

ANÁLISIS DEL RETO

Santiago Alberto Quiroz Pintor, s.quirozp@uniandes.edu.co. 202216453.

Alberto Mario Pertuz, a.pertuz@uniandes.edu.co. 202025856.

Alejandro Guerra, a.guerra@uniandes.edu.co. 202122640

Requerimiento <<CARGA>>

Descripción

Se crean 2 grafos los cuales uno será de distancias y el otro de pesos también se crean listas auxiliares y unos mapas de consulta de información para facilitar la búsqueda de info

Requerimiento <<1>>

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Punto de origen (una localización geográfica con latitud y longitud) y Punto de destino (una localización geográfica con latitud y longitud)
Salidas	El total de vértices que contiene el camino encontrado.
Implementado (Sí/No)	Si se implementó

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Como genera un subgrafo a través de dfs su complejidad se aumenta a	$O(v+e) * \log(v)$
Se generará el camino del dfs y el vertice ingresado por el usuario	$O(v)$
Genera un for de iteración sobre la carga del Search por lo cual su complejidad aumenta a	$O(N)$
TOTAL	$O(n * v^2 * \log(v))$

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: comparendo 2019	Tiempo (s)
Toma 1	4808.995500
Toma 2	4994.503500
Toma 3	4979.084600
Toma 4	5179.486800
Toma 5	6908.861800
Toma 6	7206.560900
Toma 7	9049.182700
Toma 8	5887.072600
Toma 9	6943.753400
Toma 10	6736.309500

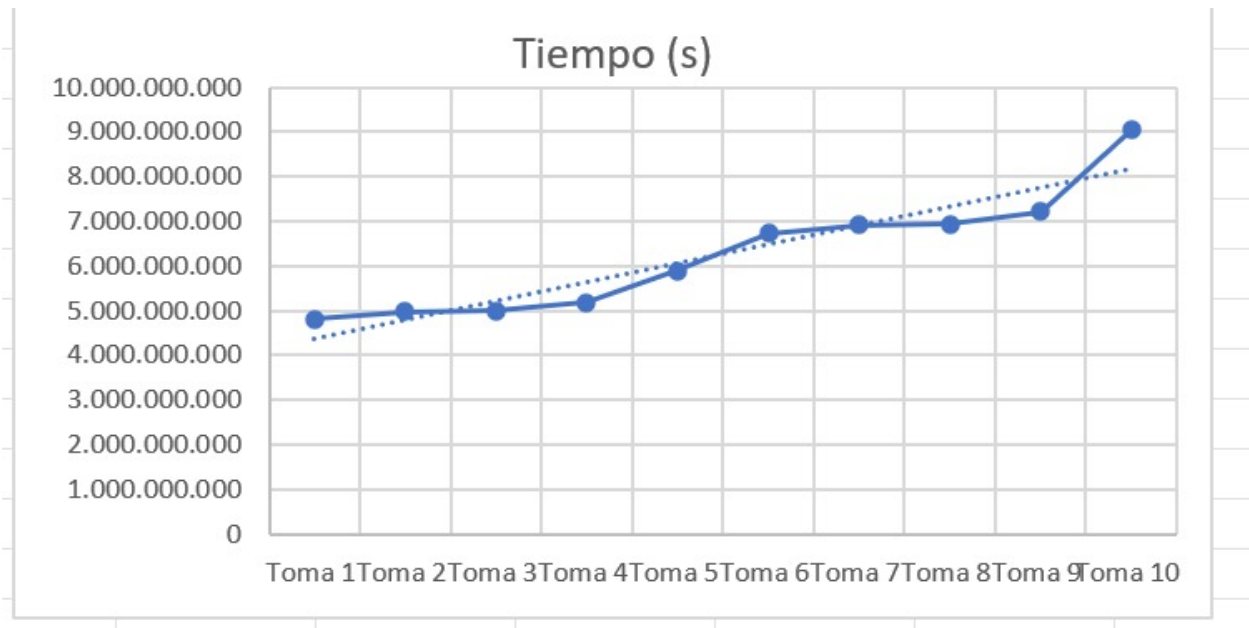
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Entrada: comparendo 2019	Tiempo (s)
Toma 1	4.808.995.500
Toma 2	4.994.503.500
Toma 3	4.979.084.600
Toma 4	5.179.486.800
Toma 5	6.908.861.800
Toma 6	7.206.560.900
Toma 7	9.049.182.700
Toma 8	5.887.072.600
Toma 9	6.943.753.400
Toma 10	6.736.309.500

