

Actividad 1: Investigación Teórica - Algoritmos de Búsqueda
Realizado por: Daniella Rocha

1. Definición de Algoritmo de Búsqueda

Un algoritmo de búsqueda es una serie de instrucciones lógicas diseñadas para localizar un elemento específico (conocido como "clave" o "target") dentro de una estructura de datos. Su objetivo principal es determinar si el elemento existe y, en caso afirmativo, retornar su posición o el objeto mismo. Son fundamentales para la recuperación eficiente de información en sistemas computacionales.

2. Diferencias entre Tipos de Búsqueda

- **Búsqueda Lineal (Secuencial):**
 - *Funcionamiento:* Recorre la estructura de datos elemento por elemento, desde el principio hasta el final, comparando cada uno con el valor buscado.
 - *Requisito:* No requiere que los datos estén ordenados.
 - *Eficiencia:* Baja en grandes volúmenes de datos, ya que en el peor de los casos debe revisar toda la lista.
- **Búsqueda Binaria:**
 - *Funcionamiento:* Utiliza la estrategia de "divide y vencerás". Selecciona el elemento central; si es mayor o menor que el buscado, descarta la mitad correspondiente y repite el proceso con la mitad restante.
 - *Requisito:* **Es obligatorio** que la lista esté ordenada previamente.
 - *Eficiencia:* Alta. Reduce drásticamente el número de comparaciones necesarias (logarítmica).
- **Búsqueda en Estructuras No Lineales:**
 - *Funcionamiento:* Se aplica en grafos o árboles donde los datos no siguen una secuencia simple. Utiliza métodos como Búsqueda en Profundidad (DFS) o Búsqueda en Anchura (BFS) para navegar entre nodos conectados.
 - *Uso:* Ideal para mapas, redes o directorios de archivos.

3. Casos de Uso Reales

- **Listas de asistencia:** Uso de búsqueda lineal para encontrar un nombre en una lista pequeña no ordenada.
- **Directorios telefónicos digitales:** Uso de búsqueda binaria para localizar rápidamente un contacto por nombre.

- **Sistemas de navegación GPS:** Uso de búsqueda en grafos (algoritmos como Dijkstra) para calcular la ruta más corta entre dos ubicaciones en un mapa.

4. Ejemplos en el Mundo Profesional

- **Motores de Búsqueda (Google/Bing):** Utilizan indexación compleja y algoritmos de grafos (como PageRank) para clasificar y recuperar millones de páginas web en milisegundos.
- **Bases de Datos (SQL/Oracle):** Emplean estructuras de árbol (B-Trees) para realizar búsquedas binarias eficientes sobre millones de registros sin tener que leer toda la tabla.
- **Inteligencia Artificial:** En juegos como el ajedrez, los algoritmos buscan en árboles de decisiones para predecir y seleccionar la mejor jugada futura.

5. Ventajas y Desventajas

- **Búsqueda Lineal:**
 - *Ventaja:* Extremadamente fácil de implementar; no requiere ordenar datos.
 - *Desventaja:* Muy lenta e ineficiente con grandes cantidades de información.
- **Búsqueda Binaria:**
 - *Ventaja:* Muy rápida y eficiente para grandes volúmenes de datos.
 - *Desventaja:* Requiere mantener los datos ordenados constantemente, lo cual tiene un costo computacional.

Referencias Bibliográficas (APA 7ma Edición)

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3a ed.). MIT Press.
- Joyanes Aguilar, L. (2008). *Fundamentos de programación: Algoritmos, estructura de datos y objetos* (4a ed.). McGraw-Hill.