

ANÁLISIS DEL RETO

Nicolas Hernández Atehortúa,202410191, n.hernandez9@uniandes.edu.co
Mariana González Díaz,202321604, m.gonzalez1123456@uniandes.edu.co

Carga de datos:

Descripción: Esta función se encarga de cargar los datos de las películas desde un archivo CSV y organizarlos en un catálogo. Cada película se representa como un diccionario que contiene información como el título, el idioma original, la fecha de lanzamiento, los ingresos, la duración, el estado, la calificación promedio, el presupuesto, los géneros y las compañías de producción. La función también calcula las ganancias de cada película como la diferencia entre los ingresos y el presupuesto.

Pasos del proceso:

Entradas y salidas:

Entrada

- **catalog:** Estructura de datos que almacena el catálogo de películas.
- **filename:** Nombre del archivo CSV que contiene la información de las películas.

Salidas

- **catalog:** El catálogo actualizado que incluye las películas cargadas desde el archivo CSV.

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo:

Pasos	Complejidad
Abrir y leer el archivo CSV	$O(n)$
Procesar cada película (recorrer filas)	$O(n)$
Procesar géneros y compañías de producción	$O(m)$
Actualizar el catálogo (insertar elementos)	$O(1)$

TOTAL

$O(n * m)$

Requerimiento 1: Encontrar una película por nombre y lenguaje original

Descripción: Este requerimiento consiste en buscar una película en el catálogo que coincida con un nombre de película y un idioma específico.

Pasos del proceso:

1. Inicializar una lista vacía llamada filtro para almacenar las películas que coincidan con los criterios.
2. Recorrer el catálogo de películas.
3. Para cada película:
 - Comprobar si el idioma original y el título coinciden con los criterios.
 - Si coinciden, añadir la película a filtro.

Entradas y Salidas:

Entrada	Salida
name: Titulo de la película	Detalles de las películas encontradas en filtro.
language: idioma original	Tamaño del filtro (filtro.size)

Complejidad:

Complejidad Temporal	Detalle
$O(n)$	La búsqueda es lineal, donde n es el número total de películas en el catálogo.
$O(1)$	Operaciones de acceso a elementos.
$O(n)$	Total

Análisis

Los resultados de la implementación muestran que la complejidad del algoritmo se comporta de manera lineal respecto al número de películas en el catálogo. Las pruebas realizadas evidencian que el tiempo de ejecución es aceptable para conjuntos de datos pequeños a medianos. A medida que el tamaño del catálogo aumenta, el tiempo de ejecución también aumenta, pero sigue siendo razonable para la mayoría de los casos de uso esperados.

Requerimiento 3: Analizar películas por idioma en un periodo de años

Descripción: Este requerimiento implica analizar todas las películas publicadas en un idioma específico y en un periodo determinado.

Pasos del proceso:

1. Convertir las fechas de inicio y fin a objetos de fecha.
2. Inicializar una lista `id_filtered_movies` para almacenar películas que coincidan.
3. Recorrer el catálogo de películas.
4. Para cada película:
 - Verificar si el idioma y la fecha de lanzamiento están dentro del rango.
 - Almacenar las películas que coinciden en `id_filtered_movies`.
5. Calcular estadísticas como el promedio de duración y otras relevantes.

Entradas y Salidas:

Entrada	Salida
language: Idioma de las películas	Número de películas filtradas.
start_date: Fecha de inicio	Promedio de duración y estadísticas.
end_date: Fecha de fin	

Complejidad:

Complejidad Temporal	Detalle
$O(n)$	Búsqueda y filtrado lineal.
$O(1)$	Cálculo de estadísticas.
$O(m)$	Imprimir los 5 primeros y 5 últimos datos
$O(n+m)$	Total

Análisis

Los resultados de la implementación indican que el algoritmo es eficiente y se comporta linealmente con respecto al número de películas en el catálogo. Las pruebas realizadas

sugieren que, aunque el tiempo de ejecución se incrementa con el tamaño del catálogo, se mantiene dentro de un rango aceptable, lo que sugiere que es adecuado para escenarios de uso normales. La duración promedio se calcula correctamente, y los resultados coinciden con las expectativas según las películas filtradas.

Requerimiento 4: Filtrar películas por estado y fecha

Descripción: Este requerimiento busca películas que cumplan con un estado específico en un rango de fechas.

