

# Proyecto 1: Ordenando Números

Ricardo Castro Jiménez  
Carmen Hidalgo Paz  
Jorge Guevara Chavarría

*Escuela de Ingeniería en Computación, Tecnológico de Costa Rica, San José, Costa Rica*

riccastro@estudiantec.cr  
carmenhidalgopaz@estudiantec.cr  
joguevara@estudiantec.cr

Se hizo un programa en C que ordena datos usando diferentes algoritmos y los muestra gráficamente con líneas dentro de un círculo. El usuario puede escoger cuántos datos quiere ver, cómo deben estar al inicio (ordenados o no) y cuál algoritmo usar. Al darle al botón de ordenar, se empieza a mostrar cómo se van moviendo los datos paso a paso.

En la parte de la interfaz gráfica se aprendió a utilizar los botones de radio para que el usuario pudiera escoger el orden de los datos iniciales [1]. El barajar los números de un vector se puede realizar de varias maneras, pero en este programa se utilizó el algoritmo de Fisher-Yates (también conocido como el algoritmo de Knuth) [2]. Además se utilizó un combo box para el despliegue del menú de algoritmos de ordenamiento [3]. Asimismo, se aprendieron a utilizar delays para la visualización de los pasos de cada algoritmo de ordenamiento.

El BubbleSort fue el primero que se agregó, un algoritmo simple que compara elementos adyacentes y los intercambia si están fuera de orden. Aunque es uno de los métodos más básicos, su funcionamiento es fácil de entender y visualizar [4].

El Cocktail Sort es una variante del anterior, pero con la diferencia de que recorre el arreglo en ambas direcciones, primero de izquierda a derecha y luego de derecha a izquierda, lo que lo hace un poco más eficiente en ciertos casos [5].

Luego se implementó el Quicksort, que se basa en dividir y conquistar. Selecciona un pivote y reordena el arreglo de modo que los mayores queden a un lado y los menores al otro. Se repite el proceso recursivamente. Es uno de los algoritmos más rápidos en la práctica y se adaptó para que funcionara en orden descendente [6].

El ShellSort mejora el método de inserción usando intervalos o saltos. Es útil para arreglos más grandes y hace menos movimientos que el Insertion Sort tradicional. Se usó la secuencia clásica de divisiones entre dos para los saltos [7].

Además, se agregó el Gnome Sort, que es un algoritmo poco convencional pero muy visual. Su lógica es simple: si dos elementos están desordenados, se intercambian y se retrocede una posición, como si se deshiciera el avance hasta que todo esté en orden. Aunque no es eficiente, se escogió por su comportamiento llamativo para efectos visuales [8].

Después se incorporó el Exchange Sort, un algoritmo que compara todos los pares posibles de elementos y los intercambia si están en el orden incorrecto. Aunque no es eficiente para listas grandes, su lógica directa lo hace útil para fines educativos y para ilustrar el funcionamiento de intercambios múltiples dentro de un arreglo [9].

También se agregó el Selection Sort, conocido por su simplicidad. Este algoritmo selecciona iterativamente el valor más pequeño del arreglo no ordenado y lo coloca en su posición final. No requiere intercambios innecesarios y mantiene un comportamiento predecible, aunque no mejora mucho en el mejor de los casos [10].

El Insertion Sort fue implementado por su eficacia en arreglos pequeños o parcialmente ordenados. Su lógica consiste en tomar cada elemento y colocarlo en la posición correcta dentro de la parte ya ordenada, como si se fueran insertando cartas en una mano. Es rápido en casos favorables y fácil de visualizar [11].

Asimismo, se añadió el Merge Sort, un algoritmo que divide recursivamente el arreglo en mitades, las ordena y luego las fusiona. Gracias a su estabilidad y eficiencia incluso en casos desfavorables, se utiliza comúnmente en situaciones donde se requiere rendimiento garantizado. Se implementó respetando el orden ascendente y aprovechando su naturaleza recursiva [12].

