

Basi di dati II, primo modulo Prova parziale — 28 marzo 2011— Compito A

Rispondere su questo fascicolo.
Tempo a disposizione: un'ora e quindici minuti.

Cognome _____ Nome _____ Matricola _____ Ordin. _____

Domanda 1 (15%)

Considerare un disco con una velocità di rotazione di $v = 15.000$ giri al minuto e un tempo medio di posizionamento della testina (tempo di seek) $t_S = 5$ msec. Ogni traccia contiene $N = 400$ blocchi e ogni blocco contiene $B = 4$ KB. Rispondere alle seguenti domande mostrando formula e valore numerico (N.B. non servono calcolatrici, i risultati sono semplici, approssimare $1 \text{ MB} = 1000 \text{ KB}$).

1. Qual è tempo medio di latenza (attesa dovuta alla rotazione) t_L ?

2. Qual è la massima velocità di trasferimento (in MB al secondo)?

3. Qual è il tempo minimo t_B necessario per leggere un blocco?

4. Qual è il tempo medio necessario per leggere un blocco?

5. Qual è il tempo medio necessario per leggere una traccia?

Domanda 2 (25%)

Si consideri una base di dati con le relazioni (entrambe con indice sulla chiave primaria)

$T1(\underline{A},B,C)$, $T2(\underline{D},E,F)$

Eseguendo le interrogazioni seguenti, su Postgres (ma il comportamento su altri sistemi è probabilmente lo stesso), si rilevano le seguenti scelte per l'operatore di join:

1.	<code>select * from T1 join T2 on C=D</code>	Hash join
2.	<code>select * from T1 join T2 on C=D where A>2 and A<23</code>	Nested loop join
3.	<code>select A, B, C from T1 join T2 on C=D where A>2 and A<23</code>	Nested loop join

Motivare ciò, valutando, per ciascuna delle tre interrogazioni, il costo di un piano di esecuzione con hash join e uno con nested loop, e supponendo che

- le relazioni abbiano rispettivamente $N_1=2.000.000$ ed $N_2=1.000.000$ ennuple, (con fattore di blocco rispettivamente $f_1=20$ e $f_2=10$)
- gli indici abbiano entrambi $p=4$ livelli (contando anche radice e foglie) e il fattore di blocco massimo dell'indice sia, in entrambi i casi $f_i=90$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=150$.

Rispondere riempiendo la tabella sottostante, indicando il costo in modo sia simbolico sia numerico (considerare, ovviamente, anche l'eventuale selezione)

	Hash join	Nested loop join
1.		
2.		
3.		

Domanda 3 (20%)

Si consideri una base di dati con le relazioni

$R1(\underline{A},B,C,D)$, $R2(\underline{E},F,G)$, $R3(\underline{H},I,L)$, $R4(\underline{M},N,P)$

con un vincolo di integrità referenziale fra l'attributo B di R1 e la chiave E di R2 e con la vista:

- create view V as
select *
from R1 join R2 on B=E
left join R3 on C=H
join R4 on D=M

Per ciascuna delle seguenti interrogazioni, indicare quali join debbono essere eseguiti per rispondere all'interrogazione stessa

select E,F from V	
select H,I,L from V	
select A,B from V	
select A,M,N,P from V	

Domanda 4 (20%)

Si consideri un B-tree con nodi intermedi che contengono due chiavi e tre puntatori e foglie con due chiavi, in cui vengano inserite chiavi (a partire dall'albero vuoto) nel seguente ordine: 42, 1, 4, 8, 10, 12, 14, 16, 20, 24, 5, 6, 13. Mostrare l'albero dopo l'inserimento di quattro, sette, dieci chiavi e alla fine.

Domanda 5 (20%)

Si supponga di avere un recovery manager che utilizzi un checkpoint non quiescente e che scriva record di update (“SetStringRecord” secondo la terminologia di SimpleDB) aventi la forma seguente:

<SETSTRING, TxID, TableName, BlkNo, Offset, BeforeValue, AfterValue >

Si noti che, rispetto alla notazione usata sul libro, l’oggetto dell’operazione viene identificato da TableName (nome della relazione o meglio del file che la memorizza), BlkNo (numero del blocco nel file), Offset (posizione del valore di interesse nel blocco).