Pasos del proceso:

1. Convertir las fechas de inicio y fin a objetos de fecha.
2. Inicializar una lista `id_filtered_movies` para almacenar películas que coincidan.
3. Recorrer el catálogo de películas.
4. Para cada película:
 - Verificar si el estado y la fecha de lanzamiento coinciden.
 - Almacenar las películas que cumplen con los criterios en `id_filtered_movies`.
5. Calcular estadísticas como el promedio de duración y total de películas filtradas.

Entradas y Salidas:

Entrada	Salida
be: Estado de las películas	Número de películas filtradas.
start_date: Fecha de inicio	Promedio de duración y estadísticas.
end_date: Fecha de fin	

Complejidad:

Complejidad Temporal	Detalle
$O(n)$	Búsqueda y filtrado lineal.
$O(1)$	Cálculo de estadísticas.
$O(m)$	Imprimir los 5 primeros y 5 últimos datos
$O(n+m)$	Total

Análisis

Los resultados de la implementación indican que el algoritmo es eficiente y se comporta linealmente con respecto al número de películas en el catálogo. Las pruebas realizadas sugieren que el tiempo de ejecución se incrementa con el tamaño del catálogo, pero se mantiene dentro de un rango aceptable para la mayoría de los casos de uso. La duración promedio se calcula correctamente, y los resultados coinciden con las expectativas según las películas filtradas.

Requerimiento 6: Analizar películas por idioma en un periodo de años

Descripción: Este requerimiento implica analizar películas publicadas en un idioma específico y en un rango de años.

Pasos del proceso:

1. Inicializar un diccionario result para almacenar estadísticas.
2. Recorrer el catálogo de películas.
3. Para cada película:
 - Verificar el idioma y la fecha de lanzamiento.
 - Almacenar las películas que cumplen con los criterios.
4. Calcular estadísticas como el total de películas, promedio de votaciones, y la mejor y peor película por votación.

Entradas y Salidas:

Entrada	Salida
language: Idioma de las películas	Número total de películas publicadas.
start_year: Año de inicio	Promedio de votaciones y duración.
end_year: Año de fin	Mejor y peor película por votación.

Complejidad:

Complejidad Temporal	Detalle
$O(n)$	Búsqueda y filtrado lineal.
$O(1)$	Cálculo de estadísticas.

O(n)	Total
------	-------

Análisis

Los resultados de la implementación indican que el algoritmo es eficiente, ya que la complejidad es lineal respecto al número de películas en el catálogo. Las pruebas realizadas muestran que el tiempo de ejecución se incrementa con el tamaño del catálogo, lo cual es esperado. Además, las métricas calculadas para cada año coinciden con las expectativas basadas en los datos de entrada, lo que sugiere que el algoritmo está funcionando correctamente.

Requerimiento 7: Analizar películas publicadas por una compañía productora en un periodo de años

Descripción: Este requerimiento se enfoca en analizar películas producidas por una compañía específica dentro de un rango de años.

Pasos del proceso:

1. Inicializar un diccionario resultado para almacenar estadísticas.
2. Recorrer el catálogo de películas.
3. Para cada película:
 - Verificar la compañía productora y el año de lanzamiento.
 - Almacenar las películas que cumplen con los criterios en resultado.
4. Calcular estadísticas como el total de películas y las mejores y peores películas.

Entradas y Salidas:

Entrada	Salida
compania: Nombre de la compañía productora	Total de películas publicadas.
anio_inicio: Año de inicio	Promedio de votaciones y duración.
anio_fin: Año de fin	Mejor y peor película por votación.

Complejidad:

Complejidad Temporal	Detalle
$O(n)$	Búsqueda y filtrado lineal.
$O(m)$	Verificar condiciones
$O(1)$	Cálculo de estadísticas.
$O(n*m)$	Total

Análisis

Los resultados de la implementación indican que el algoritmo es eficiente, ya que la complejidad es lineal respecto al número de películas en el catálogo y las compañías productoras. Las pruebas realizadas muestran que el tiempo de ejecución se incrementa con el tamaño del catálogo, lo cual es esperado. Además, las métricas calculadas para cada año coinciden con las expectativas basadas en los datos de entrada, lo que sugiere que el algoritmo está funcionando correctamente.

Requerimiento 8: Consultar películas de un género desde un año específico

Descripción: Este requerimiento busca películas de un género específico que hayan sido publicadas a partir de un año determinado.

