



UNIVERSIDAD AUTÓNOMA DE CAMPECHE

Facultad de Ingeniería



INGENIERÍA EN SISTEMAS COMPUTACIONALES

Estructura de Datos II

Prof. García Reyes José Ángel

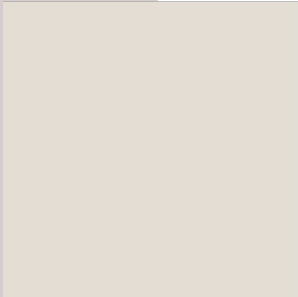
MÉTODO DE BÚSQUEDA SECUENCIAL Y SECUENCIAL INDEXADA

Integrantes:

Negrón Salazar Danna G.
Patiño Tun Hugo A.
Rios Fuentes Sebastian E.

¿Qué es?


BÚSQUEDA SECUENCIAL



La *búsqueda secuencial* es un algoritmo simple que se utiliza para encontrar un elemento específico en una lista o arreglo. Se revisa cada elemento de manera secuencial, desde el principio hasta el final, hasta que se encuentra el elemento o se llega al final de la lista.

ALGORITMO

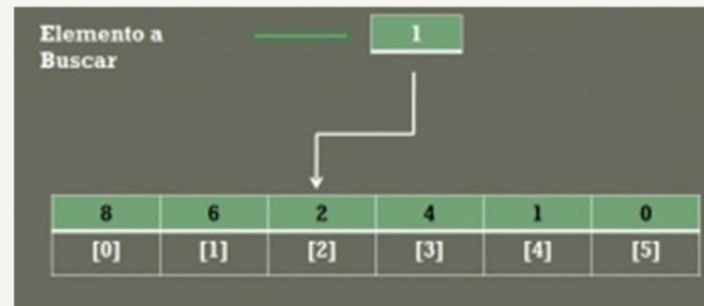
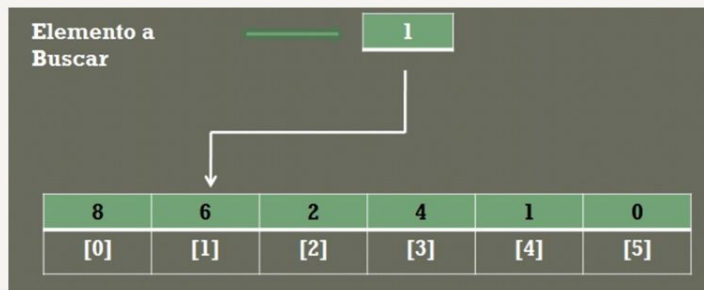
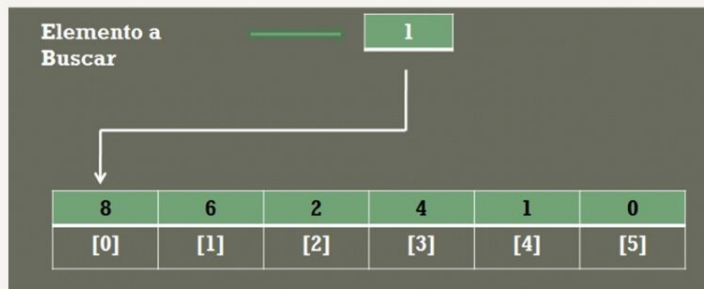
El algoritmo de búsqueda secuencial sigue un procedimiento muy simple.

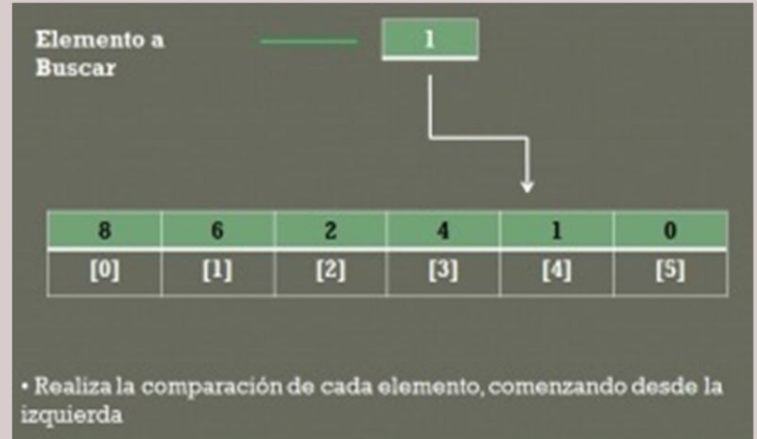
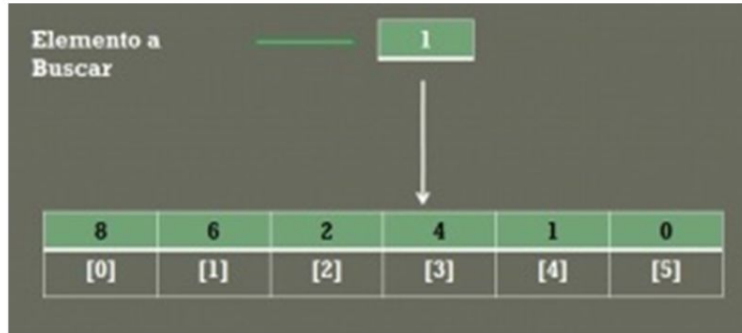


Consiste en recorrer una lista desde el primer elemento hasta el último, comparando cada uno con el valor buscado.