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Como la complejidad de la implementación de bfs y sus funciones, etc genera gran impacto en el tiempo de ejecución del requerimiento por lo anterior se puede decir que afecta en la toma temporal dependiendo de la cantidad de datos que existan entre los dos vértices encontrados.

Requerimiento <<2>>

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Punto de origen (una localización geográfica con latitud y longitud) y Punto de destino (una localización geográfica con latitud y longitud)
Salidas	La distancia total que tomará el camino entre el punto de encuentro de origen y el de destino, El total de vértices que contiene el camino encontrado
Implementado (Sí/No)	Si se implementó.

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

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

Paso 1	$O(v+e)$
Paso 2	$O(v)$
TOTAL	$O(v^2)$

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: comparendo 2019	Tiempo (s)
Toma 1	4.917.890.600
Toma 2	5.027.324.700
Toma 3	5.130.920.100
Toma 4	5.264.193.800
Toma 5	5.310.007.500
Toma 6	5.393.834.600
Toma 7	5.730.430.900
Toma 8	6.484.887.000
Toma 9	6.930.803.800
Toma 10	6.971.832.400

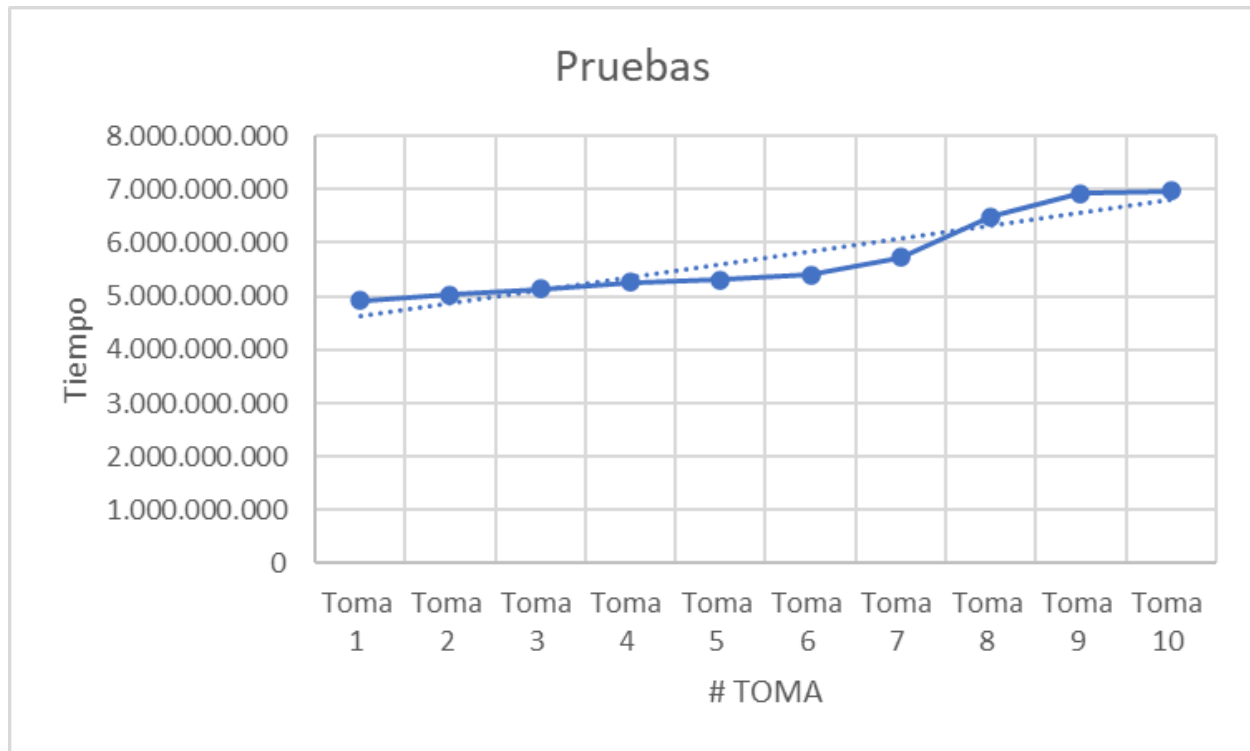
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Entrada: comparendo 2019	Tiempo (s)
Toma 1	4.917.890.600
Toma 2	5.027.324.700
Toma 3	5.130.920.100
Toma 4	5.264.193.800
Toma 5	5.310.007.500
Toma 6	5.393.834.600
Toma 7	5.730.430.900
Toma 8	6.484.887.000
Toma 9	6.930.803.800
Toma 10	6.971.832.400

