



**Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos  
Fundamentales (Evidencia Competencia)**

Reflexión Individual

Ayleen Osnaya Ortega - A01426008

## Índice

<b>1 Ordenamiento.....</b>	<b>1</b>
1.1 Importancia:.....	2
1.2 Eficiencia:.....	2
<b>2 Búsqueda.....</b>	<b>2</b>
2.1 Importancia:.....	3
2.2 Eficiencia:.....	3
<b>3 Reflexión personal.....</b>	<b>3</b>
<b>4 Referencias:.....</b>	<b>4</b>

Los algoritmos de **ordenamiento** y **búsqueda** son fundamentales en la informática y en la resolución de problemas reales, ya que permiten manipular grandes cantidades de información de manera más rápida y eficiente. Ya que usamos ordenamiento para acomodar nuestros datos y así hacer que la búsqueda de estos sea más sencilla.

## 1 Ordenamiento

En la programación, a menudo trabajamos con listas o conjuntos de datos que necesitamos organizar o acomodar de manera específica. Para lograrlo, se usan los algoritmos de ordenamiento. Los cuales son métodos que permiten colocar los elementos de alguna forma determinada, ya sea de menor a mayor o de mayor a menor. Estos algoritmos son importantes porque facilitan tareas como la búsqueda de información y el manejo de datos, su importancia recae en que facilita, agiliza y hace esta estructura ordenada de manera eficiente.

### 1.1 Importancia:

- Facilita la búsqueda de datos, ya que una colección ordenada permite aplicar métodos más rápidos como la búsqueda binaria.
- Mejora la legibilidad y análisis de la información en bases de datos, reportes y sistemas en tiempo real.

- Es un paso previo en muchos otros algoritmos, por ejemplo, en la detección de duplicados, en compresión de datos o en la optimización de recursos.

## 1.2 Eficiencia:

- Algoritmos como **Bubble Sort** o **Insertion Sort** son fáciles de implementar, pero poco eficientes para grandes volúmenes de datos ( $O(n^2)$ ).
- Algoritmos como **Merge Sort** o **Quick Sort** ofrecen mayor eficiencia promedio ( $O(n \log n)$ ) y son ampliamente usados en la práctica.

## 2 Búsqueda

En la programación, buscar información significa localizar y recuperar datos específicos usando diferentes herramientas y técnicas digitales. En la actualidad, esta habilidad es fundamental, ya que nos permite resolver problemas, aprender cosas nuevas y mantenernos al día en un mundo cada vez más digital. Saber encontrar información relevante de manera rápida y eficiente es clave tanto en el ámbito académico como en el profesional.

### 2.1 Importancia:

- Permite encontrar información específica en colecciones grandes (archivos, bases de datos, redes sociales, etc.).
- Es esencial en motores de búsqueda, sistemas de recomendación, comercio electrónico y en software científico.

### 2.2 Eficiencia:

- **Búsqueda secuencial:** útil en listas pequeñas o desordenadas, pero con complejidad  $O(n)$ .

- **Búsqueda binaria:** requiere datos ordenados, pero permite encontrar elementos en  $O(\log n)$ , siendo mucho más eficiente en grandes colecciones.
- 

### 3 Reflexión personal

En la vida real, no basta con que un algoritmo “funcione”; debe ser eficiente según el tamaño del problema y los recursos disponibles. Por ejemplo, en una aplicación pequeña con pocos datos, un algoritmo de ordenamiento básico puede ser suficiente. Sin embargo, en una base de datos con millones de registros, elegir el algoritmo adecuado puede significar la diferencia entre un proceso que tarda segundos o minutos, y uno que tarda horas. Justamente este fue parte de nuestro reto como equipo, deducir cuál sería el mejor tipo de ordenamiento según nuestro problema o en este caso bitácora.

Además, la elección del algoritmo no depende solo de la eficiencia teórica ( $O(n)$ ,  $O(\log n)$ ), sino también del contexto práctico: la memoria disponible, si los datos ya están parcialmente ordenados, si las búsquedas son muy frecuentes, o si se necesita ordenar en paralelo.

En conclusión, comprender y aplicar correctamente los algoritmos de ordenamiento y búsqueda es esencial porque permite optimizar el tiempo de ejecución y el uso de recursos, lo que a su vez impacta directamente en la experiencia del usuario, y la competitividad de una empresa. Es decir que un código eficiente y rápido tiene mayor probabilidad de ser preferido o querido, ya que eso hace que el trabajo de la persona se vuelva a su vez más rápido. En nuestro caso, elegimos quick sort, ya que es el sort más implementado por su rapidez y eficiencia, y el que usas la mayoría de programas como default.

### 4 Referencias:

BlackeyeB. (2023, 1 mayo). *Algoritmos de ordenación explicados con ejemplos en JavaScript, Python, Java y C++*. freeCodeCamp.org.

<https://www.freecodecamp.org/espanol/news/algoritmos-de-ordenacion-explicados-con-ejemplos-en-javascript-python-java-y-c/>

*TecnoDigital. (2024, 3 noviembre). Los 10 algoritmos de ordenamiento más populares. Informática y Tecnología Digital.*

<https://informatecdigital.com/los-10-algoritmos-de-ordenamiento-mas-populares/>

Servernet. (2023, 22 enero). *La búsqueda de información en informática: herramientas y objetivos* - ServerNet. ServerNet.

<https://servernet.com.ar/busqueda-de-informacion-en-informatica/>