

# OBSERVACIONES DE LA PRACTICA

Alejandro Salcedo 202321921

Samuel Osorio 202324806

Julián Restrepo 202320177

1) ¿Qué estructura de datos se usa para este índice? Para este índice se usa un mapa o tabla de símbolos

2) ¿Cuántos elementos se espera almacenar inicialmente?

Se desean almacenar 800 elementos. Esto se puede ver en el código, debido a que el primer parámetro para implementar mapas es numelements que equivale a el tamaño o cantidad de elementos que se desea almacenar, para este caso 800.

3) ¿Cuál es el tamaño de las tablas de hash para **'years'** y **'authors'**?

Para authors el tamaño equivale al parámetro numelements, en este caso 800 y para years el tamaño de la tabla de hash es de 40

4) ¿Cuál es el factor de carga máximo?

Para el caso de authors el factor de carga es de 4, lo que significa que el promedio de nodos es de 4 y que cuando lo exceda, haga un rehash. Para el caso de years, el factor

de carga es 0,5 es decir 50% (debido a que es “probing”).

5) ¿Qué hace la instrucción “**mp.put(...)**”?

Esta instrucción añade una pareja llave valor en el hash y reemplaza el valor si la llave ya existe.

6) ¿Qué papel cumple “**book[‘goodreads\_book\_id’]**” en esa instrucción?

Esta parte del código asigna la llave que va a ser usada en la inserción a el hash.

7) ¿Qué papel cumple el tercer parámetro “**book**” en esa instrucción?

En esta instrucción “book” son los datos que van a ser almacenados en el hash después de usar el put function.

8) ¿Qué hace la instrucción “**mp.get(...)**”?

La instrucción mp.get recibe por parámetro la estructura, es decir, el mapa y una llave, esto nos retornará el valor asociado a esa llave que recordemos es única, dado el contexto se ingresa el mapa que es una llave que hace parte del catálogo “catalog[“years”]” y la llave years y se retorna la pareja llave valor correspondiente.

9) ¿Qué papel cumple “**year**” en esa instrucción?

Year ingresa por parámetro como una llave del mapa y después se guarda como el valor correspondiente encontrado por mp.get(), si el valor está disponible se hace uso de la función me.getValue()

10) ¿Qué hace la instrucción titulada “**me.getValue(...)**”?

La instrucción me.getValue() recibe por parámetro el mapa y retorna el valor de una pareja del map, dado el contexto así se obtiene los libros asociados del año específico dentro del catálogo.

11) ¿Cuál es la llave y el valor del nuevo índice implementado?, describa el tipo de dato y/o las estructuras de datos involucradas.

Para este nuevo índice, la llave sería los títulos de los libros y el valor el libro como tal, es decir todos los datos sobre el libro. Para esta implementación decidimos que el valor fuera un diccionario con diferentes características de los libros (promedio, autores, año de publicación, etc). En otras palabras nuestra implementación tiene como llave el título, es decir, un string, y como valor un diccionario con las características del libro

12) ¿Qué pasa cuando existan dos o más libros con el mismo título (ej.: diferente edición)?

El problema de qué suceda esto es que el mapa solo va a retener el último valor asociado a ese título. En otras palabras si dos libros tienen el mismo título va a almacenar únicamente el último título que ingrese al mapa y va a sobrescribir el anterior.

13) Con el índice implementado ¿Cómo solucionaría que dos o más libros tengan el mismo título?

Para solucionar este problema manejaríamos el problema diferente, por ejemplo creando mapas que tengan como llave el título pero como valor otro mapa o una lista con todos los libros bajo ese título. De esta manera, en vez de sobrescribir uno sobre el otro, se guardarían todos los libros bajo ese título en otra estructura de datos diferente.