Tarea 2 - Diccionarios en Memoria Secundaria

CC4102 - Diseño y Análisis de Algoritmos
Profesor: Gonzalo Navarro Auxiliar: Jorge Bahamonde
Ayudantes: Fabián Mosso - Sebastián Ferrada - Tomás Wolf

1 Introducción

El objetivo de esta tarea es implementar y evaluar en la práctica las diferentes formas de implementar diccionarios en memoria secundaria vistas en clases:

- B-Trees
- Hashing Lineal
- Hashing Extendible

Se espera que se *implementen* los algoritmos, *realicen* los experimentos correspondientes y se entregue un *informe* que indique claramente los siguientes puntos:

1. Las *hipótesis* escogidas antes de realizar los experimentos.
2. El *diseño experimental*, incluyendo los detalles de la implementación de los algoritmos, la generación de las instancias y las medidas de rendimiento utilizadas.
3. La *presentación de los resultados* en forma de una descripción textual, tablas y/o gráficos.
4. El *análisis e interpretación* de los resultados.

2 Implementación

Se pide que implemente cuatro diccionarios en memoria secundaria:

1. B-Trees
2. Hashing Extendible
3. Dos variantes de Hashing Lineal.

Las variantes de Hashing Lineal se diferencian en las políticas utilizadas para la expansión y contracción de la estructura: estas políticas deberán ser escogidas por usted. Explique claramente en su informe las políticas que escoja.

Escoja el parámetro B (el tamaño de la unidad mínima de I/O) para sus estructuras de acuerdo a las características de su máquina. Documente las características de su máquina, el sistema operativo, lenguaje y compilador utilizados, RAM, y características del disco duro.

Funciones de Hash

En esta tarea se trabajará con cadenas de ADN. De esta forma es necesario definir una función de hash adecuada para las estructuras de Hashing Extendible y Lineal. Utilice como hash una codificación binaria de las cadenas, usando dos bits para cada base nitrogenada (G, C, A, T). Así, el hash es la representación binaria de la cadena misma.

3 Experimentos y Datos

Trabaje con los siguientes conjuntos de datos:

Datos aleatorios

Genere 2^{25} cadenas de bases nitrogenadas (es decir, compuestas por los caracteres G, C, A, T) de largo 15 de forma aleatoria.

- Inserte estas cadenas en la estructura, realizando las mediciones que se indican más adelante. Luego de insertar 2^i elementos en la estructura, para $i \in \{20, ..., 25\}$, realice lo siguiente:
  - Obtenga 10000 cadenas elegidas al azar de las ya insertadas. Utilice éstas para probar búsquedas exitosas en la estructura.
  - Obtenga 10000 cadenas de ADN de largo 15, generadas al azar. Utilice éstas para probar búsquedas infructuosas en la estructura.
- Elimine todos los elementos de la estructura, en un orden aleatorio.

Datos reales

Realice el mismo experimento de la parte previa, pero utilizando 2^{25} cadenas (nuevamente de largo 15) extraídas desde el genoma de algún organismo[1]. Note que, para realizar las búsquedas exitosas, deberá obtener cadenas desde las ya insertadas en la estructura, nuevamente.

Mediciones

Obtenga las siguientes medidas de rendimiento:

Ocupación de la estructura

Mida el espacio efectivamente ocupado por la información insertada en la estructura dividido por el espacio comprendido por los bloques de disco utilizados. Mida esta cantidad:

- Luego de insertar 2^i elementos en la estructura, con $i \in \{20, ..., 25\}$.
- Al llegar a 2^i elementos en la estructura en el proceso de borrado, con $i \in \{20, ..., 24\}$.

Operaciones de lectura/escritura en disco

Mida el número de operaciones de escritura y lectura en disco en los siguientes casos:

- Luego de insertar 2^i elementos en la estructura, para $i \in \{20, ..., 25\}$.
- Al realizar las búsquedas de los 2^i elementos (para $i \in \{20, ..., 25\}$) ya insertados (búsquedas exitosas).
- Al realizar las búsquedas de los 2^i elementos (para $i \in \{20, ..., 25\}$) generados al azar (búsquedas infructuosas).
- De la misma forma, mida el número de operaciones de lectura y escritura a disco necesarias al llegar a 2^i elementos en la estructura, durante el proceso de borrado, con $i \in \{20, ..., 24\}$.
- Mida además, el número necesario para vaciar completamente la estructura: así, podrá determinar el número de operaciones necesarias para vaciar una estructura con 2^i elementos, con $i \in \{20, ..., 25\}$.

Determine, además, y de acuerdo a las repeticiones que haga de sus experimentos, el error asociado a los promedios para las mediciones que realice, y utilícelo en sus gráficos.

[1] Vea los enlaces al final de este enunciado para información sobre cómo conseguir estos datos.

4 Entrega de la Tarea

- La tarea puede realizarse en grupos de a lo más 2 personas.
- Se descontará 1 punto por día (o fin de semana) de atraso.
- Para la implementación puede utilizar `C`, `C++`, `Java`. No puede utilizar `Python`. Para el informe se recomienda utilizar LaTeX.
- Escriba un informe claro y conciso. Las ponderaciones del informe y la implementación en su nota final son las mismas.
- La entrega será a través de U-Cursos y deberá incluir el informe junto con el código fuente de la implementación (y todas las indicaciones necesarias para su ejecución).

5 Links

- En `ftp://ftp.ncbi.nih.gov/genomes/` puede encontrar el genoma de diversas especies. Para una especie en particular, la carpeta `Assembled_chromosomes/seq` posee la información genética de cada cromosoma. Lleve todos los caracteres a mayúsculas (o equivalentemente, a minúsculas) y luego elimine aquellos que no correspondan a una de las cuatro bases nitrogenadas. Es probable que necesite utilizar la información de más de un cromosoma (eventualmente, de más de una especie) para obtener todos los datos que necesitará.