Si encuentra el valor, termina y devuelve la posición o un indicador de éxito; si no, sigue buscando hasta el final de la lista

Ejemplificar





Elemento a buscar

- **Mejor caso:** $O(1)$, si el valor está en la primera posición.
- **Peor caso:** $O(n)$, si el valor está en la última posición o no está en la lista.
- **Promedio:** $O(n/2)$, ya que en promedio encontrará el elemento a la mitad de la lista.

COMPLEJIDAD



Aplicaciones

- 1 Búsqueda en Listas Pequeñas
- 2 Archivos de Texto
- 3 Bases de Datos Pequeñas
- 4 Búsqueda en Dispositivos de
Entrada

¿Cuándo debemos utilizarla?

Desventajas


- Ineficiencia para grandes listas.
- No explota propiedades de los datos.

Ventajas

- Es fácil de implementar y comprender.
- No requiere que la lista esté ordenada.

Preguntas

- ¿Qué es búsqueda secuencial?
- ¿Cuáles son sus dos condiciones de finalización?
- Menciona las ventajas
- Menciona las desventajas
- ¿Qué es la búsqueda secuencial indexada y cómo funciona en comparación con la búsqueda secuencial simple?



Secuencial indexada

Surge para optimizar la búsqueda en grandes conjuntos de datos.



Primera implementación

Sistemas de almacenamiento fuera de memoria principal (archivos en discos).



Usos

En bases de datos y sistemas de gestión de archivos para balancear la velocidad de acceso y simplicidad

¿Qué es?

Es una técnica de búsqueda que utiliza un índice para dividir datos en secciones más pequeñas en donde:

- Cada entrada del índice apunta al inicio de un bloque
- La búsqueda localiza el bloque adecuado mediante el índice
- Se realiza búsqueda secuencial dentro del bloque correspondiente

Complejidad

- **Peor caso:** $O(n)$ (si el elemento está al final o no existe).
- **Mejor caso:** $O(1)$ (si está en la primera posición del bloque).
- **Construcción del índice:** $O(m)$, donde m es el tamaño del índice.

Comparaciones totales:

1. Búsqueda en el índice: $O(\sqrt{n})$
2. Búsqueda en el bloque: $O(\sqrt{n})$

Complejidad combinada: $O(2\sqrt{n}) \approx O(\sqrt{n})$

Algoritmo

- **Construcción del índice:** Seleccionar algunas posiciones clave del conjunto de datos y almacenarlas en una tabla (array) índice.
- **Búsqueda en el índice:** Identificar el bloque donde podría encontrarse el elemento objetivo.
- **Búsqueda secuencial:** Una vez localizado el bloque, recorrerlo de forma secuencial hasta encontrar el elemento o confirmar que no está presente.

Ejemplo

Imagina que tenemos un arreglo ordenado de 20 elementos y queremos encontrar un valor específico usando búsqueda secuencial indexada.

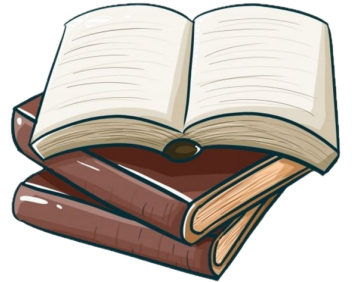
Dividiremos el arreglo en bloques más pequeños y crearemos un índice que nos ayudará a localizar rápidamente el bloque donde podría estar el valor buscado.

[5, 8, 12, 15, 22, 28, 31, 34, 39, 41, 47, 50, 55, 58, 62, 68, 72, 78, 84, 90]

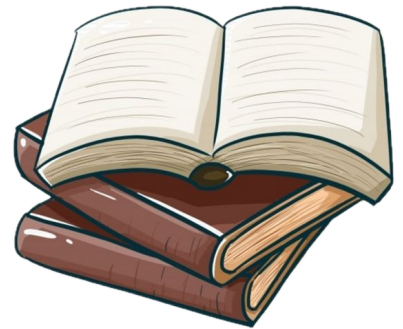
Por si no lo entendiste...

Imagina que trabajas en una biblioteca digital con una lista de 10,000 libros ordenados alfabéticamente por título. Cada título tiene información adicional, como el autor y el género. La lista es demasiado grande para recorrerla completamente cada vez que buscas un libro, por lo que decides implementar un índice para optimizar la búsqueda.

Supongamos que queremos buscar el libro “Moby Dick”.