In tale contesto, supporre che il recovery manager, al riavvio dopo un crash, trovi i seguenti record nel log:

```
<START, 2>
<SETSTRING, 2, Impiegati, 33, 0, xxxx, Rossi>
<START, 1>
<SETSTRING, 1, Impiegati, 44, 0, xxxx, Neri>
<START, 3>
<COMMIT, 2>
<SETSTRING, 3, Impiegati, 33, 0, Rossi, Verdi>
<START, 4>
<SETSTRING, 4, Impiegati, 55, 0, xxxx, Bruni>
<NQCKPT, 1, 3, 4>
<SETSTRING, 4, Impiegati, 55, 0, Bruni, Neri>
<SETSTRING, 4, Impiegati, 66, 0, xxxx, Bianchi>
<START, 5>
<COMMIT, 4>
```

1. Fino a quale record del log arriva la scansione a ritroso?

2. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-redo?

3. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-only?

4. È possibile che la transazione T_1 abbia modificato il buffer contenente il blocco 70 della relazione Impiegati? Spiegare perché e in caso affermativo spiegarne le conseguenze.

5. È possibile che la transazione T_1 abbia modificato su disco il blocco 70 della relazione Impiegati? Spiegare

6. È possibile che la transazione T_1 abbia modificato su disco il blocco 44 della relazione Impiegati?

Basi di dati II, primo modulo Prova parziale — 28 marzo 2011— Compito B

Rispondere su questo fascicolo.
Tempo a disposizione: un'ora e quindici minuti.

Cognome _____ Nome _____ Matricola _____ Ordin. _____

Domanda 1 (15%)

Considerare un disco con una velocità di rotazione di $r = 15.000$ giri al minuto e un tempo medio di posizionamento della testina (tempo di seek) $t_S = 5$ msec. Ogni traccia contiene $P = 200$ blocchi e ogni blocco contiene $B = 4$ KB. Rispondere alle seguenti domande mostrando formula e valore numerico (N.B. non servono calcolatrici, i risultati sono semplici, approssimare $1 \text{ MB} = 1000 \text{ KB}$).

1. Qual è tempo medio di latenza (attesa dovuta alla rotazione) t_L ?

2. Qual è la massima velocità di trasferimento (in MB al secondo)?

3. Qual è il tempo minimo t_B necessario per leggere un blocco?

4. Qual è il tempo medio necessario per leggere un blocco?

5. Qual è il tempo medio necessario per leggere una traccia?

Domanda 2 (25%)

Si consideri una base di dati con le relazioni (entrambe con indice sulla chiave primaria)

$R1(\underline{A},B,C)$, $R2(\underline{D},E,F)$

Eseguendo le interrogazioni seguenti, su Postgres (ma il comportamento su altri sistemi è probabilmente lo stesso), si rilevano le seguenti scelte per l'operatore di join:

1.	<code>select * from R1 join R2 on C=D</code>	Hash join
2.	<code>select * from R1 join R2 on C=D where A>2 and A<23</code>	Nested loop join
3.	<code>select A, B, C from R1 join R2 on C=D where A>2 and A<23</code>	Nested loop join

Motivare ciò, valutando, per ciascuna delle tre interrogazioni, il costo di un piano di esecuzione con hash join e uno con nested loop, e supponendo che

- le relazioni abbiano rispettivamente $L_1=1.000.000$ ed $L_2=2.000.000$ ennuple, (con fattore di blocco rispettivamente $f_1=10$ e $f_2=20$)
- gli indici abbiano entrambi $i=4$ livelli (contando anche radice e foglie) e il fattore di blocco massimo dell'indice sia, in entrambi i casi $f_i=90$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=150$.

Rispondere riempiendo la tabella sottostante, indicando il costo in modo sia simbolico sia numerico (considerare, ovviamente, anche l'eventuale selezione)

	Hash join	Nested loop join
1.		
2.		
3.		

Domanda 3 (20%)

Si consideri una base di dati con le relazioni

$T1(\underline{A},B,C,D)$, $T2(\underline{E},F,G)$, $T3(\underline{H},I,L)$, $T4(\underline{M},N,P)$

con un vincolo di integrità referenziale fra l'attributo B di T1 e la chiave E di T2 e con la vista:

- create view V as
select *
from T1 join T2 on B=E
join T3 on C=H
left join T4 on D=M

Per ciascuna delle seguenti interrogazioni, indicare quali join debbono essere eseguiti per rispondere all'interrogazione stessa

select E,F from V	
select H,I,L from V	
select A,B from V	
select A,M,N,P from V	

Domanda 4 (20%)

Si consideri un B-tree con nodi intermedi che contengono due chiavi e tre puntatori e foglie con due chiavi, in cui vengano inserite chiavi (a partire dall'albero vuoto) nel seguente ordine: 35, 1, 4, 5, 7, 9, 11, 13, 17, 21, 2, 3, 10. Mostrare l'albero dopo l'inserimento di quattro, sette, dieci chiavi e alla fine.