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Como se puede evidenciar cada momento de la toma ampliaremos el rango lo cual hacía que cada vez implementará más recursos y tiempos en arrojar una respuesta, pero no fue muy alejada de la realidad. Por lo que evidenciamos que la implementación de BFS aumenta la complejidad pero genera una fácil implementación a los requerimientos

Requerimiento <<3>>

Descripción

El requerimiento se abordó desde la carga de datos donde se hizo un mapa donde se contenía la información total de localidades como llaves luego se buscan y se filtran y se empieza a sacar las llaves se agregan y se recorren en dfs para luego hacer un recorrido Dijkstra

Entrada	Cantidad de cámaras y la localidad de consulta
Salidas	Devuelve el total del peso de los arcos entre el primer vértice y el segundo vértice para luego multiplicarlo por el precio del kilómetro de fibra óptica
Implementado (Sí/No)	Si se implementó por Santiago Quiroz Pintor

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Paso 1: Se recorre el mapa y se filtra la información para después agregarla a una lista nativa y generamos un primer bsf a partir del primer nodo ya organizado	$O(n)$ -> Si planteamos que todos los datos son de la misma localidad
Paso 2: Se genera el bfs y empieza el recorrido dijkstra a partir del camino de origen ya puesto (Solo busca los M vertices ingresados por el usuario)	$O((v * e) * \log v)$
Se printea	$O(\dots)$
TOTAL	$O(n * (v * e) * \log(v))$

Pruebas Realizadas

Se usaron los datos específicos del reto (Dados por los implementadores) pero no fue posible la toma de datos debido a que se crashea por falta de RAM

