



**INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**ESTRUCTURAS DE DATOS**

**GRUPO: 1CM2**

**PRACTICA 1**

**“TABLA HASH”**

**ALUMNO:**

**Barrera Pérez Carlos Tonatihu: 2016630023**

**PROFESOR:**

**Rafael Norman Saucedo Delgado**

# Índice

1. [Introducción](#)
2. [Desarrollo](#)
3. [Pruebas](#)
4. [Conclusión](#)
5. [Referencias](#)

# Introducción

Una tabla hash es una estructura de datos que utiliza claves con valores para realizar almacenamiento y búsqueda de datos. La operación más importante que realiza es la búsqueda, la cual se realiza de manera eficiente además de que permite el acceso a los elementos almacenados a partir de una clave generada a partir de una función hash. Esto quiere decir que con una función diseñada de acuerdo a las necesidades del programa los elementos se transforman en un valor el cual hace referencia a una casilla dentro de la tabla hash.

Cuando se trabaja con tablas hash es frecuente que se produzcan colisiones. Una colisión ocurre cuando para dos elementos de información distintos, la función de dispersión les asigna la misma clave. Para solucionar este problema se debe de trabajar con una función rehash que nos da otra clave para el elemento que genera la colisión.

Existen dos tipos de tablas hash:

- Abiertas: Las colisiones se resuelven insertándose en una lista. De esa forma tendríamos como estructura un arreglo de listas. De este modo si se presenta una colisión el dato a guardar se almacenará en una posición de la lista.
- Cerradas: Utilizamos un arreglo como representación y cuando se produzca una colisión la resolvemos reasignándole otro valor hash a la clave hasta que encontremos un hueco (rehash).

# Desarrollo

El principal que se generó a la hora de desarrollar esta práctica fue pensar en una manera rápida de guardar las palabras para después mostrarlas; pero después de discutirlo con un compañero opte por usar listas para guardar las letras y así simular una palabra, para después guardar esas palabras en las cubetas de la tabla hash. Otro problema que al inicio no me pareció muy obvio fue que la última palabra del archivo no se guardaba, esto era porque en la última iteración sólo se guardaba la última letra de la última palabra pero después ya no entraba en el ciclo, pero este problema se soluciono muy rápido.

# Pruebas

Resultados generales de las pruebas.

## Prueba 1

```
alunno@PC-1:~/Escritorio/tabla_hash/practica.hash$ make
./a.out test.txt

-----Cubeta 0-----
-dolor
-veniam
-dolor
Numero de palabras encontradas: 12

-----Cubeta 1-----
Numero de palabras encontradas: 0

-----Cubeta 2-----
-quis
-quis
-quis
Numero de palabras encontradas: 3

-----Cubeta 3-----
Numero de palabras encontradas: 0

-----Cubeta 4-----
-adipisicing
-magna
-voluptate
Numero de palabras encontradas: 12

-----Cubeta 5-----
-dolore
-ad
-dolore
Numero de palabras encontradas: 15

-----Cubeta 6-----
-ea
-cillum
-ea
Numero de palabras encontradas: 6

-----Cubeta 7-----
-amet
-irure
-amet
Numero de palabras encontradas: 6

-----Cubeta 8-----
-pariatur
-pariatur
-pariatur
Numero de palabras encontradas: 3

-----Cubeta 9-----
-ut
-Ut
```

Prueba 2 con un texto diferente al de la primera prueba

```
$ ./a test.txt
-----Cubeta 0-----
Numero de palabras encontradas: 0
-----Cubeta 1-----
Numero de palabras encontradas: 0
-----Cubeta 2-----
Numero de palabras encontradas: 0
-----Cubeta 3-----
Numero de palabras encontradas: 0
-----Cubeta 4-----
-Carlos
Numero de palabras encontradas: 1
-----Cubeta 5-----
-Estructuras
Numero de palabras encontradas: 1
-----Cubeta 6-----
Numero de palabras encontradas: 0
-----Cubeta 7-----
-IPN
Numero de palabras encontradas: 1
-----Cubeta 8-----
Numero de palabras encontradas: 0
-----Cubeta 9-----
-de
Numero de palabras encontradas: 1
-----Cubeta 10-----
Numero de palabras encontradas: 0
-----Cubeta 11-----
Numero de palabras encontradas: 0
```

```
-----Cubeta 12-----
-Tonatihu
Numero de palabras encontradas: 1
-----Cubeta 13-----
Numero de palabras encontradas: 0
-----Cubeta 14-----
Numero de palabras encontradas: 0
-----Cubeta 15-----
Numero de palabras encontradas: 0
-----Cubeta 16-----
Numero de palabras encontradas: 0
-----Cubeta 17-----
Numero de palabras encontradas: 0
-----Cubeta 18-----
Numero de palabras encontradas: 0
-----Cubeta 19-----
Numero de palabras encontradas: 0
-----Cubeta 20-----
Numero de palabras encontradas: 0
-----Cubeta 21-----
Numero de palabras encontradas: 0
-----Cubeta 22-----
Numero de palabras encontradas: 0
-----Cubeta 23-----
-ESCOM
Numero de palabras encontradas: 1
-----Cubeta 24-----
Numero de palabras encontradas: 0
-----Cubeta 25-----
Numero de palabras encontradas: 0
```

```
-----Cubeta 26-----
Numero de palabras encontradas: 0
-----Cubeta 27-----
-Datos
Numero de palabras encontradas: 1
-----Cubeta 28-----
Numero de palabras encontradas: 0
-----Cubeta 29-----
Numero de palabras encontradas: 0
-----Cubeta 30-----
Numero de palabras encontradas: 0
-----Cubeta 31-----
Numero de palabras encontradas: 0
```

# Conclusión

El desarrollo de una tabla hash es relativamente sencillo, lo complicado es el desarrollo de la función que hará el hash, otro punto importante es la implementación que se le da ya que esta se puede complicar dependiendo de la situación, como en este caso que no resultaba obvio cómo manejar los datos en un inicio.

# Referencias

Wikipedia. (2016). ASCII. mayo 3, 2016, de Wikipedia Sitio web:  
<https://es.wikipedia.org/wiki/ASCII>

Alfred V. Aho, John E. Hopcroft & Jeffrey D. Ullman. (1998). Estructuras de datos y algoritmos. México, DF: S.A. Alhambra Mexicana.

Joyanes, L. (1990). Fundamentos de Programación. Algoritmos y estructura de datos. México: McGraw-Hill.

González, A.. (2002). Tabla Hash. mayo 4, 2016, de Universidad Técnica Federico Santa María Sitio web:  
<http://profesores.elo.utfsm.cl/~agv/elo320/01and02/dataStructures/hashig.pdf>