

**Instituto Tecnológico de Mérida**

*Campus Poniente*

***“Estructura de datos”***

Alumno: Luis Alberto Salazar Canché

Matricula: **E23080244**

Carrera: Ingeniería en sistemas computacionales

Fecha: 16/03/24

Materia: Estructura de datos

Docente: Ing. Armando López Valadez

***ADA5***

El alumno deberá de:

Explicará en qué consiste cada ejercicio

Por qué usan ese método de búsqueda

Si sé pudiera mejorar con otro método de búsqueda

Conclusiones

***Método Hash***

Texto

Descripción generada automáticamente

***¿Cómo funciona?***

La búsqueda mediante una tabla hash funciona utilizando una función de hash para convertir una clave (como un nombre) en un índice dentro de una estructura de datos, generalmente un arreglo (array) o una lista. La función de hash asigna de manera eficiente la clave a una ubicación específica en la tabla hash, donde se almacena el valor asociado (como una edad). Cuando se quiere buscar un elemento, se aplica nuevamente la función de hash a la clave para determinar su índice, y se busca el valor correspondiente en esa posición. Esto permite obtener resultados rápidamente, con una complejidad promedio de tiempo O(1), ideal para aplicaciones que requieren recuperación rápida de datos basada en claves.

**El método de búsqueda con tablas hash se elige por varias razones**:

**Eficiencia de búsqueda**: La búsqueda en una tabla hash tiene una complejidad promedio de O(1), lo que significa que la búsqueda es muy rápida y no depende del tamaño de la estructura de datos. Esto lo hace ideal para aplicaciones que necesitan recuperar rápidamente datos basados en claves.

**Estructura de datos dinámica:** Las tablas hash pueden crecer o disminuir dinámicamente según la cantidad de datos, manteniendo un rendimiento eficiente en términos de tiempo de búsqueda.

**Implementación sencilla**: Es relativamente fácil de implementar y utilizar, especialmente en comparación con estructuras de datos más complejas como los árboles de búsqueda.

métodos de búsqueda pueden ser más adecuados:

**Requisito de ordenación**: Las tablas hash no mantienen un orden específico de los elementos, por lo que si es necesario iterar o recorrer los elementos en un orden específico, otros métodos como los árboles de búsqueda podrían ser más apropiados.

**Colisiones**: Si hay muchas colisiones (cuando dos claves diferentes se asignan al mismo índice), puede afectar el rendimiento de la tabla hash. Se pueden implementar técnicas como el manejo de colisiones con listas enlazadas o el rehashing para mitigar este problema, pero en ciertos casos, otros métodos pueden ser más eficientes.

**Complejidad de la función hash**: El rendimiento de una tabla hash depende en gran medida de la función de hash utilizada. Si la función de hash no distribuye bien las claves, puede resultar en un rendimiento deficiente.