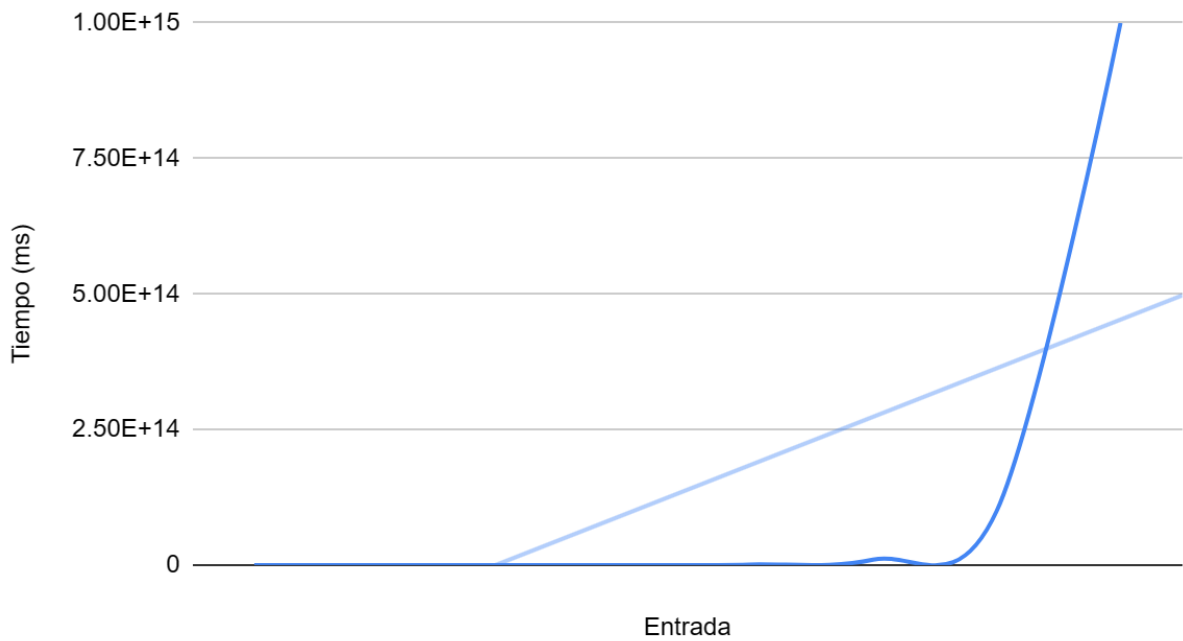
Entrada: cantidad comparendos	Tiempo (s)
Se intento aproximadamente 3 veces	
Pero se crasheo el computador por falta de RAM	

Tablas de datos

Personalmente considero que es un error de recursos de mi máquina debido a que solo poseé 4 gb de RAM pero está sujeto a evaluación en la sustentación en caso de que se vea alguna implementación mal hecha, también el uso del algoritmo Dijkstra es de gran problema debido a la cantidad de recursos debido a la complejidad del algoritmo lo cual pudo ser el problema por el cual no arroja una respuesta el algoritmo, ya que son demasiados vértices y arcos en el bfs que ya de por sí tiene una complejidad elevada.

Graficas

Requerimiento 3



Análisis

Se estima que su complejidad pueda ser mayor y que la implementación del BFS estuvo correcta pero debido a que se quedó aproximadamente 40 minutos sin dar respuesta por lo anterior afirmó que es un error de mi máquina.

Personalmente considero que es un error de recursos de mi máquina debido a que solo poseé 4 gb de RAM pero está sujeto a evaluación en la sustentación en caso de que se vea alguna implementación mal hecha, también el uso del algoritmo Dijkstra es de gran problema debido a la cantidad de recursos debido a la complejidad del algoritmo lo cual pudo ser el problema por el cual no arroja una respuesta el algoritmo, ya que son demasiados vértices y arcos en el bfs que ya de por sí tiene una complejidad elevada.

Requerimiento <<5>>

Descripción

Se recorre la lista y se meten los vertices en una lista nativa para después buscarlos en el grafo y lograr hallar los vertices con mayor cantidad de accidentes e implementar las cámaras

Entrada	Cantidad de cámaras y el tipo de carro de consulta
---------	--

Salidas	Devuelve el total del peso de los arcos entre el primer vértice y el segundo vértice para luego multiplicarlo por el precio del kilómetro de fibra óptica
Implementado (Sí/No)	Si se implementó por Alberto Mario Pertuz

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Paso 1: Se recorre la lista y se filtra la información para después agregarla a una lista nativa y generamos un primer bfs a partir del primer nodo ya organizado	$O(n^2)$ -> Si planteamos que todos los datos son de la misma localidad debido al for indentado
Paso 2: Se genera el bfs y empieza el recorrido dijkstra a partir del camino de origen ya puesto (Solo busca los M vértices ingresados por el usuario)	$O((v * e) * \log v)$
Se printea	$O(1)$
TOTAL	$O(n[2 * (v * e) * \log(v)])$

