UNIDAD TEMÁTICA 6 – Diccionarios, Mapas, Hashing e implementaciones JAVA

Trabajo de Aplicación 1

EJERCICIO 1

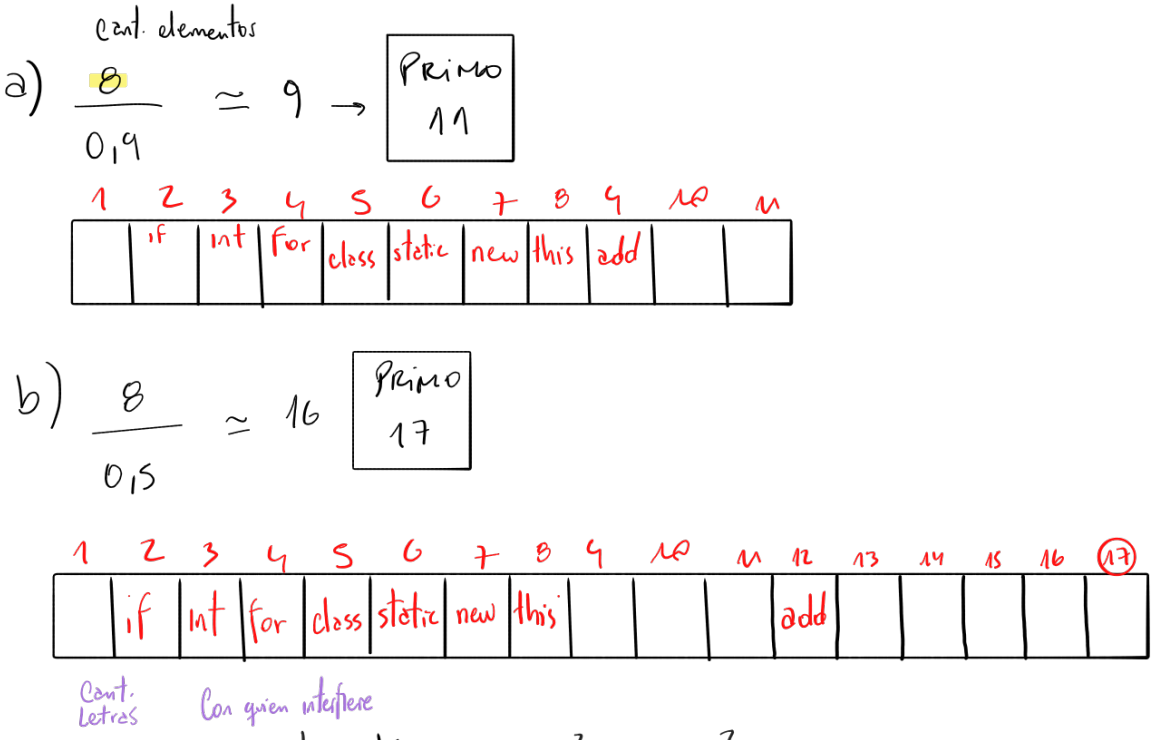
Dado el siguiente conjunto de palabras

if, int, for, static, class, new, this, add

Insertarlas en el orden dado en una tabla hash, usando

a) sondeo lineal (0,9)
b) sondeo cuadrático (0,5)

Para cada caso, calcular el tamaño de tabla adecuado y usar una función sencilla, rápida y adecuada a la cantidad de claves.



EJERCICIO 2

Utilizando los conceptos manejados de TDA, es necesario implementar un THash en Java.

Dadas las siguientes operaciones:

- **public int buscar(int unaClave)** -devuelve la cantidad de comparaciones realizadas-
- **public int insertar(int unaClave)** -devuelve la cantidad de comparaciones realizadas-
- **public int funcionHashing(int unaClave)** -devuelve la posición generada por la función-

Se solicita:

1) Analizar en pseudo-código las operaciones solicitadas.
2) Dimensionar la tabla de hash de acuerdo a las mejores prácticas (para las claves contenidas en el archivo **"claves_insertar.txt"**.
3) Desarrollar una función de hash lo más eficiente posible (pseudo-código).
4) Implementar las operaciones de inserción y búsqueda. Es necesario tener en cuenta que se debe retornar la cantidad de comparaciones realizadas.
5) Preparar el código para insertar y buscar en forma masiva, desde los archivos "claves_buscar.txt" y "claves_insertar.txt"

Notas: El método de resolución de colisiones a usar es el direccionamiento abierto lineal: **h(i) = h(0) + i,** circular. El método principal es el encargado de crear la tabla de hashing, luego de leer las claves del archivo de entrada, estableciendo un tamaño de la tabla igual a **cantClaves/0.9**

Dinámica de trabajo para el ejercicio:

Se ha de trabajar en dos sub-equipos.

PASO 1:

Sub-equipo "A"

- desarrolla el seudocódigo de la inserción y la búsqueda.

Sub-equipo "B"

- desarrolla el seudocódigo de una función de hashing lo más eficiente posible.

PASO 2:

Los sub equipos intercambian los seudocódigos, corrigen y elaboran POSTER final.

EJERCICIO 3

A partir de los seudocódigos desarrollados, se implementa la tabla y sus operaciones de inserción y búsqueda.

PASO 1:

Sub-equipo "A"

- implementa el método con la función de hashing y lo prueba con un conjunto pequeño de datos.

Sub-equipo "B"

Implementa los métodos de inserción y búsqueda.

- Diseña el programa principal encargado de recorrer el archivo con claves a insertar, insertarlas, recorrer el archivo con claves a buscar, buscarlas, y escribir por consola la cantidad de comparaciones realizadas **en promedio** -para las búsquedas exitosas e infructuosas

PASO 2:

- Integrar todo el código y efectuar las pruebas con los archivos provistos, **"claves_buscar.txt"** y **"claves_insertar.txt"**, indicando **cantidad de comparaciones** tanto al insertar como al buscar (y en este último caso también si la búsqueda tiene o no éxito)