

## Auxiliar 3 - Memoria Externa

CC4102 - Diseño y Análisis de Algoritmos

Profesor: Gonzalo Navarro      Auxiliar: Jorge Bahamonde

28 de Septiembre del 2015

1. Dados dos arreglos  $A[1, N]$  y  $B[1, N]$  de enteros que no caben en memoria principal, se desea construir (también en disco) el arreglo  $C[i] = A[B[i]]$ . Diseñe un algoritmo eficiente de memoria externa para construir  $C$  y analícelo.
2. Diseñe un algoritmo eficiente para memoria secundaria que, dada una relación binaria  $R$  sobre el dominio  $\{1 \dots n\}$ , determine si  $R$  es simétrica. Suponga que la relación  $R$  se presenta en disco como un archivo secuencial de todos sus pares  $(i, j) \in \{1 \dots n\} \times \{1 \dots n\}$ .
3. Considere un arreglo de  $N$  elementos que no caben en memoria principal. Se desea encontrar un elemento que sea mayoría (es decir, que aparezca más de  $\frac{N}{2}$  veces) y reportar su frecuencia. Si no existe tal elemento, se debe reportar no. Se dispone de una memoria de tamaño  $M = \Theta(1)$ . Diseñe un algoritmo eficiente que resuelva este problema.

**Propuesto:** Extender este problema a encontrar  $k$  elementos que aparezcan más de  $\frac{N}{k+1}$  veces

## Propuestirijillos

1. Dado un grafo dirigido  $G = (V, E)$ , con  $|V| = n$  y  $|E| = e$ , ambos mucho mayores que  $M$ . El grafo es un archivo secuencial de pares  $(u, v)$  en disco, donde se listan las aristas. Se desconoce  $n$  y  $e$ . Considere  $V = \{1 \dots n\}$  para simplificar. Se pide construir y analizar algoritmos eficientes en el modelo de memoria secundaria para:

- Calcular el grado interior y exterior (número de arcos entrantes y salientes) de todos los nodos.
- Calcular el cuadrado del grafo, definido como  $G^2 = (V, E')$ , donde:

$$E' = \{(u, v) \mid u, v \in V \wedge \exists w \in V, (u, w) \wedge (w, v) \in E\}$$

Puede suponer, para este caso, que  $M \geq n$ , pero no que  $M \geq e$ .

- **Bonus track:** calcule cualquier potencia del grafo. Proponga un algoritmo eficiente en memoria y analícelo. No es aceptable una solución que cueste  $k - 1$  veces o más que lo que obtuvo en el punto anterior.
2. Considere dos matrices esparsas de  $N \times N$ , almacenadas en disco en forma de una secuencia  $(i, j, v)$  para cada valor  $A(i, j) = v \neq 0$ . Se requiere obtener el producto de las dos matrices, pero se dispone solamente de una RAM de tamaño  $M = \Theta(N)$ . Diseñe un algoritmo eficiente en términos de  $p$  y  $p'$ , las cantidades de celdas no cero en las dos matrices. (Hint: se puede conseguir básicamente  $O(pp')/(MB)$ )).