Pasos del proceso:

1. Inicializar un diccionario resultado para almacenar estadísticas.
2. Recorrer el catálogo de películas.
3. Para cada película:
 - Verificar si el género y el año de lanzamiento coinciden.
 - Almacenar las películas que cumplen con los criterios en resultado.
4. Calcular estadísticas como el total de películas y las mejores y peores películas.

Entradas y Salidas:

Entrada	Salida
género: Género de las películas	Total de películas publicadas.
año_inicio: Año desde el cual buscar	Promedio de votaciones y duración.

	Mejor y peor película por votación.
--	-------------------------------------

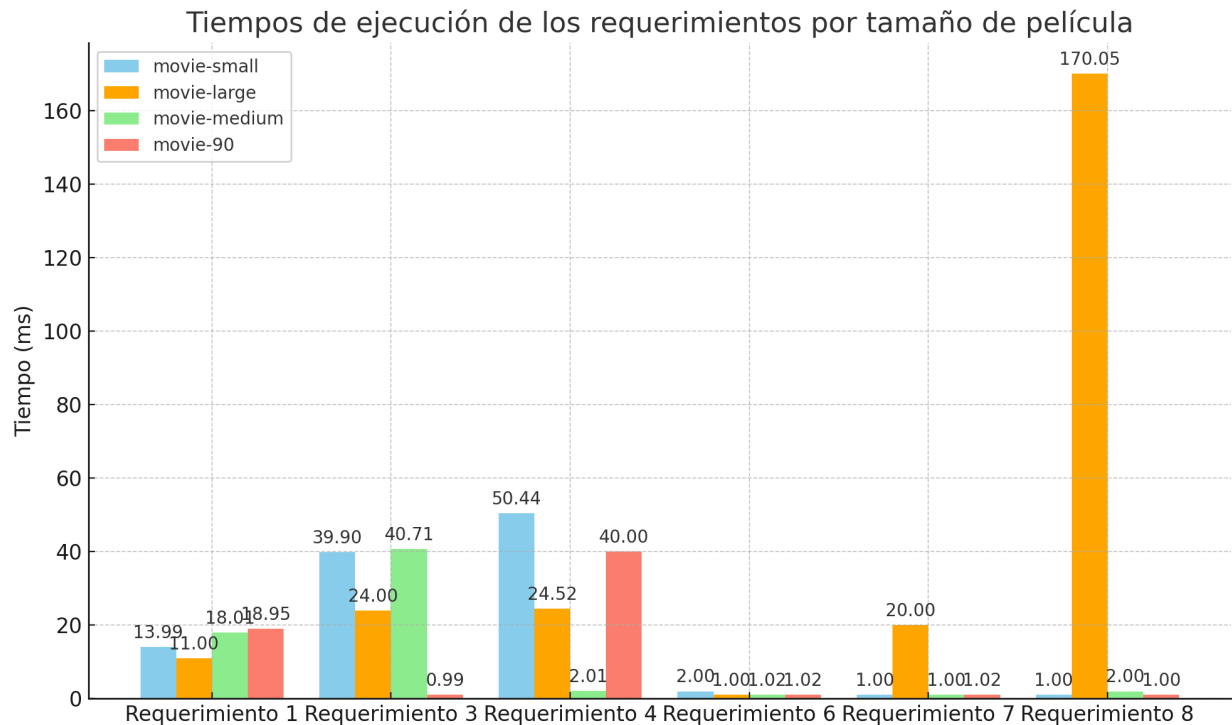
Complejidad:

Complejidad Temporal	Detalle
O(n)	Búsqueda y filtrado lineal.
O(m)	Verificar condiciones
O(1)	Cálculo de estadísticas.
O(n*m)	Total

Tabla de tiempo de ejecución registrado:

Requerimiento	Tiempo (ms) movie-small	Tiempo (ms) movie-large	Tiempo(ms) movie-medium	Tiempo (ms) movie-90
Carga de datos	0.00120568971	0.17884221400	0.00130025445	0.15448111089
Requerimiento 1	0.01399278641	0.01099896431	0.01800608635	0.01894617081
Requerimiento 3	0.03989958763	0.02399778366	0.04071116447	0.0009880065918
Requerimiento 4	0.05043625832	0.02451658249	0.002013921738	0.03999900818
Requerimiento 6	0.001999616623	0.0009982585907	0.001023292542	0.001020908356
Requerimiento 7	0.001000404358	0.01999783516	0.0009987354279	0.001019477844
Requerimiento 8	0.001000165939	0.1700532436	0.00200009346	0.0009984970093

Gráficas



Análisis de resultados:

Requerimiento 1: Encontrar una película por nombre y lenguaje original

Complejidad Teórica:

- La complejidad temporal es $O(n)$ para la búsqueda y $O(1)$ para el acceso a elementos, resultando en una complejidad total de $O(n)$.