Domanda 5 (20%)

Si supponga di avere un recovery manager che utilizzi un checkpoint non quiescente e che scriva record di update (“SetStringRecord” secondo la terminologia di SimpleDB) aventi la forma seguente:

<SETSTRING, TxID, TableName, BlkNo, Offset, BeforeValue, AfterValue >

Si noti che, rispetto alla notazione usata sul libro, l’oggetto dell’operazione viene identificato da TableName (nome della relazione o meglio del file che la memorizza), BlkNo (numero del blocco nel file), Offset (posizione del valore di interesse nel blocco).

In tale contesto, supporre che il recovery manager, al riavvio dopo un crash, trovi i seguenti record nel log:

```
<START, 2>
<SETSTRING, 2, Impiegati, 33, 0, xxxx, Rossi>
<START, 1>
<SETSTRING, 1, Impiegati, 44, 0, xxxx, Neri>
<START, 3>
<COMMIT, 2>
<SETSTRING, 3, Impiegati, 33, 0, Rossi, Verdi>
<START, 4>
<SETSTRING, 4, Impiegati, 55, 0, xxxx, Bruni>
<NQCKPT, 1, 3, 4>
<SETSTRING, 4, Impiegati, 55, 0, Bruni, Neri>
<SETSTRING, 4, Impiegati, 66, 0, xxxx, Bianchi>
<START, 5>
<COMMIT, 4>
```

1. Fino a quale record del log arriva la scansione a ritroso?

2. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-redo?

3. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-only?

4. È possibile che la transazione T_1 abbia modificato il buffer contenente il blocco 70 della relazione Impiegati? Spiegare perché e in caso affermativo spiegarne le conseguenze.

5. È possibile che la transazione T_1 abbia modificato su disco il blocco 70 della relazione Impiegati? Spiegare

6. È possibile che la transazione T_1 abbia modificato su disco il blocco 44 della relazione Impiegati?

Basi di dati II, primo modulo Prova parziale — 28 marzo 2011— Compito C

Rispondere su questo fascicolo.
Tempo a disposizione: un'ora e quindici minuti.

Cognome _____ Nome _____ Matricola _____ Ordin. _____

Domanda 1 (15%)

Considerare un disco con una velocità di rotazione di $v = 15.000$ giri al minuto e un tempo medio di posizionamento della testina (tempo di seek) $t_S = 5$ msec. Ogni traccia contiene $P = 400$ blocchi e ogni blocco contiene $B = 4$ KB. Rispondere alle seguenti domande mostrando formula e valore numerico (N.B. non servono calcolatrici, i risultati sono semplici, approssimare $1 \text{ MB} = 1000 \text{ KB}$).

1. Qual è tempo medio di latenza (attesa dovuta alla rotazione) t_L ?

2. Qual è la massima velocità di trasferimento (in MB al secondo)?

3. Qual è il tempo minimo t_B necessario per leggere un blocco?

4. Qual è il tempo medio necessario per leggere un blocco?

5. Qual è il tempo medio necessario per leggere una traccia?

Domanda 2 (25%)

Si consideri una base di dati con le relazioni (entrambe con indice sulla chiave primaria)

$T1(\underline{A},B,C)$, $T2(\underline{D},E,F)$

Eseguendo le interrogazioni seguenti, su Postgres (ma il comportamento su altri sistemi è probabilmente lo stesso), si rilevano le seguenti scelte per l'operatore di join:

1.	<code>select * from T1 join T2 on C=D</code>	Hash join
2.	<code>select * from T1 join T2 on C=D where A>2 and A<23</code>	Nested loop join
3.	<code>select A, B, C from T1 join T2 on C=D where A>2 and A<23</code>	Nested loop join

Motivare ciò, valutando, per ciascuna delle tre interrogazioni, il costo di un piano di esecuzione con hash join e uno con nested loop, e supponendo che

- le relazioni abbiano rispettivamente $N_1=2.000.000$ ed $N_2=1.000.000$ ennuple, (con fattore di blocco rispettivamente $f_1=20$ e $f_2=10$)
- gli indici abbiano entrambi $p=4$ livelli (contando anche radice e foglie) e il fattore di blocco massimo dell'indice sia, in entrambi i casi $f_i=90$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=150$.