Pruebas Realizadas

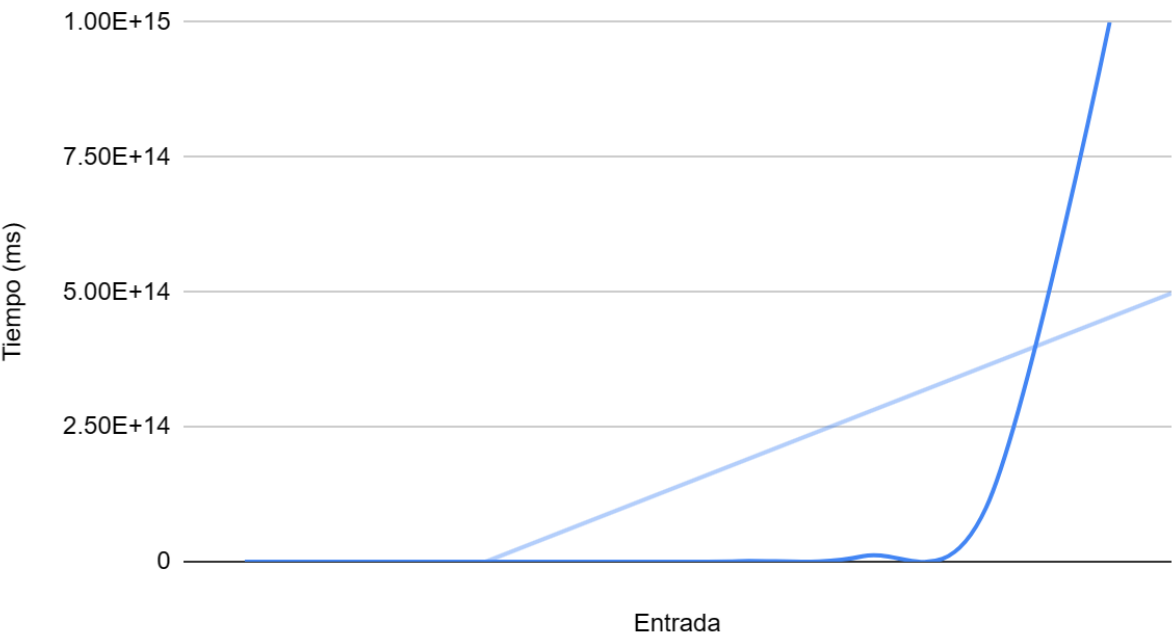
Se usaron los datos específicos del reto (Dados por los implementadores) pero no fue posible la toma de datos debido a que se crashea por falta de RAM -> Sufrimos el mismo problema que en el req 3

Entrada: cantidad comparendos	Tiempo (s)
Se intento aproximadamente 3 veces	
Pero se crasheo el computador por falta de RAM	

Tablas de datos

Graficas

Requerimiento 5



Análisis

En cuanto al tiempo decimos que es congruente aunque no nos brinde una respuesta se implemento de buena manera, los prints previos lo demuestran, pues genera la información que se buscaba para el uso del BFS

Requerimiento <<6>>

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	La cantidad de comparendos que se desea responder
Salidas	El total de vértices del camino, Los vértices incluidos (identificadores)., Los arcos incluidos (Id vértice inicial e Id vértice final)., La cantidad de kilómetros del camino
Implementado (Sí/No)	Si se implementó.