Finalmente, se integró el Pancake Sort, un algoritmo poco convencional que ordena mediante flips, como si se tratara de apilar y dar la vuelta a panqueques. Aunque su rendimiento no es competitivo frente a otros métodos modernos, fue escogido por su carácter visual y la forma en que ordena los datos usando solo flips [13].

## FUENTES CONSULTADAS:

- [1] K. O’Kane, “Linux Gtk Glade Programming Part 1”, *YouTube*, 22 de mar. del 2019.  
[video en línea]. Disponible en:  
[https://www.youtube.com/watch?v=g-KDOH\\_uqPk&list=PLmMgHNtOIstZEvqYJncYUx52n8\\_OV0uWy&index=2](https://www.youtube.com/watch?v=g-KDOH_uqPk&list=PLmMgHNtOIstZEvqYJncYUx52n8_OV0uWy&index=2) [Accesado: 8 de mar. del 2025]
- [2] Inside code, “How to shuffle an array (Fisher-Yates algorithm) - Inside code”, *YouTube*, 18 de oct. del 2020. [video en línea]. Disponible en:  
<https://www.youtube.com/watch?v=4zx5bM2OcvA&t=57s> [Accesado: 25 de feb. del 2025]
- [3] K. O’Kane, “Linux Gtk Glade Programming Part 3 - Adding Radio Buttons”, *YouTube*, 24 de mar. del 2019. [video en línea]. Disponible en:  
<https://www.youtube.com/watch?v=FGGB8wKE6MY> [Accesado: 4 de abr. del 2025]
- [4] GeeksforGeeks, “Bubble Sort”, [En línea]. Disponible en:  
<https://www.geeksforgeeks.org/bubble-sort/> [Accesado: 6 de abril del 2025].
- [5] GeeksforGeeks, “Cocktail Sort”, [En línea]. Disponible en:  
<https://www.geeksforgeeks.org/cocktail-sort/> [Accesado: 6 de abril del 2025].
- [6] GeeksforGeeks, “QuickSort – Data Structure and Algorithm Tutorials”, [En línea].  
Disponible en: <https://www.geeksforgeeks.org/quick-sort/> [Accesado: 6 de abril del 2025].
- [7] GeeksforGeeks, “ShellSort – Data Structure and Algorithm Tutorials”, [En línea].  
Disponible en: <https://www.geeksforgeeks.org/shellsort/> [Accesado: 6 de abril del 2025].
- [8] GeeksforGeeks, “Gnome Sort – Data Structure and Algorithm Tutorials”, [En línea].  
Disponible en: <https://www.geeksforgeeks.org/gnome-sort-a-stupid-one/> [Accesado: 6 de abril del 2025].
- [9] GeeksforGeeks, “Introduction to Exchange Sort Algorithm”, [En línea]. Disponible en:  
<https://www.geeksforgeeks.org/introduction-to-exchange-sort-algorithm/>
- [10] GeeksforGeeks, “Selection Sort”, [En línea]. Disponible en:  
[https://www.geeksforgeeks.org/selection-sort-algorithm-2/?ref=header\\_outind](https://www.geeksforgeeks.org/selection-sort-algorithm-2/?ref=header_outind)
- [11] GeeksforGeeks, “Insertion Sort Algorithm”, [En línea]. Disponible en:  
[https://www.geeksforgeeks.org/insertion-sort-algorithm/?ref=header\\_outind](https://www.geeksforgeeks.org/insertion-sort-algorithm/?ref=header_outind)
- [12] GeeksforGeeks, “Merge Sort – Data Structure and Algorithms Tutorials”, [En línea].  
Disponible en: [https://www.geeksforgeeks.org/merge-sort/?ref=header\\_outind](https://www.geeksforgeeks.org/merge-sort/?ref=header_outind)
- [13] GeeksforGeeks, “A Pancake Sorting Problem”, [En línea]. Disponible en:  
[https://www.geeksforgeeks.org/a-pancake-sorting-question/?ref=header\\_outind](https://www.geeksforgeeks.org/a-pancake-sorting-question/?ref=header_outind)