

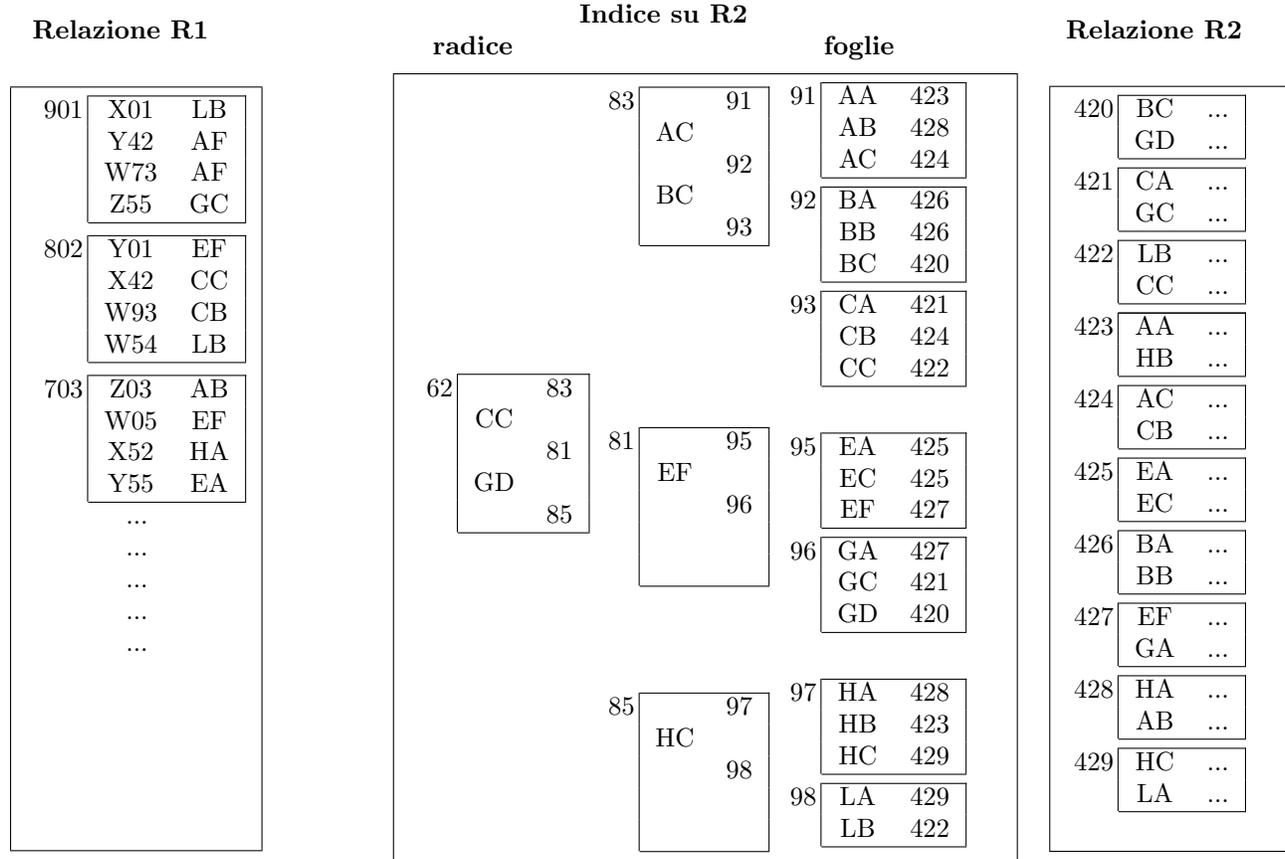
Basi di dati II — Esame — 22 settembre 2017 — Compito A

Tempo a disposizione: due ore.

Cognome _____ Nome _____ Matricola _____

Domanda 1 (20%)

Considerare le relazioni R1 ed R2 e l'indice I2 su R2 schematizzati sotto. I riquadri interni indicano i blocchi e il numero a fianco a ciascun riquadro indica l'indirizzo del blocco. Nell'indice, i valori numerici sono riferimenti ai blocchi (blocchi dell'indice, per la radice e il livello intermedio, e blocchi di R2 per le foglie).



Supponendo di disporre di un buffer di **sette** pagine, considerare l'esecuzione del join di R1 ed R2, sulla base dei valori del secondo attributo di R1 e del primo di R2, con un **nested loop con accesso diretto** tramite l'indice di R2.

Indicare gli indirizzi dei blocchi su cui si eseguono operazioni di pin (o fix) per produrre le prime quattro ennuple del risultato.