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Obtener datos, y entrar en maps son constantes pero el get_element hace que genere una complejidad directa de	$O(n)$
Uso del algoritmo Dijkstra lo cual vuelve su complejidad	$O(V+E)\log V$
<i>El resto son inicializaciones de variables</i>	$O(1)$
TOTAL	$O(v * (V+E)\log V)$

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: cantidad comparendos	Tiempo (s)
2	1.071200
4	2.853100
6	3.586800
8	5.295300
10	5.200700
12	6.474800
14	8.817800
16	8.920300
18	9.514800
20	8.532300

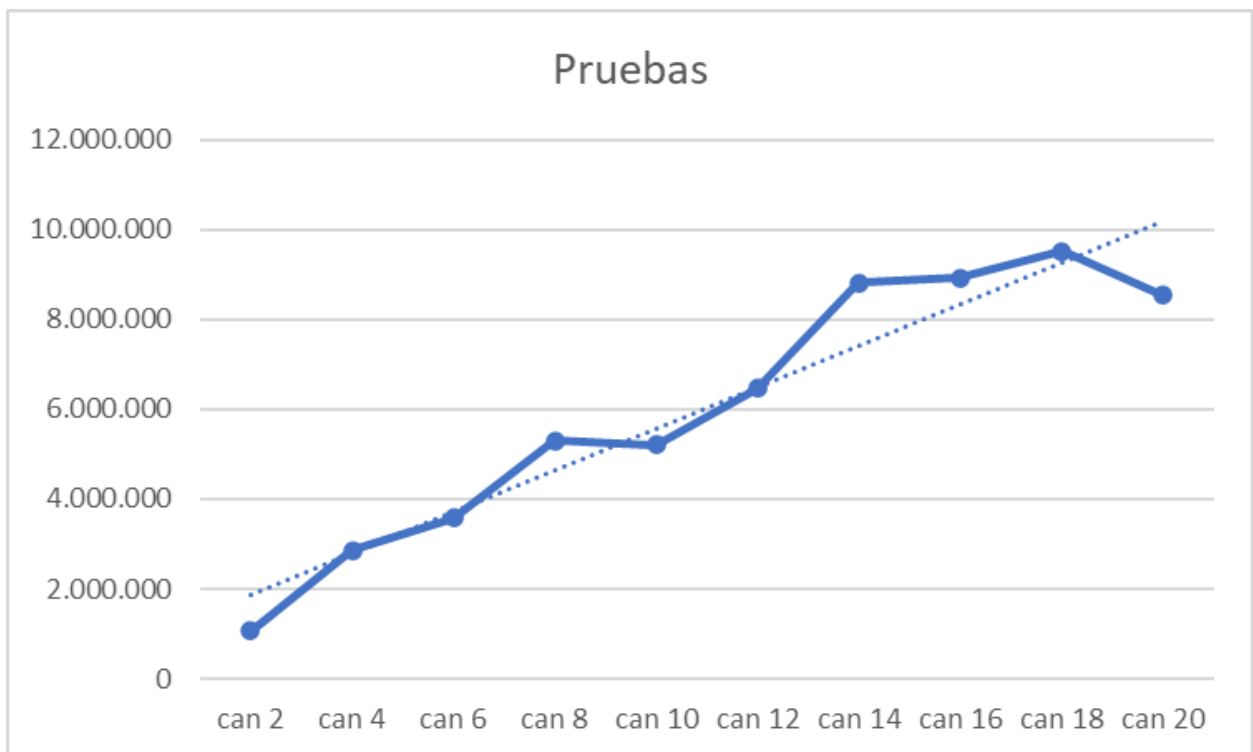
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Entrada: cantidad comparendos	Tiempo (s)
can 2	1.071.200
can 4	2.853.100
can 6	3.586.800
can 8	5.295.300
can 10	5.200.700
can 12	6.474.800
can 14	8.817.800
can 16	8.920.300
can 18	9.514.800
can 20	8.532.300

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Segun gráfica podemos ver un comportamiento exponencial debido al aumento en la latitud y longitud en cada toma.

