Tecnologia delle basi di dati
Prova di autovalutazione — 9 maggio 2008

Note

- Gli homework sono facoltativi ma è estremamente importante svolgerli (e anche discuterli ad esempio sul forum), perché le domande proposte nei compiti di esame possono essere molto simili.
- Questo homework verrà discusso in aula nell'esercitazione prevista per il giorno 19 o 20 maggio. La discussione dovrà avvenire attraverso una presentazione delle soluzioni fatta da studenti (e non da parte del docente, che tutt'al più potrà commentarle).

Domanda 1 Siano r_1 ed r_2 due relazioni contenenti rispettivamente N_1 e N_2 ennuple, con fattore di blocco rispettivamente F_1 e F_2. Si supponga che il sistema abbia a disposizione un buffer di dimensione pari a $M = 101$ blocchi. Calcolare il numero di accessi a memoria secondaria necessario per eseguire un join $r_1 \bowtie_{A_1=A_2} r_2$ (con A_1 attributo di r_1 e A_2 attributo di r_2), nei seguenti casi, da considerare separatamente l'uno dall'altro e assumendo che il DBMS sia in grado di eseguire il join solo con il metodo *nested-loop* (eventualmente utilizzando l'accesso diretto tramite un indice invece della scansione interna) e che utilizzi solo strutture primarie disordinate. Si supponga infine che il blocco abbia dimensioni $B = 1$ Kbyte, che i puntatori occupino $p = 4$ byte e i valori dei due attributi in questione occupino $k = 20$ byte ciascuno.

1. $N_1 = 100.000$ e $N_2 = 1.000.000$, con $F_1 = F_2 = 10$; A_2 è la chiave di r_2 mentre i valori di A_1 in r_1 si ripetono mediamente in $e = 6$ ennuple; non vi sono indici
2. $N_1 = 100.000$ e $N_2 = 1.000.000$, con $F_1 = F_2 = 10$; A_1 è la chiave di r_1 mentre i valori di A_2 in r_2 si ripetono mediamente in $e = 6$ ennuple; sono definiti un indice su $r_1(A_1)$ e uno su $r_2(A_2)$
3. $N_1 = 100.000$ e $N_2 = 1.000.000$, con $F_1 = F_2 = 10$; gli attributi coinvolti non sono chiave e hanno, ciascuno, valori che si ripetono mediamente in $e = 6$ ennuple; è definito un indice su $r_1(A_1)$
4. $N_1 = 5.000$ e $N_2 = 1.000.000$, con $F_1 = F_2 = 20$; A_1 è la chiave di r_1 e A_2 è la chiave di r_2; sono definiti un indice su $r_1(A_1)$ e uno su $r_2(A_2)$

Domanda 2
Alcuni DBMS permettono una tecnica di memorizzazione chiamata "co-clustering" o "clustering eterogeneo," in cui un file contiene record di due o più relazioni e tali record sono allocati (ad esempio ordinati) secondo i valori di opportuni campi dell'una e dell'altra relazione. Ad esempio, date due relazioni

- *Ordini(CodiceOrdine,Cliente,Data,Totale)*
- *LineeOrdine(CodiceOrdine,Linea,Prodotto,Importo)*

questa tecnica (con riferimento agli attributi *CodiceOrdine* delle due relazioni) permetterebbe una memorizzazione contigua di ciascun ordine con le rispettive "linee d'ordine," cioè dei prodotti ordinati (ciascun ordine fa riferimento a più prodotti, ognuno su una "linea").
Con riferimento all'esempio, indicare quali delle seguenti operazioni possono trarre vantaggio dall'uso di questa opportunità e quali ne possono essere penalizzate (spiegare la risposta possibilmente anche in termini quantitativi, attraverso l'uso di esempi, che prevedano ipotesi sulle dimensioni dei vari attributi; ipotizzare anche una valutazione di convenienza complessiva, rispetto a possibili frequenze delle tre operazioni):

1. stampa dei dettagli (cioè delle linee d'ordine) di tutti gli ordini (ordinati per codice)
2. stampa dei dettagli di un ordine
3. stampa delle informazioni sintetiche (codice, cliente, data, totale) di tutti gli ordini

Domanda 3
Si considerino un sistema con blocchi di dimensione $B = 1000$ byte e puntatori ai blocchi di $P = 2$ byte e una relazione *R(A,B,C,D,E)* di cardinalità pari circa a $N = 1.000.000$, con ennuple di $L = 50$ byte e campo chiave A di $K = 5$ byte. Valutare i pro e i contro (in termini di numero di accessi a memoria secondaria e trascurando le problematiche relative alla concorrenza) relativamente alla presenza di un indice secondario sulla chiave A e di un altro, pure secondario, su B, in presenza del seguente carico applicativo:

1. inserimento di una nuova ennupla (con verifica del soddisfacimento del vincolo di chiave), con frequenza $f_1 = 100$
2. ricerca di una ennupla sulla base del valore della chiave A con frequenza $f_2 = 100$
3. ricerca di ennuple sulla base del valore di B con frequenza $f_3 = 500$