



INSTITUTO POLITÉCNICO NACIONAL
Unidad Profesional Interdisciplinaria de Ingeniería
Campus Zacatecas.

MATERIA:

- **Ánalysis y Diseño de Algoritmos**

Practica 01:

- **Analisis de Casos**

DOCENTE:

- **Erika Sánchez Femat**

Alumno:

- **Gael García Torres**

Introducción

Los métodos de burbuja, incluyendo el método de burbuja estándar y la versión optimizada, son dos algoritmos de ordenación simples pero menos eficientes utilizados principalmente con fines educativos y como punto de partida para comprender los conceptos básicos de la ordenación.

Método de Ordenamiento de Burbuja:

Este método consiste en acomodar el vector moviendo el mayor hasta la última casilla, comenzando desde la casilla cero del vector hasta aver acomodado el número más grande en la última posición, una vez acomodado el más grande, prosigue a encontrar y acomodar el siguiente más grande comparando de nuevo los números desde el inicio del vector y así sigue hasta ordenar todos los elementos de todo el arreglo.

Método de Ordenamiento de 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.

Análisis de Casos

Método de Ordenamiento de Burbuja :

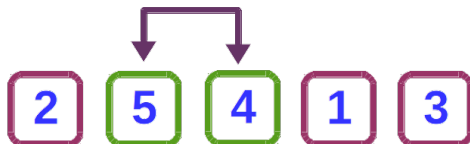
Comenzamos con una lista de elementos no ordenados



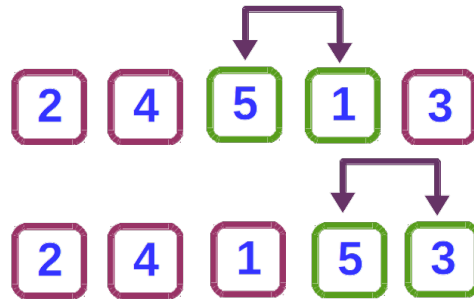
Tomamos los primeros dos números y si no están ordenados se intercambian los lugares



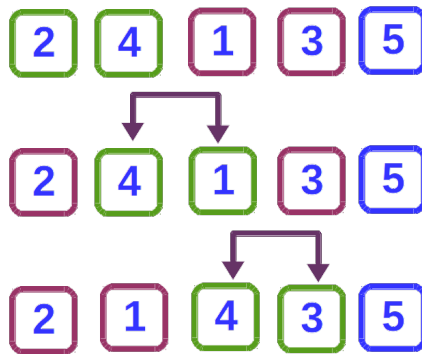
Se repite el proceso con los siguientes dos números



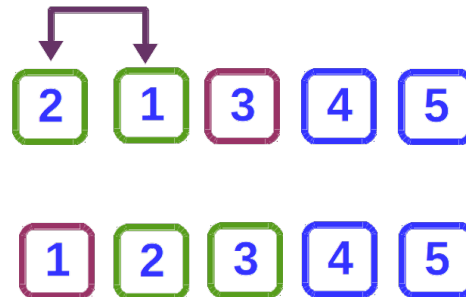
El proceso continua hasta llegar al final



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



En la tercera iteración no se evalúan los últimos dos valores



La cuarta iteración se finaliza sin que se haya realizado un intercambio por lo que el algoritmo termina

1 2 3 4 5

Metodo de Ordenamiento de Burbuja Optimizada

Paso 1:

Vector original

45	17	23	67	21
----	----	----	----	----

Iteración 1:

45	17	23	67	21	Se genera cambio
17	45	23	67	21	Se genera cambio
17	23	45	67	21	No hay cambio
17	23	45	67	21	Se genera cambio
17	23	45	21	67	Fin primera iteración

Paso 2:

Iteración 2:

17	23	45	21	67	No hay cambio
17	23	45	21	67	No hay cambio
17	23	45	21	67	Se genera cambio
17	23	21	45	67	Fin segunda iteración

Paso 3:

Iteración 3:

17	23	21	45	67	No hay cambio
17	23	21	45	67	Se genera cambio
17	21	23	45	67	Fin tercera iteración

Paso 4:

Iteración 4:

17	21	23	45	67	No hay cambio
17	21	23	45	67	Fin cuarta iteración

Metodo de Ordenamiento de Burbuja

Mejor Caso: El mejor caso ocurre cuando la lista ya está ordenada. En este caso, el algoritmo realiza solo una pasada a través de la lista para verificar que no se hagan intercambios. La complejidad en tiempo es $O(n)$, donde "n" es el número de elementos en la lista.