Requerimiento <<7>>

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

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	Punto de origen (una localización geográfica con latitud y longitud) y Punto de destino (una localización geográfica con latitud y longitud)
Salidas	El total de vértices del camino, Los vértices incluidos (identificadores), Los arcos incluidos (Id vértice inicial e Id vértice final), La cantidad de comparendos del camino, La cantidad de kilómetros del camino
Implementado (Sí/No)	Si se implementó

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Paso 1	$O(n)$
Paso 2	$O(V+E)\log V$
TOTAL	$O(v * (V+E)\log V)$

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: comparendo 2019	Tiempo (s)
Toma 1	4808.995500
Toma 2	4994.503500
Toma 3	4979.084600
Toma 4	5179.486800
Toma 5	6908.861800
Toma 6	7206.560900
Toma 7	9049.182700
Toma 8	5887.072600
Toma 9	6943.753400
Toma 10	6736.309500

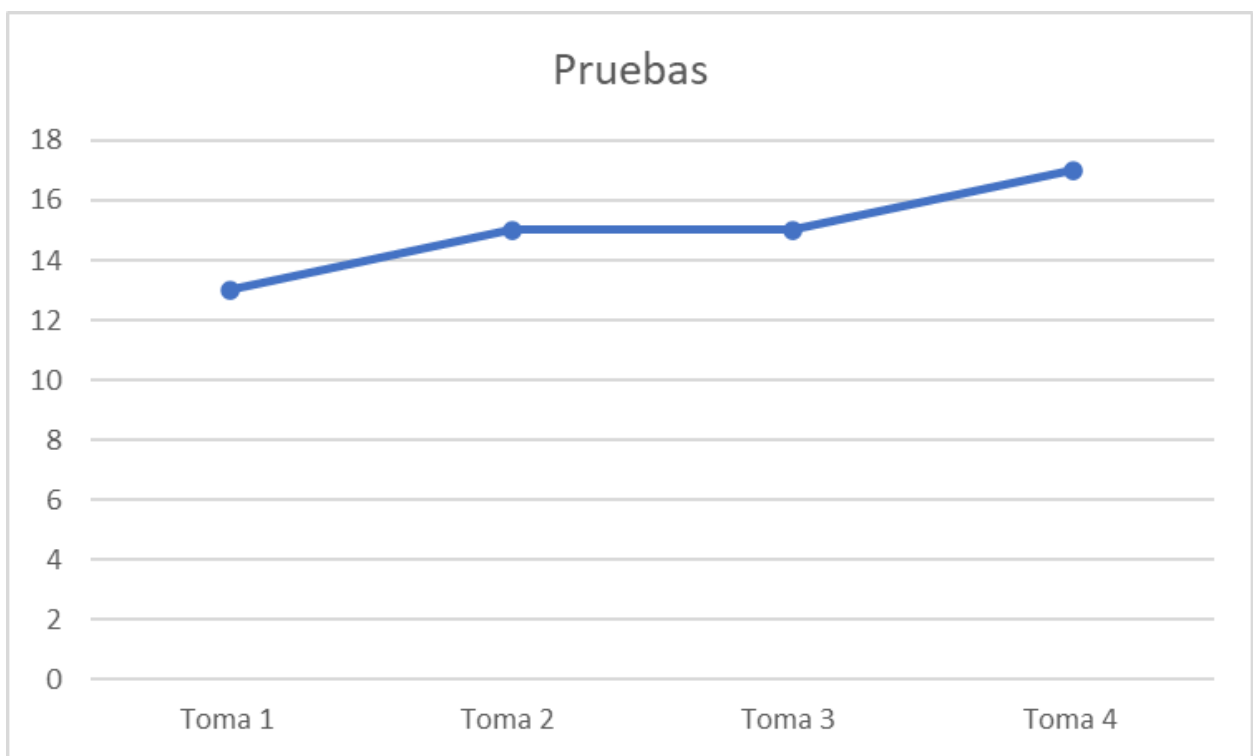
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Entrada: comparendo 2019	Tiempo (min)
Toma 1	13
Toma 2	15
Toma 3	15
Toma 4	17

Graficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

se demora mas de 20 minutos en tomar cada prueba, por eso la corta cantidad de estas, se ve un comportamiento exponencial al cambio mayor en las variables. Puede ser un error de mi maquina el hecho de que no corra

