

INGENIERIA EN SISTEMAS

Materia: Programación III

ING: JOSÉ MIGUEL VILLATORO HIDALGO

Fecha: 22/05/2024



Hoja de trabajo #8

GRUPO	
Nombre	Carne
Francisco Alexander Chic Barrios	9490-22-2513
Herbert Daniel Jocol Morataya	9490-22-423
Eros Andre Motta Escobar	9490-21-1813

Índice

Introducción	3
Tablas Hash.....	4
Conclusión	8
LINK.....	9

Introducción

Las tablas hash son una de las estructuras de datos fundamentales en informática debido a su eficiencia en la búsqueda, inserción y eliminación de datos. Su funcionamiento se basa en el uso de funciones hash que transforman claves de entrada en índices de una tabla, lo que permite acceder a los datos de manera rápida. Sin embargo, a pesar de su eficiencia, las tablas hash enfrentan desafíos como las colisiones, que ocurren cuando dos claves diferentes producen el mismo índice. Para abordar estos problemas, se han desarrollado diversas técnicas de manejo de colisiones y se ha estudiado la importancia del factor de carga, que mide el nivel de ocupación de la tabla y afecta directamente su rendimiento.

Este trabajo busca profundizar en el estudio de las tablas hash, explorar sus aplicaciones y comprender los conceptos claves asociados, como funciones hash, colisiones y factor de carga. Además, se desarrollará un ejemplo práctico en Python que implementa una tabla hash, permitiendo la inserción manual y masiva de datos, así como la consulta por clave y valor. Este ejercicio práctico servirá para consolidar los conocimientos teóricos y demostrar la aplicación de técnicas adecuadas para evitar colisiones, garantizando un manejo eficiente de datos.

Tablas Hash

Tablas hash

Estructura de datos que se utiliza para implementar un tipo de array asociativo, una estructura que puede asignar claves a valores. Esta estructura es particularmente útil por su eficiencia en la búsqueda, inserción y eliminación de pares clave-valor.

FUNCIONAMIENTO

INSERCIÓN:

Cuando se inserta un par clave-valor, la función hash se aplica a la clave para obtener un índice en el array. El valor se almacena en esa posición de array.

BÚSQUEDA:

Para buscar un valor asociado a una clave se aplica la función hash a la clave para obtener el índice y luego se accede a esa posición en el array.

ELIMINACIÓN:

Similar a la búsqueda, se utiliza la función hash para encontrar la posición del elemento a eliminar y se remueve.

Colisiones

Ocurren cuando dos claves diferentes producen el mismo índice. las principales técnicas para manejar colisiones:

Encadenamiento

Dirección Abierta

Factor carga.

Relación entre el número de elementos en la tabla y el tamaño del array.

Afecta directamente el rendimiento:

Bajo factor de carga (< 0.7)

Alto factor de carga (> 0.7).

Aplicaciones en Informática

Diccionarios: Implementación de diccionarios en lenguajes como python

Caches: Almacenan resultados de operaciones costosas para un acceso rápido

Indices en base de datos:

Para buscar registros

Conjuntos: Implementación de conjuntos donde se necesitan operaciones rápidas de inserción y búsqueda.

Técnicas para evitar colisiones

- Buena función hash
- Rehashing
- Tamaño del Array

Importancia de factor de carga.

El factor de carga influye en la eficiencia de las operaciones de tabla hash. Un factor de carga alto significa más colisiones y por lo tanto un rendimiento más bajo.

Controlar el factor de carga asegura un rendimiento óptimo

Estructura de datos relacionadas

1 Árboles binarios de búsqueda (BST)

Permiten búsqueda, inserción y eliminación en $O(\log n)$ en promedio pero pueden ser menos eficientes.

Listas enlazadas:

Utilizadas en enrutamiento para manejar colisiones

Árboles AVL y Red-Black:

Variantes balanceadas

Tries: Eficiente para operaciones de prefijo y búsqueda de cadenas

Conclusión

El estudio y la implementación de tablas hash proporcionan una comprensión valiosa sobre una de las estructuras de datos más eficientes y versátiles en informática. A través de la investigación, se ha comprendido la importancia de las funciones hash, las técnicas de manejo de colisiones y el impacto del factor de carga en el rendimiento de las tablas. El desarrollo del ejemplo práctico en Python no solo demostró la teoría aprendida, sino que también puso de manifiesto los desafíos y soluciones prácticas en el manejo de datos.

Las tablas hash se utilizan en diversas aplicaciones, desde bases de datos hasta sistemas de archivos y aplicaciones de redes, lo que subraya su relevancia y utilidad en el campo de la informática. La capacidad de insertar, buscar y manipular datos de manera eficiente hace que las tablas hash sean una herramienta indispensable para los desarrolladores. Al implementar un programa que maneja datos masivos de un archivo CSV, se ha demostrado la capacidad de las tablas hash para gestionar grandes volúmenes de información de manera eficiente. Este trabajo no solo ha reforzado el conocimiento teórico, sino que también ha desarrollado habilidades prácticas esenciales para la programación y el diseño de estructuras de datos eficientes.

LINK

<https://github.com/AlexanderCBarrios/ProyectosUMG.git>