Rispondere riempiendo la tabella sottostante, indicando il costo in modo sia simbolico sia numerico (considerare, ovviamente, anche l'eventuale selezione)

	Hash join	Nested loop join
1.		
2.		
3.		

Domanda 3 (20%)

Si consideri una base di dati con le relazioni

$R_1(\underline{A}, B, C, D)$, $R_2(\underline{E}, F, G)$, $R_3(\underline{H}, I, L)$, $R_4(\underline{M}, N, P)$

con un vincolo di integrità referenziale fra l'attributo B di R_1 e la chiave E di R_2 e con la vista:

- create view V as
select *
from R1 join R2 on B=E
join R3 on C=H
left join R4 on D=M

Per ciascuna delle seguenti interrogazioni, indicare quali join debbono essere eseguiti per rispondere all'interrogazione stessa

select E,F from V	
select H,I,L from V	
select A,B from V	
select A,M,N,P from V	

Domanda 4 (20%)

Si consideri un B-tree con nodi intermedi che contengono due chiavi e tre puntatori e foglie con due chiavi, in cui vengano inserite chiavi (a partire dall'albero vuoto) nel seguente ordine: 34, 1, 4, 6, 8, 9, 11, 13, 17, 21, 2, 3, 10. Mostrare l'albero dopo l'inserimento di quattro, sette, dieci chiavi e alla fine.

Domanda 5 (20%)

Si supponga di avere un recovery manager che utilizzi un checkpoint non quiescente e che scriva record di update (“SetStringRecord” secondo la terminologia di SimpleDB) aventi la forma seguente:

<SETSTRING, TxID, TableName, BlkNo, Offset, BeforeValue, AfterValue >

Si noti che, rispetto alla notazione usata sul libro, l’oggetto dell’operazione viene identificato da TableName (nome della relazione o meglio del file che la memorizza), BlkNo (numero del blocco nel file), Offset (posizione del valore di interesse nel blocco).

In tale contesto, supporre che il recovery manager, al riavvio dopo un crash, trovi i seguenti record nel log:

```
<START, 2>
<SETSTRING, 2, Impiegati, 33, 0, xxxx, Rossi>
<START, 1>
<SETSTRING, 1, Impiegati, 44, 0, xxxx, Neri>
<START, 3>
<COMMIT, 2>
<SETSTRING, 3, Impiegati, 33, 0, Rossi, Verdi>
<START, 4>
<SETSTRING, 4, Impiegati, 55, 0, xxxx, Bruni>
<NQCKPT, 1, 3, 4>
<SETSTRING, 4, Impiegati, 55, 0, Bruni, Neri>
<SETSTRING, 4, Impiegati, 66, 0, xxxx, Bianchi>
<START, 5>
<COMMIT, 4>
```

1. Fino a quale record del log arriva la scansione a ritroso?

2. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-redo?

3. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-only?

4. È possibile che la transazione T_1 abbia modificato il buffer contenente il blocco 70 della relazione Impiegati? Spiegare perché e in caso affermativo spiegarne le conseguenze.

5. È possibile che la transazione T_1 abbia modificato su disco il blocco 70 della relazione Impiegati? Spiegare

6. È possibile che la transazione T_1 abbia modificato su disco il blocco 44 della relazione Impiegati?

Basi di dati II, primo modulo Prova parziale — 28 marzo 2011— Compito D

Rispondere su questo fascicolo.
Tempo a disposizione: un'ora e quindici minuti.

Cognome _____ Nome _____ Matricola _____ Ordin. _____

Domanda 1 (15%)

Considerare un disco con una velocità di rotazione di $r = 15.000$ giri al minuto e un tempo medio di posizionamento della testina (tempo di seek) $t_S = 5$ msec. Ogni traccia contiene $N = 200$ blocchi e ogni blocco contiene $B = 4$ KB. Rispondere alle seguenti domande mostrando formula e valore numerico (N.B. non servono calcolatrici, i risultati sono semplici, approssimare $1 \text{ MB} = 1000 \text{ KB}$).