Ejemplo de Mejor Caso:

Lista inicial: [1,2,3,4,5]

Peor Caso: El peor caso ocurre cuando la lista está ordenada en orden inverso, y se deben realizar intercambios en cada paso de pasada. La complejidad en tiempo es $O(n^2)$.

Ejemplo de Peor Caso:

Lista inicial: [5,4,3,2,1]

Caso Promedio: El caso promedio suele ser $O(n^2)$ y puede ocurrir en listas desordenadas aleatoriamente.

Ejemplo de Caso Promedio::

Lista inicial: [3,1,5,4,2]

Metodo de Ordenamiento de Burbuja Optimizada

Mejor Caso: El mejor caso ocurre cuando la lista ya está completamente ordenada. El algoritmo detecta esto en la primera pasada hacia adelante y no realiza ningún intercambio adicional. La complejidad en tiempo es $O(n)$, donde "n" es el número de elementos en la lista.

Ejemplo de Mejor Caso:

Lista inicial: [1,2,3,4,5]

Peor Caso: El peor caso ocurre cuando la lista está ordenada en orden inverso, y se deben realizar intercambios en cada pasada hacia adelante y hacia atrás. La complejidad en tiempo sigue siendo $O(n^2)$.

Ejemplo de Peor Caso:

Lista inicial: [5,4,3,2,1]

Caso Promedio: El caso promedio suele ser $O(n^2)$ y puede ocurrir en listas desordenadas aleatoriamente. La optimización ayuda a detener el proceso antes en comparación con el método de burbuja estándar, pero la complejidad en tiempo generalmente sigue siendo cuadrática.

Ejemplo de Caso Promedio:

Lista inicial: [3,1,5,4,2]

El ordenamiento de burbuja optimizado puede ser más eficiente en el mejor caso en comparación con el método de burbuja estándar, pero sigue siendo ineficiente en el peor caso y no se utiliza comúnmente en aplicaciones prácticas para ordenar listas grandes debido a su complejidad cuadrática.

Complejidad:

Mejor caso:

El algoritmo de ordenamiento burbuja tiene una complejidad en el mejor caso de $O(n)$.

Peor caso:

La complejidad en el peor caso del algoritmo de ordenamiento de burbuja es $O(n^2)$, donde "n" es el número de elementos en la lista que se está ordenando. Esto significa que el tiempo de ejecución del algoritmo aumenta cuadráticamente con el número de elementos en la lista.

Promedio Caso:

La complejidad en tiempo en el caso promedio también es $O(n^2)$. Esto se debe a que el algoritmo de burbuja realiza intercambios de elementos adyacentes repetidamente hasta que la lista esté ordenada, y en el peor caso, esto implica comparaciones e intercambios en cada par de elementos.

Conclusión:

Al comparar los tiempos de ejecución entre el método de burbuja y el método de burbuja optimizada en sus mejores casos, observamos que el método de burbuja optimizada demuestra un rendimiento superior. Concretamente, en el mejor caso, el método de burbuja toma un tiempo de ejecución de 10.91369, mientras que el método de burbuja optimizada logra completar la operación en 7.64151 unidades de tiempo. Estos resultados indican claramente que el método de burbuja optimizada es más eficiente en términos de tiempo de ejecución en condiciones ideales. Este menor tiempo requerido para ordenar una lista implica una utilización más eficiente de los recursos computacionales disponibles.

Referencias:

http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro9/metodo_de_burbuja.html

<https://tutospoo.jimdofree.com/tutoriales-java/m%C3%A9todos-de-ordenaci%C3%B3n/burbuja-optimizado/#:~:text=Implementa%20una%20mejora%20al%20m%C3%A9todo,e%20intercambi%C3%A1ndolo%20de%20ser%20necesario.>

https://issuu.com/ernestoalonsopestanajimenez/docs/metodo_burbuja_optimizado.pptx#:~:text=Conclusiones%20El%20m%C3%A9todo%20Burbuja%20Optimizado,utilizando%20menos%20recursos%20del%20computador.

<https://tutospoo.jimdofree.com/tutoriales-java/m%C3%A9todos-de-ordenaci%C3%B3n/burbuja-optimizado/>

<https://patsoptimizado.wordpress.com/>