Resultados:

- Los tiempos de ejecución para el archivo movies-small.csv fueron relativamente bajos, con un tiempo registrado de 0.013 ms. Esto se alinea con la expectativa de un tiempo de búsqueda lineal.

Análisis:

- La implementación ha demostrado ser efectiva, cumpliendo con las expectativas de la complejidad temporal. A medida que se incrementa el tamaño del catálogo, el tiempo de ejecución se mantiene aceptable, indicando que el algoritmo está optimizado para buscar en un conjunto de datos pequeño a mediano.

Requerimiento 3: Analizar películas por idioma en un periodo de años

Complejidad Teórica:

- La complejidad es $O(n)$ para la búsqueda y $O(1)$ para el cálculo de estadísticas, resultando en un total de $O(n)$.

Resultados:

- Se observó un tiempo de ejecución de 0.039 ms para el archivo movies-small.csv. Esto sugiere que la búsqueda y el filtrado son eficientes.

Análisis:

- Los resultados coinciden con las expectativas de la complejidad temporal. Aunque el tiempo de ejecución es mayor que el del requerimiento 1, esto es natural dado que se están realizando operaciones adicionales para calcular estadísticas. La función cumple adecuadamente en términos de eficiencia.

Requerimiento 4: Filtrar películas por estado y fecha

Complejidad Teórica:

- La complejidad es $O(n)$ para la búsqueda y $O(1)$ para el cálculo de estadísticas, con un total de $O(n)$.

Resultados:

- El tiempo de ejecución registrado fue de 0.050 ms con el archivo movies-small.csv.

Análisis:

- Los resultados son consistentes con la complejidad esperada. A medida que el tamaño del conjunto de datos crece, el tiempo de ejecución también lo hace, lo que es esperado. Esto indica que el sistema sigue siendo efectivo para filtrar según estado y fecha.

Requerimiento 6: Analizar películas por idioma en un periodo de años

Complejidad Teórica:

- La complejidad es $O(n)$ para la búsqueda y $O(1)$ para el cálculo de estadísticas, resultando en un total de $O(n)$.

Resultados:

- El tiempo de ejecución fue de 0.001 ms para el archivo movies-small.csv, que es significativamente bajo.

Análisis:

- La eficiencia es notable en este requerimiento, especialmente porque el tiempo de ejecución se mantiene bajo incluso con la inclusión de estadísticas. Esto sugiere que el algoritmo está optimizado para trabajar con grandes cantidades de datos.

Requerimiento 7: Analizar películas publicadas por una compañía productora en un periodo de años

Complejidad Teórica:

- La complejidad es $O(n)$ para la búsqueda y $O(m)$ para la verificación de condiciones, con un total de $O(n * m)$.

Resultados:

- El tiempo de ejecución registrado fue de 0.001 ms para movies-small.csv.

Análisis:

- Los tiempos de ejecución son aceptables, pero la complejidad $O(n * m)$ implica que el rendimiento puede verse afectado si hay muchas compañías productoras por película. Sin embargo, para conjuntos de datos pequeños, el sistema parece funcionar bien.

Requerimiento 8: Consultar películas de un género desde un año específico

Complejidad Teórica:

- La complejidad es $O(n)$ para la búsqueda y $O(m)$ para la verificación de condiciones, con un total de $O(n * m)$.

Resultados:

- El tiempo de ejecución para movies-small.csv fue de 0.001 ms.

Análisis:

- Al igual que en el requerimiento 7, el rendimiento es eficiente para conjuntos de datos pequeños. La complejidad podría impactar en el rendimiento si se utilizan conjuntos de datos más grandes, pero la implementación actual es efectiva.

Conclusión General

La implementación general de los requerimientos ha demostrado ser eficiente y se alinea con las expectativas establecidas por la complejidad temporal teórica. A medida que se incrementa el tamaño del conjunto de datos, los tiempos de ejecución aumentan, pero se mantienen dentro de límites razonables para la mayoría de los casos de uso.

La función de carga de datos inicial también parece manejar adecuadamente los archivos CSV, aunque se deben tener en cuenta posibles problemas de formato en el futuro.