INSTITUTO POLITÉCNICO NACIONAL

Unidad Profesional Interdisciplinaria de Ingeniería
Campus Zacatecas.

Materia:

Análisis y diseño de algoritmos

Practica 01: Analisis de casos

Docente:

Erika Sanchez Femat

Nombre del alumno:

Dalia Naomi García Macías

Fecha de entrega:

18 de septiembre de 2023

Índice

1. Introduction

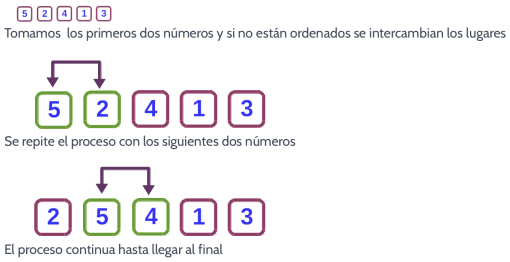
Durante la práctica trabajaremos con el algoritmo burbuja y el algoritmo de burbuja mejorada, como bien ya sabemos, el algoritmo de burbuja, es un método de ordenamiento simple que se usa para ordenar una lista o arreglo de elementos, recibe el nombre de burbuja dado que es la descripción de como los elementos más grandes dentro de la lista van "burbujeando" hacia la parte superior de la lista o arreglo.
La idea básica detrás del algoritmo de burbuja es comparar pares de elementos adyacentes en la lista y, si están en el orden incorrecto, intercambiarlos. Este proceso se repite hasta que no se realicen intercambios en un pase completo a través de la lista, lo que indica que la lista está ordenada. Sin embargo, a pesar de funcionar a la perfección, este algoritmo es muy ineficiente, dado que la forma en la que esta estructurado hace que la computadora repita demasiadas veces los ciclos dentro del algoritmos, siendo esto en la mayoria de los casos inecesario.Por lo que al darse cuenta de esto, los programadores crearon un nuevo algoritmo llamado burbuja mejorada, algoritmo parecido al anterior a excepción de una optimización que les evitaría hacer operaciones inecesarias ocasionando que el tiempo de ejecucion fuera mayor.
En el algoritmo mejorado, cada que se recorre el arreglo va diciendo donde y cada cuando ya se realizaron intercambios de elementos, para asi evitar hacerlos de nuevo si no es necesario.

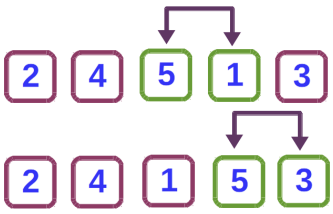
2. Desarrollo de la practica

2.1. Implementación del algoritmo

Algoritmo Burbuja
Burbuja(Bubble Sort en inglés) es un sencillo algoritmo de ordenamiento. Funciona revisando cada elemento de la lista que va a ser ordenada con el siguiente, intercambiándolos de posición si están en el orden equivocado.
Es necesario revisar varias veces toda la lista hasta que no se necesiten más intercambios, lo cual significa que la lista está ordenada. Este algoritmo obtiene su nombre de la forma con la que suben por la lista los elementos durante los intercambios, como si fueran pequeñas burbujas.
También es conocido como el método del intercambio directo. Dado que solo usa comparaciones para operar elementos, se lo considera un algoritmo de comparación, siendo el más sencillo de implementar.

A continuacion se muestra un ejemplo:





El último número ya queda ordenado por lo que en la siguiente iteración ya no se evalúa acortando el proceso

Al finalizar el algoritmo tenemos como resultado la lista ordenado

Algoritmo Burbuja Optimizada

Implementa una mejora al método de ordenamiento burbuja. Este método recorre todo el arreglo comparando cada uno de los elementos con el elemento siguiente e intercambiándolo de ser necesario. Al finalizar la iteración el elemento mayor queda ubicado en la última posición, mientras los elementos menores ascienden una posición.

La mejora es que, como al final de cada iteración el elemento mayor queda situado en su posición, ya no es necesario volverlo a comparar con ningún otro número, reduciendo así el número de comparaciones por iteración.

A continuacion se muestra un ejemplo:

Código

Primero que nada antes de empezar a desarrollar el código se debían tomar en cuenta las especificaciones que incluiría, las cuales decían que nuestro código debia contener un menú para que asi el usuario pudiera tener la facilidad de escoger que método de ordenamiento quería usar, del mismo modo con el tamaño y los elementos que contendrían el arreglo.

Teniendo en cuenta esto, tomé la decisión de usar funciones, ya que así podría incluir ambos métodos de manera más sencilla dentro del menú. Creé cuatro funciones: primero, una que recopilara los datos del arreglo dados por el usuario y a su vez me imprimiera el arreglo sin ordenar; luego, la función "burbuja" que tendría el funcionamiento del primer método (el método de la burbuja); después, otra que contuviera el desarrollo del segundo método (el método de la burbuja mejorada); y finalmente, una que me mostrara cómo quedaría el arreglo ya ordenado.

