Auxiliar 4

Memoria secundaria II

Profesores: Benjamín Bustos y Gonzalo Navarro
Auxiliares: Sergio Rojas y Diego Salas

P1.-

Compararemos dos técnicas clásicas para calcular el join natural de dos tablas en una base de datos relacional. Esta consiste en, fijadas dos columnas c_1 y c_2 de las tablas T_1 y T_2 respectivamente, obtener las filas f_1 y f_2 de T_1 y T_2 tal que $f_1[c_1] = f_2[c_2]$.

- **Nested loop:** Para cada tupla de una relación, se recorre la otra, emitiendo todo par de columnas que satisfaga la igualdad.
- **Merge join:** Se ordenan las filas de ambas tablas por la columna deseada y luego se recorren ambas secuencialmente, buscando los pares que satisfagan la igualdad.

1. Indique el costo, en el modelo de memoria externa, de ambos métodos. Suponga que los tamaños de las tablas son $N_1 > N_2 >> M$.
2. Indique en qué caso usaría cada una, según los tamaños de las tablas, en términos asintóticos.
3. Muchos manejadores de bases de datos indexan las tablas, preparándolas para ciertos joins. Indique nuevamente las complejidades del merge join en caso de que:
   (1) Ambas columnas ya vengan ordenadas.
   (2) Solo la de la tabla mayor venga ordenada.
   (3) Solo la de la tabla menor venga ordenada.

   ¿Cuál es ahora la mejor estrategia, en cada caso?.

P2.-

Dado un grafo $G(V, E)$ dirigido de N nodos y representado por la lista de A pares $(u, v) \in E$ almacenada en memoria externa, calcule de manera eficiente:

- El grado interior y exterior(la cantidad de aristas que llegan y salen de un nodo). En cada caso, debe crear un archivo de N enteros con el grado de cada nodo. Para esto realice lo siguiente:
  (a) Un algoritmo naïve que requiera A lecturas y $2A$ escrituras en disco.
  (b) Un algoritmo de comparaciones eficiente en memoria secundaria y analícelo.
  (c) Un algoritmo que no trabaje con comparaciones y que sea eficiente en memoria secundaria.
  (d) Comentar cómo sería un algoritmo usando hashing.

- Calcular de manera eficiente el cuadrado del grafo $G^2 = G(V, E^2)$, donde:

$$E^2 = \{(u, v) : u, v \in V, \exists w \in V, (u, w) \in E \wedge (w, v) \in E\}$$

  Considerando que $N \leq M$.

- La i-ésima potencia de un grafo $G^i = G(V, E^i)$ y:

$$E^i = \{(u, v) : u, v \in V \wedge \text{existe un camino desde } u \text{ hasta } v \text{ de largo } i\}$$