Assumendo una politica di rimpiazzo *LRU*, indicare gli indirizzi dei blocchi effettivamente letti da memoria secondaria e caricati nel buffer (nell'ordine) per produrre le prime quattro ennuple del risultato.

In tal caso, indicare gli indirizzi dei blocchi che si può presumere si trovino nei buffer nel momento in cui si produce la quarta ennupla.

Domanda 2 (20%)

Si consideri uno schema dimensionale con la seguente tabella dei fatti (in cui il significato delle dimensioni e misure è intuitivo):

- $FATTIVENDITE(KProdotto, KNegozio, KMese, Ricavo)$

Supporre che la dimensione relativa ai mesi abbia la struttura seguente

- $DIMMESI(KMese, NomeMese, KAnno, Anno, \dots)$

Con riferimento a tale schema, si supponga che

- la tabella dei fatti sia “densa” (cioè che esista una ennupla per ciascun mese, negozio e prodotto) e che i prodotti diversi siano $p = 1000$, i mesi di interesse $m = 48$ (quattro anni) e i negozi $n = 20$.
- le operazioni più frequenti siano:
 1. ricavi complessivi per un prodotto (tutti i negozi) anno per anno, con frequenza $f_1 = 10$
 2. ricavi complessivi (tutti i prodotti e tutti i negozi) mese per mese (nei quattro anni), con frequenza $f_2 = 100$
 3. ricavi complessivi per un negozio (tutti i prodotti) anno per anno, con frequenza $f_3 = 50$
- sia possibile realizzare una sola vista materializzata.

Scegliere, fra le seguenti tre viste, che corrispondono alle tre interrogazioni, quella che supporta meglio il carico applicativo:

1. $RICAVIPRODOTTOANNO(KProdotto, KAnno, Ricavo)$
2. $RICAVIMESE(KMese, Ricavo)$
3. $RICAVINEGOZIOANNO(KNegozio, KAnno, Ricavo)$

Trascurare il costo del join e considerare il costo di ciascuna interrogazione pari al numero delle ennuple della tabella dei fatti (o della vista) coinvolte nell'interrogazione

Domanda 3 (20%)

Considerare una tabella R appena creata (e quindi vuota), con le seguenti ipotesi

- R è definita su due campi, A di lunghezza $a = 16$ byte e B di lunghezza $b = 62$ byte, senza vincoli espliciti di chiave (e quindi le operazioni si possono fare senza verifiche particolari);
- la struttura fisica utilizzata per R è heap, senza indici, con una memorizzazione a lunghezza fissa (in cui supponiamo che, oltre ai byte necessari per i campi, ne servano 2 ulteriori per la memorizzazione) e in cui si marcano come liberi gli spazi dei record eliminati, riutilizzandoli per successivi inserimenti (come avviene in SimpleDB);
- il sistema utilizza blocchi di dimensione $D = 8$ Kbyte (approssimabili a 8000).

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. inserimento di $N = 400.000$ ennuple
2. eliminazione di $N/2 = 200.000$ ennuple (sulla base di una condizione verificabile durante la scansione)
3. dopo la conclusione e la chiusura della scansione precedente, inserimento di altre $N/2 = 200.000$ ennuple
4. ricompattazione della tabella, con utilizzo dei blocchi senza spazi

Rispondere alle domande seguenti, indicando formule e valori numerici:

Numero di scritture di blocchi necessarie per memorizzare i dati nella prima serie di inserimenti (punto 1):
Numero di letture di blocchi necessarie per la fase di eliminazione (punto 2):
Numero di scritture di blocchi necessarie per la fase di eliminazione (punto 2):
Numero di scritture di blocchi necessarie per la seconda serie di inserimenti (punto 3):
Numero di letture di blocchi necessarie per la fase di ricompattazione (punto 4); supporre che i blocchi vengano letti tutti:
Numero di scritture di blocchi necessarie per la fase di ricompattazione (punto 4):

Domanda 4 (20%)

Per ciascuno degli schedule sotto riportati, indicare, scrivendo **sì** o **no** in ciascuna casella, a quali classi appartiene: S (seriale, rispetto a letture e scritture, ignorare commit e abort), CSR (conflict-serializzabile), S2PL (cioè generabile da uno scheduler basato su 2PL stretto), MV (cioè generabile da uno scheduler multiversion con controllo di serializzabilità: “a serializable transaction cannot modify or lock rows changed by other transactions after the serializable transaction began”). Negli schedule, s_i indica l’inizio della transazione i e c_i il suo commit.