1. **“1984”** – George Orwell
2. **“A Tale of Two Cities”** – Charles Dickens
3. **“Brave New World”** – Aldous Huxley
4. **“Crime and Punishment”** – Fyodor Dostoevsky
5. **“Don Quijote de la Mancha”** – Miguel de Cervantes
6. **“Moby Dick”** – Herman Melville
7. **“Pride and Prejudice”** – Jane Austen
8. **“The Great Gatsby”** – F. Scott Fitzgerald
9. **“The Odyssey”** – Homero
10. **“War and Peace”** – Leo Tolstoy

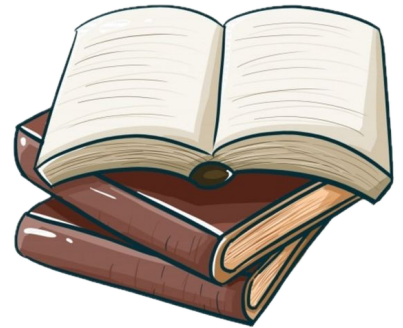


Paso 1: Se define el tamaño de bloque

Se divide la lista en bloques de 3 libros

Paso 2: Se definen los índices

- “1984”
- “Crime and Punishment”
- “Pride and Prejudice”
- “War and Peace”



Paso 3: Se dividen los libros en bloques (de índice a índice)

Bloque 1:

- “1984”
(índice)
- “A Tale of Two Cities”
- “Brave New World”

Bloque 2:

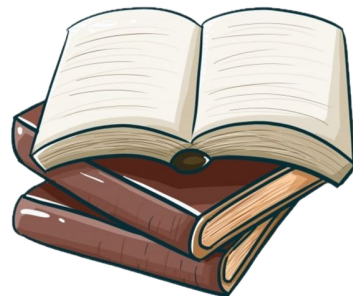
- “Crime and Punishment”
(índice)
- “Don Quijote de la Mancha”
- “Moby Dick”

Bloque 3:

- “Pride and Prejudice”
(índice)
- “The Great Gatsby”
- “The Odyssey”

Bloque 4:

- “War and Peace”
(índice)



Paso 4: Se recorren los bloques de uno en uno hasta hallar el elemento

Bloque 1:

- “1984”
(índice)
- “A Tale of Two Cities”
- “Brave New World”

Bloque 2:

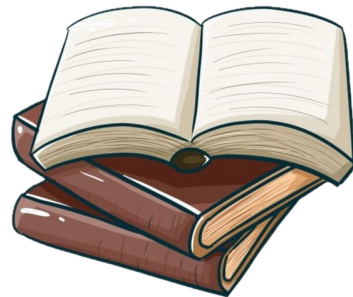
- “Crime and Punishment”
(índice)
- “Don Quijote de la Mancha”
- “Moby Dick”

Bloque 3:

- “Pride and Prejudice”
(índice)
- “The Great Gatsby”
- “The Odyssey”

Bloque 4:

- “War and Peace”
(índice)



Paso 4: Se recorren los bloques de uno en uno hasta hallar el elemento

Bloque 1:

- “1984”
(índice)
- “A Tale of Two Cities”
- “Brave New World”

Bloque 2:

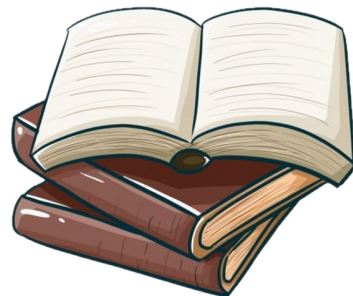
- “Crime and Punishment”
(índice)
- “Don Quijote de la Mancha”
- “Moby Dick”

Bloque 3:

- “Pride and Prejudice”
(índice)
- “The Great Gatsby”
- “The Odyssey”

Bloque 4:

- “War and Peace”
(índice)



Aplicaciones

- **Sistemas de archivos:** Acceso a registros almacenados en archivos.
- **Bases de datos:** Optimización en búsquedas con índices secundarios.
- **Gestión de memoria virtual:** Acceso eficiente a páginas en memoria secundaria.

Ventajas

- Más eficiente que la búsqueda secuencial pura en grandes volúmenes de datos.
- No requiere cargar todos los datos en memoria.
- Fácil de implementar y mantener.

Desventajas

- Costosa en tiempo y espacio para crear el índice.
- Menos eficiente que búsquedas más avanzadas (como la binaria) en datos estrictamente ordenados.
- Un índice mal diseñado pierde efectividad.

Preguntas

- ¿Cuáles son los pasos clave involucrados en el proceso de creación de índices para la búsqueda secuencial indexada?
- ¿Cuáles son las principales ventajas de usar la búsqueda secuencial indexada en conjuntos de datos grandes?
- ¿En qué tipos de aplicaciones o escenarios es más efectivo implementar la búsqueda secuencial indexada en comparación con otros métodos de búsqueda?