2.2. Análisis de casos

Algoritmo burbuja y burbuja optimizada

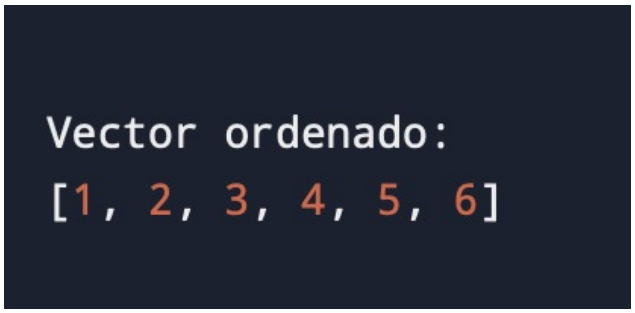
En estos algoritmos, el mejor, peor y caso promedio sera el mismo, dado que el procedimiento es el mismo en ambos algoritmos aunque la burbuja optimizada tenga ciertas mejorias. Las mejorias de esta solo modifican el como recorre el ciclo, lo que hace es ahorrarle vueltas y comparaciones dentro del arreglo, por lo que sigue siendo el procedimiento del primer codigo solo un poco mejor.

- Mejor caso:
  El mejor caso para estos algoritmos seria que solo tuviera que hacer un intercambio de números, y que este fuera el de las dos ultimas posiciones del arreglo, dado que así solo tendría que hacer eso y la lista ya estaría ordenada.
  Un ejemplo seria que teniendo un arreglo con seis datos solo tuviera que intercambiar los elementos de las ultimas dos posiciones.

```
Vector original:
[1, 2, 3, 4, 6, 5]

[1 > 2, 3, 4, 6, 5]
[1, 2 > 3, 4, 6, 5]
[1, 2, 3 > 4, 6, 5]
[1, 2, 3, 4 > 6, 5]
[1, 2, 3, 4, 6 > 5]

Vector ordenado:
[1, 2, 3, 4, 5, 6]
```

- Caso promedio:
  El caso promedio del algoritmo burbuja seria que solo tuviera que cambiar de lugar la mitad de los números, por ejemplo teniendo un arreglo de 6 elementos, nuestro caso promedio seria que tuviera que intercambiar los elementos de las ultimas tres posiciones nadas mas.

```
Vector original:
[1, 3, 6, 4, 5, 2]

[1 > 3, 6, 4, 5, 2]
[1, 3 > 6, 4, 5, 2]
[1, 3, 6 > 4, 5, 2]
[1, 3, 4, 6 > 5, 2]
[1, 3, 4, 5, 6, 2]
[1, 3, 4, 5, 6 > 2]
[1, 3, 4, 5, 2, 6]
[1 > 3, 4, 5, 2, 6]
[1, 3 > 4, 5, 2, 6]
[1, 3, 4 > 5, 2, 6]
[1, 3, 4, 5 > 2, 6]
[1, 3, 4, 2, 5, 6]
[1, 3, 4, 2, 5 > 6]
[1 > 3, 4, 2, 5, 6]
[1, 3 > 4, 2, 5, 6]
[1, 3, 4 > 2, 5, 6]
[1, 3, 2, 4, 5, 6]
[1, 3, 2, 4 > 5, 6]
[1, 3, 2, 4, 5 > 6]
[1 > 3, 2, 4, 5, 6]
[1, 3 > 2, 4, 5, 6]
[1, 2, 3, 4, 5, 6]
```



- Peor caso:
  En cuanto al peor caso de este algoritmo, seria que el código tuviese que recorrer todo el arreglo varias veces intercambiando todos los elementos de todas las posicicones del arreglo, que ningun elemento estuviese en la posicion que le corresponde.

2.3. Complejidad

Ambos algoritmos cuentan con una complejidad $O(n^2)$, ya que en el peor de los casos al algoritmo debera recorrer muchsimas veces el arreglo y hacer muchisimos intercambios de elementos. Esto nos indica qu eentre mayor sean las entradas del codigo se haran mas comparaciones, todo esto aunque el resultado del odigo siempre sea el mismo arreglo ordenado. Esto mismo sucedera on el algoritmo de burbuja optimizada, ya que aunque hace menos cambios que el algoritmo burbuja normal, igual entre mas entradas reciva mas veces ejecutara el ciclo y su resultado seguira siendo el mismo.

2.4. Comparación de resultados

Ya con ambos algoritmos hechos, use la ibreria de tiempo para poder ver el tiempo que tarda en ejecutarse cada uno de los algoritmos, como ya me imaginada, el algoritmo de burbuja optimizada tarda menos tiempo en ejecutarse, aunque no es mucha la diferencia entre ambos tiempos, el de burbuja optimizada sigue siendo menos y claro el tiempo tambien es distinto segun los casos, en ambos algoritmos para el mejor caso tardo poco en comparacion con el del peor caso.

3. Conclusiones

Como ya se menciono anteriormente en el documento, el metodo burbuja es un algoritmo de ordenamiento de datos, que aunque funciona a la aperfeccion, se vulve deficiente a medida que crece el numero de las entradas, ya que entre mayor sea el arreglo, mas elementos tendra que recoorrer y comparar, lo que hara que funcione lento a diferencia de otros metodos de ordenamiennto. Aun asi a difernciа de otros es facil comprender su funcionamiento y la forma de programarlo, ya que usa una estructura simple y sencilla.

4. Referencias

- `https://juncotic.com/ordenamiento-de-burbuja-algoritmos-de-ordenamiento/`
- `https://runestone.academy/ns/books/published/pythoned/SortSearch/ElOrdenamientoBurbuja.html`
- `https://tutospoo.jimdofree.com/tutoriales-java/m%C3%A9todos-de-ordenaci%C3%B3n/burbuja-optimizad`