	S	CSR	S2PL	MV
$s_2, r_2(x), w_2(x), c_2, s_1, r_1(x), w_1(x), c_1$				
$s_1, r_1(x), s_2, r_2(x), w_1(x), c_1, w_2(x), c_2$				
$s_2, s_1, r_2(x), w_2(x), c_2, r_1(x), w_1(x), c_1$				
$s_1, r_1(x), s_2, r_2(x), w_2(x), r_2(y), w_2(y), c_2, r_1(y), c_1$				

Domanda 5 (20%)

Una catena di supermercati ha una base di dati dei propri clienti che dispongono di una “tessera fedeltà,” con varie informazioni su ciascun cliente, fra cui (a) il totale dei punti acquisiti attraverso l’uso della tessera e (b) il negozio della catena cui fa riferimento (ad esempio, quello presso cui ha inizialmente richiesto la tessera). Si vuole eseguire su di essa l’interrogazione che calcola, per ciascun negozio, il numero dei clienti, la somma dei punti fedeltà dei clienti e la relativa media per cliente. Indicare quale livello di isolamento (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ o SERIALIZABLE) si potrebbe scegliere in ciascuno dei seguenti casi (si supponga che, in generale, sia stato rilevato che, nel corso degli inserimenti e delle modifiche, vengono inseriti valori sbagliati anche di vari ordini di grandezza, che sono poi corretti prima del commit):

1. L’operazione è eseguita mentre vengono inseriti alcuni nuovi clienti (per ciascun negozio pochi rispetto a quelli già presenti), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
2. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di tutti i clienti (a seguito di una ridefinizione dei criteri di assegnazione dei punti stessi), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
3. L’operazione è eseguita in un momento in cui non ci sono aggiornamenti di alcun genere, con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
4. L’operazione è eseguita mentre vengono inseriti alcuni nuovi clienti (per ciascun negozio pochi rispetto a quelli già presenti), con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
5. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di tutti i clienti (a seguito di una ridefinizione dei criteri di assegnazione dei punti stessi), con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.

1.	2.	3.	4.	5.

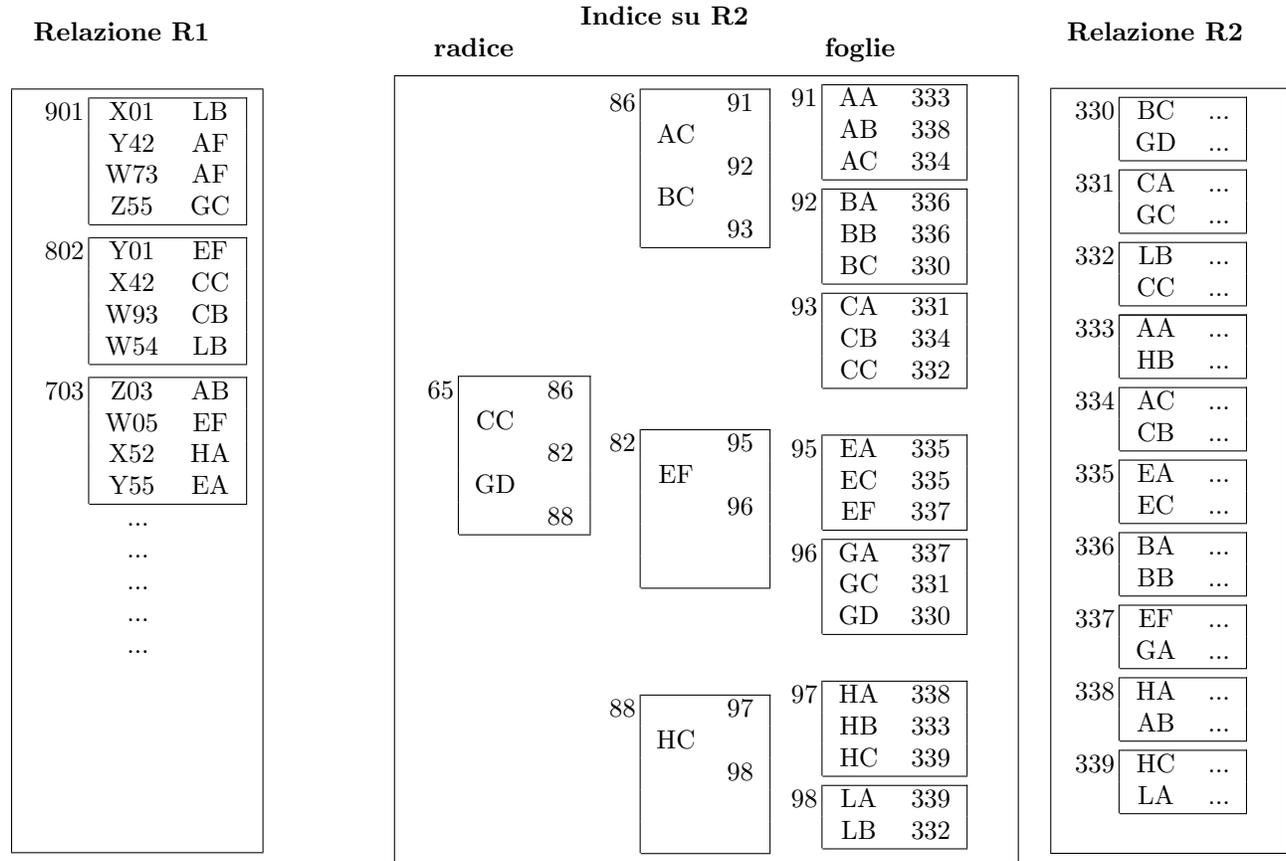
Basi di dati II — Esame — 22 settembre 2017 — Compito B

