

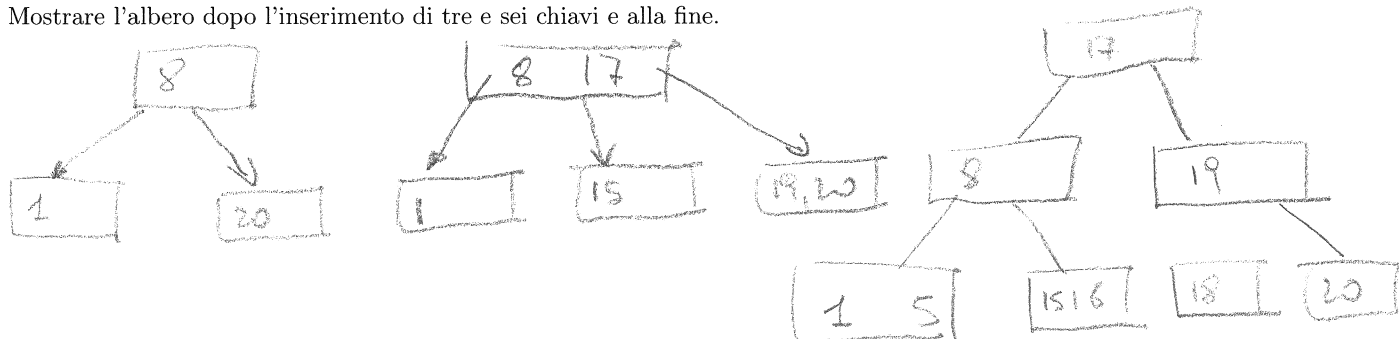
## Basi di dati II, primo modulo — Tecnologia delle basi di dati

### Prova parziale — 27 marzo 2009 — Compito A

Scrivere il nome su questo foglio e su quello protocollo. Rispondere su questo foglio, eventualmente con rimandi (ben chiari) al foglio protocollo ove indispensabile. Usare il foglio protocollo come brutta copia.

Tempo a disposizione: un'ora.

**Domanda 1** Si consideri un B-tree con nodi intermedi che contengono due chiavi e tre puntatori e foglie con due chiavi, in cui vengano chiavi (a partire dall'albero vuoto) nel seguente ordine: 1, 8, 20, 15, 17, 19, 16, 18, 5. Mostrare l'albero dopo l'inserimento di tre e sei chiavi e alla fine.



**Domanda 2** Considerare le seguenti richieste di scrittura (per le quali viene indicato il valore da scrivere) e lettura ricevute da un gestore del controllo di concorrenza basato su timestamp con multiversioni (assumendo che si tratti delle prime richieste ricevute dopo l'avvio del sistema e che i valori di  $x$  e  $y$  siano all'inizio 100 e 200, rispettivamente) e indicare per ciascuna se viene accettata o meno e, per le letture, quale valore viene letto:

$r_3(x)$	SI	100	RTM = 3
$r_2(x)$	SI	100	
$r_4(y)$	SI	200	RTM = 4
$w_2(x, 200)$	NO		
$r_6(y)$	SI	200	RTM = 6
$r_1(x)$	SI	100	
$w_3(x, 300)$	SI		300 WTH = 3
$w_4(y, 400)$	NO		
$w_7(x, 900)$	SI		900 WTH = 7
$w_6(y, 600)$	SI		600 WTH = 6
$r_5(x)$	SI	300	RTM = 5

**Domanda 3** Considerare il seguente log, trovato dopo un crash:

$B(T_1), B(T_2), U(T_2, x), I(T_1, y), B(T_3), C(T_1), B(T_4), U(T_3, z), U(T_4, y), CKPT(T_2, T_3, T_4), C(T_4), B(T_5), U(T_3, w), U(T_5, y), D(T_3, u), A(T_3), C(T_5), I(T_2, v)$

Indicare quali transazioni vanno inserite nei due insiemi UNDO e REDO.

- UNDO:  $T_2, T_3$       ~~2 3 4 5~~
- REDO:  $T_4, T_5$       4 5

## Compito A

**Domanda 4** Alcuni DBMS prevedono la possibilità di definire indici parziali, cioè indici definiti solo su parte di una relazione. Ad esempio, data una relazione  $R(\underline{A}, D, E, F)$ , si può definire un indice

CREATE INDEX RIX ON R (D) WHERE D>100

che supponiamo organizzato come un B+-tree sui valori di  $D$ , limitatamente a quelli che soddisfano la condizione  $D > 100$  (e quindi solo le ennuple che soddisfano la condizione sono accessibili attraverso l'indice).

Indicare la strategia di esecuzione (scansione o accesso diretto) e il costo (indicando la formula e un valore approssimato) di ciascuna delle seguenti interrogazioni, assumendo una struttura heap,  $T = 2.000.000$  ennuple, di  $l = 20$  byte ciascuna ( $l_A = 5$  per ciascun campo), blocchi di dimensione  $B = 2.000$  e puntatori ai blocchi di lunghezza  $p = 5$ ; assumere che i valori di  $D$  siano tutti positivi, che quelli minori di 100 si ripetano ciascuno molte volte (il 95% delle ennuple contiene valori minori di 100) e che quelli maggiori di 100 si ripetano pochissime volte ciascuno (mediamente 2, essendo comunque presenti quasi tutti quelli compresi fra 100 e 1000):

1. SELECT \* FROM R WHERE D=5 AND E>20

strategia: SCANSIONE SEQUENZIALE

costo (formula): N

costo (valore approssimato): 20'000

$$N = \frac{T}{B/l} = 20'000$$

2. SELECT \* FROM R WHERE D=200

strategia: ACCESSO DIRETTO

costo (formula):  $P_1 + 2$   $P_2$  PROFONDITA' INDICE = 3

costo (valore approssimato): 5

3. SELECT \* FROM R WHERE D<200 AND F=20

strategia: SCANSIONE SEQUENZIALE

costo (formula): N

costo (valore approssimato): 20'000

**Domanda 5** Si consideri una base di dati sulle seguenti relazioni, ognuna delle quali ha una struttura heap e un indice secondario sulla chiave (si suppongano le relazioni tutte molto grandi, con  $R_1$  dieci volte più piccola delle altre due)

- $R_1(\underline{ABC})$
- $R_2(\underline{DEF})$
- $R_3(\underline{GHL})$

e con una vista definita come segue:

- CREATE VIEW V AS SELECT \* FROM (R1 LEFT JOIN R2 ON B=D) LEFT JOIN R3 ON C=G

In tale contesto, supponendo che il sistema esegua join (senza trarre profitto significativo dal buffer) solo con nested loop, utilizzando gli indici ove definiti, mostrare un possibile piano di esecuzione (in termini di operatori dell'algebra relazionale e loro realizzazioni) per ciascuna delle seguenti interrogazioni

1. SELECT A, B, C FROM V
2. SELECT A, E, F FROM V
3. SELECT D, E, F FROM V

