

IIC2413 – Bases de Datos

Guía Índices y Algoritmos

Índices

Considere la relación `Estudiante(n_alumno int PRIMARY KEY, nombre varchar(20), edad int)` la cual esta indexada con un Hash Index clustered sobre la llave y además con un B+Tree unclustered sobre la edad. Asuma que la relación tiene 10000 tuplas, que el numero de alumno va desde el 12000 hasta el 21999, que en una pagina caben 50 tuplas o 200 punteros, que la altura del B+Tree es h , que sus hojas estan ocupadas a un 70 % y que el hash index tiene solo una pagina por bucket.

Indique el costo en I/O de las siguientes consultas.

1. `SELECT * FROM Estudiante WHERE n_alumno = 14632683`
2. `SELECT * FROM Estudiante WHERE nombre = 'Jerry Mendoza'`
3. `SELECT * FROM Estudiante WHERE edad = 21`
4. `SELECT * FROM Estudiante WHERE edad >20 AND edad <25`
5. `SELECT * FROM Estudiante WHERE n_alumno >16000`

Algoritmos

Considere las relaciones `R(a int, b int)` y `S(b int, c int)`

1. Explique como funcionaria un Block Nested Loop Join y un Nested Loop Join para computar $R \bowtie S$: por qué el BNLJ puede llegar a ser mucho mas rapido que el NLJ?
2. Imagine que por alguna razón ninguna de las relaciones esta indexada: cómo podría hacer la consulta $R \bowtie S$? (hint: piense en el algoritmo de ordenación)
3. Escriba en pseudo-código como ejecutaría la consulta `SELECT DISTINCT b FROM R`
4. Asuma que ambas relaciones están indexadas con un hash index clustered sobre el atributo b. Explique como podría computar $R \bowtie S$. Asuma que conoce el valor minimo y maximo del atributo b tanto en R como en S.