Tempo a disposizione: due ore.

Cognome _____ Nome _____ Matricola _____

Domanda 1 (20%)

Considerare le relazioni R1 ed R2 e l'indice I2 su R2 schematizzati sotto. I riquadri interni indicano i blocchi e il numero a fianco a ciascun riquadro indica l'indirizzo del blocco. Nell'indice, i valori numerici sono riferimenti ai blocchi (blocchi dell'indice, per la radice e il livello intermedio, e blocchi di R2 per le foglie).



Supponendo di disporre di un buffer di **sette** pagine, considerare l'esecuzione del join di R1 ed R2, sulla base dei valori del secondo attributo di R1 e del primo di R2, con un **nested loop con accesso diretto** tramite l'indice di R2.

Indicare gli indirizzi dei blocchi su cui si eseguono operazioni di pin (o fix) per produrre le prime quattro ennuple del risultato.

Assumendo una politica di rimpiazzo *LRU*, indicare gli indirizzi dei blocchi effettivamente letti da memoria secondaria e caricati nel buffer (nell'ordine) per produrre le prime quattro ennuple del risultato.

In tal caso, indicare gli indirizzi dei blocchi che si può presumere si trovino nei buffer nel momento in cui si produce la quarta ennupla.

Domanda 2 (20%)

Si consideri uno schema dimensionale con la seguente tabella dei fatti (in cui il significato delle dimensioni e misure è intuitivo):

- $FATTI_{VENDITE}(K_{Prodotto}, K_{Negozio}, K_{Mese}, Ricavo)$

Supporre che la dimensione relativa ai mesi abbia la struttura seguente

- $DIM_{MESI}(K_{Mese}, NomeMese, K_{Anno}, Anno, \dots)$

Con riferimento a tale schema, si supponga che

- la tabella dei fatti sia “densa” (cioè che esista una ennupla per ciascun mese, negozio e prodotto) e che i prodotti diversi siano $p = 1000$, i mesi di interesse $m = 48$ (quattro anni) e i negozi $n = 20$.
- le operazioni più frequenti siano:
 1. ricavi complessivi per un prodotto (tutti i negozi) anno per anno, con frequenza $f_1 = 10$
 2. ricavi complessivi (tutti i prodotti e tutti i negozi) mese per mese (nei quattro anni), con frequenza $f_2 = 100$
 3. ricavi complessivi per un negozio (tutti i prodotti) anno per anno, con frequenza $f_3 = 50$
- sia possibile realizzare una sola vista materializzata.

Scegliere, fra le seguenti tre viste, che corrispondono alle tre interrogazioni, quella che supporta meglio il carico applicativo:

1. $RICAVI_{PRODOTTOANNO}(K_{Prodotto}, K_{Anno}, Ricavo)$
2. $RICAVI_{MESE}(K_{Mese}, Ricavo)$
3. $RICAVI_{NEGOZIOANNO}(K_{Negozio}, K_{Anno}, Ricavo)$

Trascurare il costo del join e considerare il costo di ciascuna interrogazione pari al numero delle ennuple della tabella dei fatti (o della vista) coinvolte nell’interrogazione

Domanda 3 (20%)

Considerare una tabella T appena creata (e quindi vuota), con le seguenti ipotesi

- T è definita su due campi, A di lunghezza $a = 6$ byte e B di lunghezza $b = 32$ byte, senza vincoli espliciti di chiave (e quindi le operazioni si possono fare senza verifiche particolari);
- la struttura fisica utilizzata per T è heap, senza indici, con una memorizzazione a lunghezza fissa (in cui supponiamo che, oltre ai byte necessari per i campi, ne servano 2 ulteriori per la memorizzazione) e in cui si marcano come liberi gli spazi dei record eliminati, riutilizzandoli per successivi inserimenti (come avviene in SimpleDB);
- il sistema utilizza blocchi di dimensione $D = 4$ Kbyte (approssimabili a 4000).

