

Ejercicio 1: Básicamente es un ciclo que va aplicando juntar. El pseudocódigo debe ser al estilo del pseudocódigo del algoritmo de reunión selectiva por bloques.

- **Algoritmo de reunión selectiva de loop anidado de bloques**

```
for each block  $B_r$  of  $r$  do begin
    for each block  $B_s$  of  $s$  do begin
        for each tuple  $t_r$  in  $B_r$  do begin
            for each tuple  $t_s$  in  $B_s$  do begin
                if  $(t_r, t_s)$  satisface condición de
                    reunión selectiva
                then add  $t_r \cdot t_s$  to the result.
            end
        end
    end
end
```

El costo del algoritmo debería ser una fórmula que depende de cantidad de bloques de r y cantidad de bloques de s .

Ejercicio 2: Leer filmina 47 donde se explica el algoritmo con texto y convertir esa explicación a pseudocódigo. Es una iteración que recorre y procesa las pasadas. (En una pasada primero se dividen las corridas en grupos y luego se combinan las corridas.) Para el estilo de redacción de pseudocódigo tomar el pseudocódigo de las dos filminas anteriores.

Ejercicio 3: pueden asumir que el tamaño de bloque es 4KiB, luego estiman el tamaño de la clave de búsqueda en bytes. Luego plantean una ecuación a resolver:

$$N * \text{tamaño clave búsqueda} + (N+1) * \text{tamaño de puntero} = \text{tamaño de bloque}$$

Averiguan en el Silberschatz el tamaño de puntero.

Despejan el N . Eso da la cantidad de tuplas que entran en el nodo de un índice.

La respuesta del calculo de la cantidad de nodos del árbol B+ es trivial.

Ver (de las filminas) los algoritmos de selección que se pueden aplicar para este ejercicio. Son dos algoritmos. Calcular el costo de cada uno y elegir el de menor costo.

Ejercicio 4: Estudiar los tres algoritmos indicados en las filminas. Entender tanto el algoritmo como su costo. Luego usar los datos numéricos sobre las tablas para calcular los valores numéricos de los costos de los tres algoritmos.

Finalmente estimar el tamaño del resultado de la reunión usando factor de selectividad.

Ejercicio 5: El conectivo negación no está implementado en el libro. Para los items a) y c) transformar los predicados en predicados equivalentes que no involucran negación y luego indicar que algoritmo de implementación usan para cada uno de esos casos.

Para el punto b. analizar si realmente aporta algo usar el índice o conviene mejor usar el algoritmo de búsqueda lineal.

Ejercicio 6: Es similar al ejercicio dado en clase que está resuelto en las filminas de procesamiento de consultas.

Primero leer el ejercicio de las filminas, luego intentar resolver el item b), luego el item a).

Ejercicio 7: Es similar al ejemplo adicional subido en el Moodle. Solo que el ejemplo considera uno de los casos.

Primero leer el ejercicio adicional subido, luego resolver el item a). Luego resolver el item b)

Ejercicio 8: Este ejercicio es del mismo tipo que los dos anteriores, pero con otros operadores. Primero estudiar los operadores remover duplicados y concatenación de tablas de las filminas. Estudiar también los operadores físicos para ellos y su costo.

Luego resolver el ejercicio. Para esto seguir el proceso de aplicar materialización explicado en las filminas 15 y 17 sobre procesamiento de consultas.