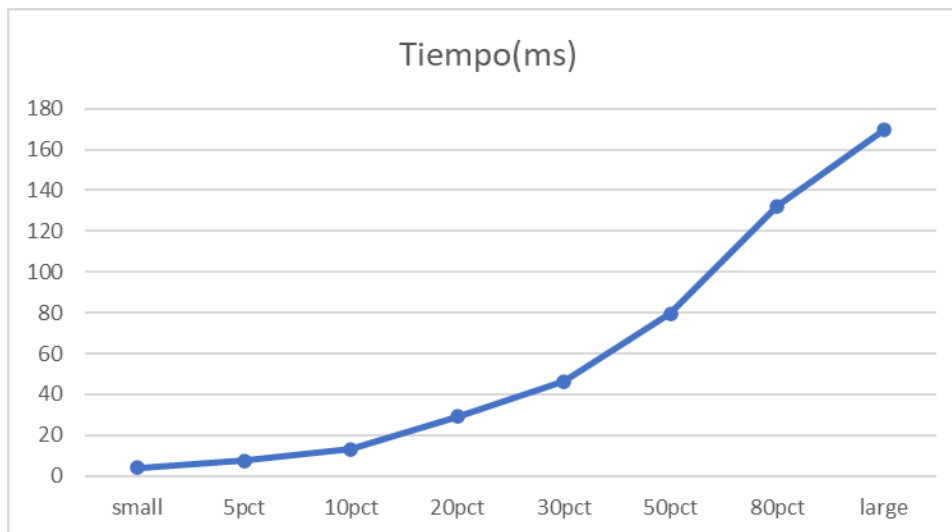
Entrada	Tiempo (ms)
small	4.03
5 pct	7.43
10 pct	12.93
20 pct	28.94
30 pct	46.47
50 pct	79.66
80 pct	132.29
large	169.7

Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	4.03
5 pct	Dato2	7.43
10 pct	Dato3	12.93
20 pct	Dato4	28.94
30 pct	Dato5	46.47
50 pct	Dato6	79.66
80 pct	Dato7	132.29
large	Dato8	169.7

Graficas



Análisis

Se ordena por año y el código de subsector económico y después de esa lista se empieza a iterar y empieza comparando si el año es el mismo y si el código del subsector es el mismo empieza a hacer la sumatoria de descuentos tributarios del subsector económico y encuentra el mayor del subsector en el caso de que sea distinto empieza a revisar, si la suma del subsector económico es mayor a otro subsector dejando solo el mayor de todos y para que vuelva a comparar el código se iguala el código de la iteración con la variable, con la cual se compara el código en cada iteración.

Requerimiento <<6>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructura de datos del modelo y el año
Salidas	Un array_list con tres array_list en el interior
Implementado (Sí/No)	Implementado por Nicolas

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Ordenar por código (mergeSort)	$O(n \log n)$
Iterar la lista	$O(n)$
Iterar	$O(n)$
TOTAL	$O(n \log n)(n)^2$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

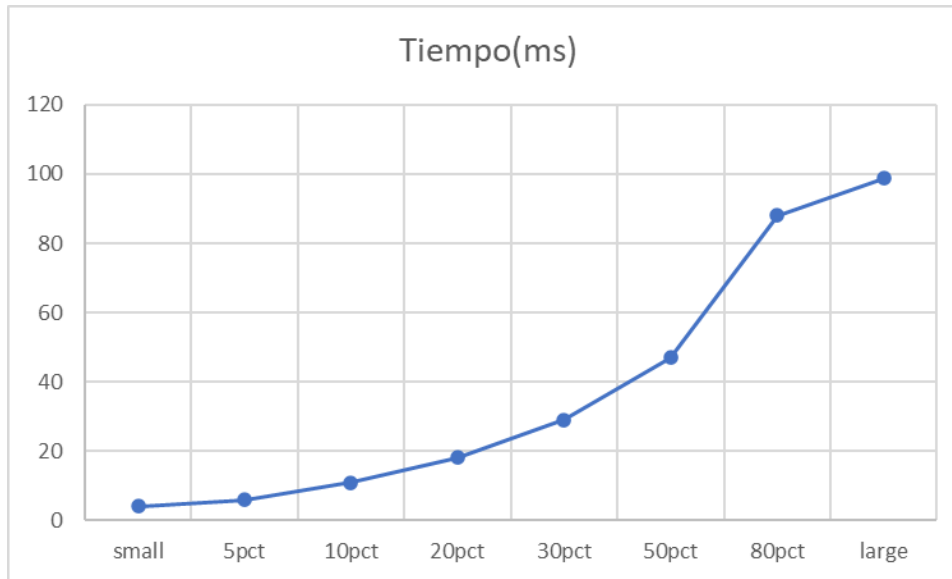
Entrada	Tiempo (ms)
small	4.00
5 pct	5.97
10 pct	10.94
20 pct	18.03
30 pct	28.94
50 pct	46.99
80 pct	88.08
large	98.85

Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	0.05
5 pct	Dato2	0.33
10 pct	Dato3	1.28
20 pct	Dato4	2.54
30 pct	Dato5	4.98
50 pct	Dato6	7.51
80 pct	Dato7	13.81
large	Dato8	25.97

Graficas



Análisis

Análisis de resultados de la implementación, tener cuenta las pruebas realizadas y el análisis de complejidad.

Primero se organiza por el código del sector económico, después se filtra por el año indicado por parámetro, luego de eso se itera haciendo que el código siempre sea igual al del primer elemento de la lista y se empieza a sumar todas las cosas necesarias, y se revisa si el código de la actividad económica es el mismo. Cuando el código distinto del anterior en la anterior iteración se agrega a un diccionario, se hace con diccionario para poder plasmarlo en una tabla de una mejor forma y se iguala el código de esa iteración al código de comparación para que se pueda comparar en la siguiente iteración.

Toca tener en cuenta que los tiempos tomados se usaron con el año 2020

Requerimiento <<7>>

Plantilla para el documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Estructuras de datos del modelo, top N, año inicial, año final
Salidas	La lista con el top N, si no existe retorna lo que hay
Implementado (Sí/No)	Implementado por xiong

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Paso 1: ordenar	$O(n \log n)$
Paso 2: iterar	$O(n)$
Paso 3: agregar	$O(1)$
Paso4: ordenar	$O(n \log n)$
TOTAL	$O((n \log)^2 * n)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)
small	0.78
5 pct	3.72
10 pct	7.14
20 pct	16.44
30 pct	26.64
50 pct	47.24
80 pct	79.68
large	109.67

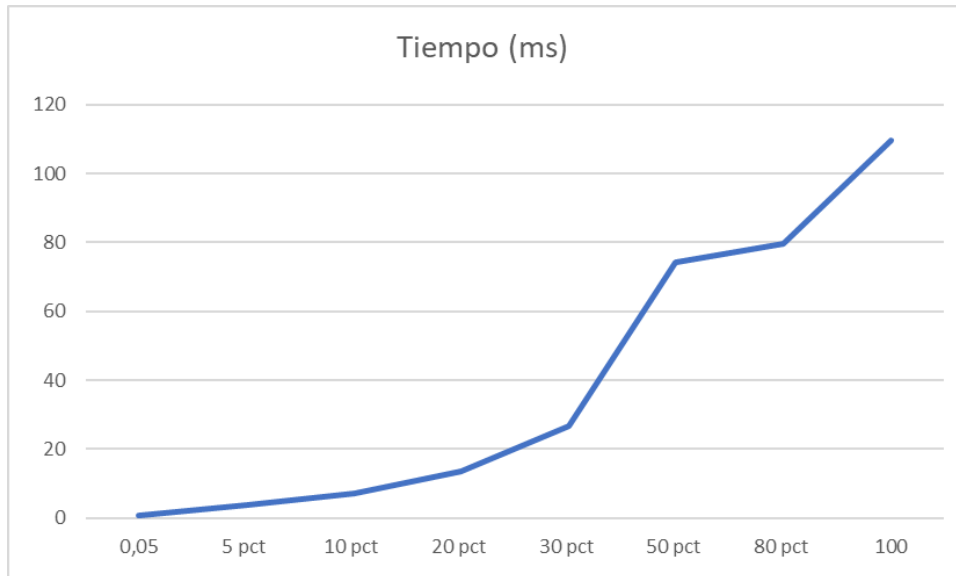
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Muestra	Salida	Tiempo (ms)
small	Dato1	0.05
5 pct	Dato2	0.33
10 pct	Dato3	1.28
20 pct	Dato4	2.54
30 pct	Dato5	4.98
50 pct	Dato6	7.51
80 pct	Dato7	13.81
large	Dato8	25.97

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

En este requerimiento se implementó array list para guardar la información, primero ordene la lista por defecto donde esta guardada la información, luego itero para que accede a cada uno de los sets de datos y para después pasar por unos filtros donde determina si es acorde o no a lo que se necesita, el filtrado es año inicial y año final. Cuando coincide se evalúa si tiene otros sets con el mismo código de actividad y sumarlos, se guarda en otra lista y finalmente se le ordena nuevamente.