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. inserimento di $L = 400.000$ ennuple
2. eliminazione di $L/2 = 200.000$ ennuple (sulla base di una condizione verificabile durante la scansione)
3. dopo la conclusione e la chiusura della scansione precedente, inserimento di altre $L/2 = 200.000$ ennuple
4. ricompattazione della tabella, con utilizzo dei blocchi senza spazi

Rispondere alle domande seguenti, indicando formule e valori numerici:

Numero di scritture di blocchi necessarie per memorizzare i dati nella prima serie di inserimenti (punto 1):
Numero di letture di blocchi necessarie per la fase di eliminazione (punto 2):
Numero di scritture di blocchi necessarie per la fase di eliminazione (punto 2):
Numero di scritture di blocchi necessarie per la seconda serie di inserimenti (punto 3):
Numero di letture di blocchi necessarie per la fase di ricompattazione (punto 4); supporre che i blocchi vengano letti tutti:
Numero di scritture di blocchi necessarie per la fase di ricompattazione (punto 4):

Domanda 4 (20%)

Per ciascuno degli schedule sotto riportati, indicare, scrivendo **sì** o **no** in ciascuna casella, a quali classi appartiene: S (seriale, rispetto a letture e scritture, ignorare commit e abort), CSR (conflict-serializzabile), S2PL (cioè generabile da uno scheduler basato su 2PL stretto), MV (cioè generabile da uno scheduler multiversion con controllo di serializzabilità: “a serializable transaction cannot modify or lock rows changed by other transactions after the serializable transaction began”). Negli schedule, s_i indica l’inizio della transazione i e c_i il suo commit.

	S	CSR	S2PL	MV
$s_2, s_1, r_2(x), w_2(x), c_2, r_1(x), w_1(x), c_1$				
$s_2, r_2(x), w_2(x), c_2, s_1, r_1(x), w_1(x), c_1$				
$s_1, r_1(x), s_2, r_2(x), w_2(x), r_2(y), w_2(y), c_2, r_1(y), c_1$				
$s_1, r_1(x), s_2, r_2(x), w_1(x), c_1, w_2(x), c_2$				

Domanda 5 (20%)

Una catena di supermercati ha una base di dati dei propri clienti che dispongono di una “tessera fedeltà,” con varie informazioni su ciascun cliente, fra cui (a) il totale dei punti acquisiti attraverso l’uso della tessera e (b) il negozio della catena cui fa riferimento (ad esempio, quello presso cui ha inizialmente richiesto la tessera). Si vuole eseguire su di essa l’interrogazione che calcola, per ciascun negozio, il numero dei clienti, la somma dei punti fedeltà dei clienti e la relativa media per cliente. Indicare quale livello di isolamento (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ o SERIALIZABLE) si potrebbe scegliere in ciascuno dei seguenti casi (si supponga che, in generale, sia stato rilevato che, nel corso degli inserimenti e delle modifiche, vengono inseriti valori sbagliati anche di vari ordini di grandezza, che sono poi corretti prima del commit):

1. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di tutti i clienti (a seguito di una ridefinizione dei criteri di assegnazione dei punti stessi), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
2. L’operazione è eseguita mentre vengono inseriti molti nuovi clienti, con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
3. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di alcuni clienti (in ciascun negozio una piccola percentuale), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
4. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di alcuni clienti (in ciascun negozio una piccola percentuale), con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
5. L’operazione è eseguita in un momento in cui non ci sono aggiornamenti di alcun genere, con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.

1.	2.	3.	4.	5.