1. Qual è tempo medio di latenza (attesa dovuta alla rotazione) t_L ?

2. Qual è la massima velocità di trasferimento (in MB al secondo)?

3. Qual è il tempo minimo t_B necessario per leggere un blocco?

4. Qual è il tempo medio necessario per leggere un blocco?

5. Qual è il tempo medio necessario per leggere una traccia?

Domanda 2 (25%)

Si consideri una base di dati con le relazioni (entrambe con indice sulla chiave primaria)

$R1(\underline{A},B,C)$, $R2(\underline{D},E,F)$

Eseguendo le interrogazioni seguenti, su Postgres (ma il comportamento su altri sistemi è probabilmente lo stesso), si rilevano le seguenti scelte per l'operatore di join:

1.	<code>select * from R1 join R2 on C=D</code>	Hash join
2.	<code>select * from R1 join R2 on C=D where A>2 and A<23</code>	Nested loop join
3.	<code>select A, B, C from R1 join R2 on C=D where A>2 and A<23</code>	Nested loop join

Motivare ciò, valutando, per ciascuna delle tre interrogazioni, il costo di un piano di esecuzione con hash join e uno con nested loop, e supponendo che

- le relazioni abbiano rispettivamente $L_1=1.000.000$ ed $L_2=2.000.000$ ennuple, (con fattore di blocco rispettivamente $f_1=10$ e $f_2=20$)
- gli indici abbiano entrambi $i=4$ livelli (contando anche radice e foglie) e il fattore di blocco massimo dell'indice sia, in entrambi i casi $f_i=90$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=150$.

Rispondere riempiendo la tabella sottostante, indicando il costo in modo sia simbolico sia numerico (considerare, ovviamente, anche l'eventuale selezione)

	Hash join	Nested loop join
1.		
2.		
3.		

Domanda 3 (20%)

Si consideri una base di dati con le relazioni

$T1(\underline{A},B,C,D)$, $T2(\underline{E},F,G)$, $T3(\underline{H},I,L)$, $T4(\underline{M},N,P)$

con un vincolo di integrità referenziale fra l'attributo B di T1 e la chiave E di T2 e con la vista:

- create view V as
select *
from T1 join T2 on B=E
left join T3 on C=H
join T4 on D=M

Per ciascuna delle seguenti interrogazioni, indicare quali join debbono essere eseguiti per rispondere all'interrogazione stessa

select E,F from V	
select H,I,L from V	
select A,B from V	
select A,M,N,P from V	

Domanda 4 (20%)

Si consideri un B-tree con nodi intermedi che contengono due chiavi e tre puntatori e foglie con due chiavi, in cui vengano inserite chiavi (a partire dall'albero vuoto) nel seguente ordine: 37, 2, 5, 7, 9, 10, 12, 13, 17, 21, 3, 4, 11. Mostrare l'albero dopo l'inserimento di quattro, sette, dieci chiavi e alla fine.

Domanda 5 (20%)

Si supponga di avere un recovery manager che utilizzi un checkpoint non quiescente e che scriva record di update (“SetStringRecord” secondo la terminologia di SimpleDB) aventi la forma seguente:

<SETSTRING, TxID, TableName, BlkNo, Offset, BeforeValue, AfterValue >

Si noti che, rispetto alla notazione usata sul libro, l’oggetto dell’operazione viene identificato da TableName (nome della relazione o meglio del file che la memorizza), BlkNo (numero del blocco nel file), Offset (posizione del valore di interesse nel blocco).

In tale contesto, supporre che il recovery manager, al riavvio dopo un crash, trovi i seguenti record nel log:

```
<START, 2>
<SETSTRING, 2, Impiegati, 33, 0, xxxx, Rossi>
<START, 1>
<SETSTRING, 1, Impiegati, 44, 0, xxxx, Neri>
<START, 3>
<COMMIT, 2>
<SETSTRING, 3, Impiegati, 33, 0, Rossi, Verdi>
<START, 4>
<SETSTRING, 4, Impiegati, 55, 0, xxxx, Bruni>
<NQCKPT, 1, 3, 4>
<SETSTRING, 4, Impiegati, 55, 0, Bruni, Neri>
<SETSTRING, 4, Impiegati, 66, 0, xxxx, Bianchi>
<START, 5>
<COMMIT, 4>
```

1. Fino a quale record del log arriva la scansione a ritroso?

2. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-redo?

3. Quali modifiche sulla base di dati debbono essere eseguite durante un recovery di tipo undo-only?

4. È possibile che la transazione T_1 abbia modificato il buffer contenente il blocco 70 della relazione Impiegati? Spiegare perché e in caso affermativo spiegarne le conseguenze.

5. È possibile che la transazione T_1 abbia modificato su disco il blocco 70 della relazione Impiegati? Spiegare

6. È possibile che la transazione T_1 abbia modificato su disco il blocco 44 della relazione Impiegati?