Basi di dati II — Esame — 22 settembre 2017 — Compito A

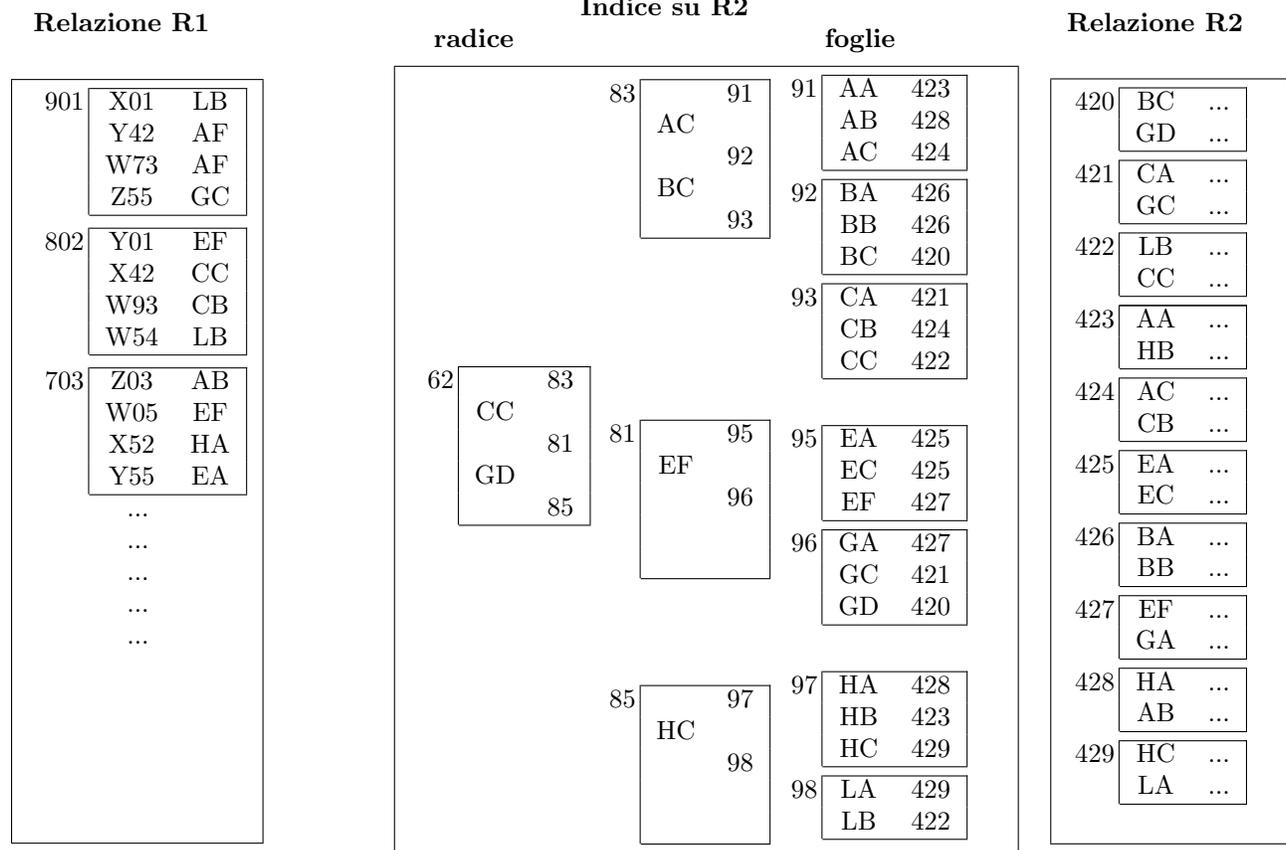
Cenni sulle soluzioni

Tempo a disposizione: due ore.

Cognome _____ Nome _____ Matricola _____

Domanda 1 (20%)

Considerare le relazioni R1 ed R2 e l'indice I2 su R2 schematizzati sotto. I riquadri interni indicano i blocchi e il numero a fianco a ciascun riquadro indica l'indirizzo del blocco. Nell'indice, i valori numerici sono riferimenti ai blocchi (blocchi dell'indice, per la radice e il livello intermedio, e blocchi di R2 per le foglie).



Supponendo di disporre di un buffer di **sette** pagine, considerare l'esecuzione del join di R1 ed R2, sulla base dei valori del secondo attributo di R1 e del primo di R2, con un **nested loop con accesso diretto** tramite l'indice di R2.

Indicare gli indirizzi dei blocchi su cui si eseguono operazioni di pin (o fix) per produrre le prime quattro ennuple del risultato.

901, 62, 85, 98, 422, — produce la ennupla (X01, LB, ...)

62, 83, 92, — cerca AF in R2

62, 83, 92, — cerca AF in R2

62, 81, 96, 421, — produce la ennupla (Z55, GC, ...)

802, 62, 81, 95, 427 — produce la ennupla (Y01, EF, ...)

62, 83, 93, 422, — produce la ennupla (X42, CC, ...)

Assumendo una politica di rimpiazzo *LRU*, indicare gli indirizzi dei blocchi effettivamente letti da memoria secondaria e caricati nel buffer (nell'ordine) per produrre le prime quattro ennuple del risultato.

901, 62, 85, 98, 422,

83, 92,

81 (rimpiazza 85), 96 (rimpiazza 98), 421 (rimpiazza 422),

802 (rimpiazza 83), 95 (rimpiazza 92), 427 (rimpiazza 96)

83 (rimpiazza 901), 93 (rimpiazza 421), 422, (rimpiazza 81)

In tal caso, indicare gli indirizzi dei blocchi che si può presumere si trovino nei buffer nel momento in cui si produce la quarta ennupla.

802, 95, 427, 62, 83, 93, 422

Domanda 2 (20%)

Si consideri uno schema dimensionale con la seguente tabella dei fatti (in cui il significato delle dimensioni e misure è intuitivo):

- $FATTIVENDITE(KProdotto, KNegozi, KMese, Ricavo)$

Supporre che la dimensione relativa ai mesi abbia la struttura seguente

- $DIMMESI(KMese, NomeMese, KAnno, Anno, \dots)$

Con riferimento a tale schema, si supponga che

- la tabella dei fatti sia “densa” (cioè che esista una ennupla per ciascun mese, negozio e prodotto) e che i prodotti diversi siano $p = 1000$, i mesi di interesse $m = 48$ (quattro anni) e i negozi $n = 20$.
- le operazioni più frequenti siano:
 1. ricavi complessivi per un prodotto (tutti i negozi) anno per anno, con frequenza $f_1 = 10$
 2. ricavi complessivi (tutti i prodotti e tutti i negozi) mese per mese (nei quattro anni), con frequenza $f_2 = 100$
 3. ricavi complessivi per un negozio (tutti i prodotti) anno per anno, con frequenza $f_3 = 50$
- sia possibile realizzare una sola vista materializzata.

Scegliere, fra le seguenti tre viste, che corrispondo alle tre interrogazioni, quella che supporta meglio il carico applicativo:

1. $RICAVIPRODOTTOANNO(KProdotto, KAnno, Ricavo)$
2. $RICAVIMESE(KMese, Ricavo)$
3. $RICAVINEGOZIOANNO(KNegozi, KAnno, Ricavo)$

Trascurare il costo del join e considerare il costo di ciascuna interrogazione pari al numero delle ennuple della tabella dei fatti (o della vista) coinvolte nell’interrogazione

1. $RICAVIPRODOTTOANNO$, cardinalità $a \times p = m/12 \times p = 4000$, supporta solo l’interrogazione 1
2. $RICAVIMESE$, cardinalità $m = 48$, supporta solo l’interrogazione 2
3. $RICAVINEGOZIOANNO$, cardinalità $n \times a = n \times m/12 = 80$, supporta solo l’interrogazione 3

Costo unitario per ogni operazione con ciascuna vista:

	con vista 1	con vista 2	con vista 3
c_1	Si deve accedere alle ennuple di un prodotto, nella vista, in tutti gli anni; costo unitario: $a = m/12 = 4$	la vista non aiuta, si deve accedere a tutte le ennuple del prodotto nella relazione base: $m \times n = 960$	la vista non aiuta, si deve accedere a tutte le ennuple del prodotto nella relazione base: $m \times n = 960$
c_2	la vista non aiuta, si deve accedere a tutte le ennuple nella relazione base: $m \times n \times p = 960.000$	Si deve accedere a tutte le ennuple della vista: $m = 48$	la vista non aiuta, si deve accedere a tutte le ennuple nella relazione base: $m \times n \times p = 960.000$
c_3	La vista non aiuta, si deve accedere alle ennuple di un negozio, in tutti i mesi; costo unitario: $p \times m = 48.000$	la vista non aiuta, si deve accedere a tutte le ennuple del negozio nella relazione base: $m \times p = 48.000$	Si deve accedere alle ennuple di un negozio, nella vista in tutti i mesi; costo unitario: $a = 4$
$\sum_i c_i \times f_i$	$4 \times 10 + 960.000 \times 100 + 48.000 \times 50 \approx 9.84 \times 10^7$	$960 \times 10 + 48 \times 100 + 48.000 \times 50 \approx 2.4 \times 10^6$	$960 \times 10 + 960.000 \times 100 + 4 \times 50 \approx 9.84 \times 10^7$

Conviene quindi scegliere la vista 2

Domanda 3 (20%)

Considerare una tabella **R** appena creata (e quindi vuota), con le seguenti ipotesi

- **R** è definita su due campi, **A** di lunghezza $a = 16$ byte e **B** di lunghezza $b = 62$ byte, senza vincoli espliciti di chiave (e quindi le operazioni si possono fare senza verifiche particolari);
- la struttura fisica utilizzata per **R** è heap, senza indici, con una memorizzazione a lunghezza fissa (in cui supponiamo che, oltre ai byte necessari per i campi, ne servano 2 ulteriori per la memorizzazione) e in cui si marcano come liberi gli spazi dei record eliminati, riutilizzandoli per successivi inserimenti (come avviene in SimpleDB);
- il sistema utilizza blocchi di dimensione $D = 8$ Kbyte (approssimabili a 8000).

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. inserimento di $N = 400.000$ ennuple
2. eliminazione di $N/2 = 200.000$ ennuple (sulla base di una condizione verificabile durante la scansione)
3. dopo la conclusione e la chiusura della scansione precedente, inserimento di altre $N/2 = 200.000$ ennuple
4. ricompattazione della tabella, con utilizzo dei blocchi senza spazi

Rispondere alle domande seguenti, indicando formule e valori numerici:

Numero di scritture di blocchi necessarie per memorizzare i dati nella prima serie di inserimenti (punto 1):	fattore di blocco $f = D/(a + b + 2) = 8000/80 = 100$ $N/f = 400.000/100 = 4000$
Numero di letture di blocchi necessarie per la fase di eliminazione (punto 2):	sempre gli stessi blocchi!
Numero di scritture di blocchi necessarie per la fase di eliminazione (punto 2):	sempre gli stessi blocchi!
Numero di scritture di blocchi necessarie per la seconda serie di inserimenti (punto 3):	sempre gli stessi blocchi!
Numero di letture di blocchi necessarie per la fase di ricompattazione (punto 4); supporre che i blocchi vengano letti tutti:	sempre gli stessi blocchi!
Numero di scritture di blocchi necessarie per la fase di ricompattazione (punto 4):	sempre gli stessi blocchi!

Domanda 4 (20%)

Per ciascuno degli schedule sotto riportati, indicare, scrivendo **sì** o **no** in ciascuna casella, a quali classi appartiene: S (seriale, rispetto a letture e scritture, ignorare commit e abort), CSR (conflict-serializzabile), S2PL (cioè generabile da uno scheduler basato su 2PL stretto), MV (cioè generabile da uno scheduler multiversion con controllo di serializzabilità: “a serializable transaction cannot modify or lock rows changed by other transactions after the serializable transaction began”). Negli schedule, s_i indica l’inizio della transazione i e c_i il suo commit.

Soluzioni per il compito A

	S	CSR	S2PL	MV
$s_2, r_2(x), w_2(x), c_2, s_1, r_1(x), w_1(x), c_1$	sì	sì	sì	sì
$s_1, r_1(x), s_2, r_2(x), w_1(x), c_1, w_2(x), c_2$	no	no	no	no
$s_2, s_1, r_2(x), w_2(x), c_2, r_1(x), w_1(x), c_1$	sì	sì	sì	no
$s_1, r_1(x), s_2, r_2(x), w_2(x), r_2(y), w_2(y), c_2, r_1(y), c_1$	no	no	no	sì *

(*) in questo caso $r_1(y)$ legge il valore di y all’inizio della transazione 1, cioè prima della scrittura $w_2(y)$

Domanda 5 (20%)

Una catena di supermercati ha una base di dati dei propri clienti che dispongono di una “tessera fedeltà,” con varie informazioni su ciascun cliente, fra cui (a) il totale dei punti acquisiti attraverso l’uso della tessera e (b) il negozio della catena cui fa riferimento (ad esempio, quello presso cui ha inizialmente richiesto la tessera). Si vuole eseguire su di essa l’interrogazione che calcola, per ciascun negozio, il numero dei clienti, la somma dei punti fedeltà dei clienti e la relativa media per cliente. Indicare quale livello di isolamento (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ o SERIALIZABLE) si potrebbe scegliere in ciascuno dei seguenti casi (si supponga che, in generale, sia stato rilevato che, nel corso degli inserimenti e delle modifiche, vengono inseriti valori sbagliati anche di vari ordini di grandezza, che sono poi corretti prima del commit):

1. L’operazione è eseguita mentre vengono inseriti alcuni nuovi clienti (per ciascun negozio pochi rispetto a quelli già presenti), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
2. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di tutti i clienti (a seguito di una ridefinizione dei criteri di assegnazione dei punti stessi), con la finalità di acquisire informazioni approssimate ma ragionevolmente indicative sugli andamenti complessivi.
3. L’operazione è eseguita in un momento in cui non ci sono aggiornamenti di alcun genere, con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
4. L’operazione è eseguita mentre vengono inseriti alcuni nuovi clienti (per ciascun negozio pochi rispetto a quelli già presenti), con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.
5. L’operazione è eseguita mentre vengono modificati i valori dei punti fedeltà di tutti i clienti (a seguito di una ridefinizione dei criteri di assegnazione dei punti stessi), con la finalità di individuare i primi tre negozi da premiare in una campagna promozionale sulla base dei punti acquisiti dai rispettivi clienti.

Risposte				
1.	2.	3.	4.	5.
RC	RR	RU	S	RR