

Basi di dati II — Prova parziale — 7 maggio 2020

Compito A

Domanda 1 (15%)

Considerare una tabella R appena creata (vuota), con le seguenti ipotesi (molto semplificative, al fine di evitare calcoli complessi)

- R è definita su due campi, A e di lunghezza $a = 8$ byte e B variabile, massimo $b = 10$ byte, senza vincoli 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 **variabile** (in cui supponiamo che, servano 2 ulteriori byte per la memorizzazione; quindi un record occupa fra 10 e 20 byte) e in cui si marciano come liberi gli spazi dei record eliminati, **senza riutilizzarli per successivi inserimenti** (se non dopo una **riorganizzazione** che ricompatti i blocchi); **poiché la lunghezza è variabile, se, in caso di modifica, la lunghezza del campo variabile aumenta, allora l'operazione viene gestita come un'eliminazione seguita da un inserimento**
- il sistema utilizza blocchi di dimensione $D = 100$ byte.

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. **inserimento** di $N = 100$ ennuple con il campo B di **lunghezza nulla**
2. **aggiornamento** di $N/2 = 50$ ennuple (sulla base di una condizione verificabile durante la scansione), con modifica del valore di B **costituito questa volta di una stringa $b = 10$ byte**,
3. **riorganizzazione** del file con **ricompattazione** dei blocchi

Indicare, negli spazi sottostanti, il numero di blocchi occupati dalla relazione, dopo ciascuna delle serie di operazioni (nella casella 1, indicare il numero di blocchi dopo le operazioni di cui al punto 1, e così via; è sufficiente indicare il valore numerico)

Numero dei blocchi occupati da R dopo le operazioni di cui al punto 1:

Numero dei blocchi occupati da R dopo le operazioni di cui al punto 2:

Numero dei blocchi occupati da R dopo le operazioni di cui al punto 3:

Basi di dati II — 7 maggio 2020 — Compito A

Domanda 2 (20%)

Si supponga di voler ordinare una relazione che occupa $N = 100$ blocchi, con un algoritmo di merge-sort a più vie, e considerare due casi:

(A) sono disponibili $A = 12$ pagine di buffer

(B) sono disponibili $B = 6$ pagine di buffer

Per ciascuno dei due casi, indicare nel corrispondente riquadro sottostante il numero di passate utilizzato e, per ciascuna passata (sia essa di ordinamento o di fusione), indicare il numero di pagine di buffer utilizzate (rendere evidente il senso della risposta, con frasi, anche sintetiche, del tipo “X passate, b1 pagine di buffer nella prima, b2 nella seconda, b3 nella terza e così via”)

Domanda 3 (25%)

Considerare un sistema i cui dischi abbiano le seguenti caratteristiche:

- tempo medio di posizionamento della testina (tempo di seek) più tempo medio di latenza (attesa dovuta alla rotazione) $T_0 = 10$ millisecondi
- tempo minimo di lettura di un blocco $T_B = 10$ microsecondi

Rispondere, negli spazi sottostanti, alle seguenti domande mostrando formula e valore numerico (entrambi approssimati)

1. Qual è il tempo medio **in secondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi non contigui, non letti di recente?
2. Qual è il tempo medio **in millisecondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi contigui, non letti di recente?
3. Qual è il tempo **in millisecondi** che si può ipotizzare necessario per eseguire $n = 10$ accessi diretti a record di un file attraverso un indice che abbia profondità $p = 4$ e fan-out (fattore di blocco dell'indice) $f = 50$, non usato di recente?
4. Qual è il tempo **in secondi** che si può ipotizzare necessario per eseguire $K = 40.000$ accessi diretti in tempi ravvicinati a record di un file (molto grande) attraverso un indice che abbia profondità $p = 4$, fan-out $f_I = 50$, con disponibilità di circa $P = 4000$ pagine di buffer?

Basi di dati II — 7 maggio 2020 — Compito A

Domanda 4 (30%)

Si consideri una base di dati con le relazioni (**entrambe con indice sulla chiave primaria ed entrambe con la chiave primaria costituita da interi consecutivi**)

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

Considerare le due interrogazioni seguenti

- | | |
|---|---|
| 1. <code>select *</code>
<code>from R1 join R2 on C=D</code> | 2. <code>select *</code>
<code>from R1 join R2 on C=D</code>
<code>where A>=21 AND A<=25</code> |
|---|---|

Indicare, per ciascuna delle due interrogazioni, il costo di un piano di esecuzione con hash join e di uno con nested loop join e utilizzo dell'indice per l'accesso alla tabella interna; per l'interrogazione 2 si deve ovviamente tenere conto della selezione. Supporre che

- le relazioni abbiano $N_1=10.000$ ed $N_2=20.000$ ennuple, (con fattore di blocco $f_1=10$ e $f_2=20$)
- entrambi gli indici abbiano $p=3$ livelli (radice e foglie incluse) e fattore di blocco massimo $f_i=40$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=200$.

Basi di dati II — 7 maggio 2020 — Compito A

Domanda 5 (30%)

Si consideri una relazione $R(\underline{A}, B, C)$, con record di $L=10$ byte su un sistema con blocchi da $P=100$ byte. Supporre che

- la relazione abbia 10.000 ennuple, l'attributo B abbia 100 valori diversi uniformemente distribuiti e l'attributo C abbia 1000 valori diversi pure uniformemente distribuiti
- la relazione abbia un indice su ciascun attributo, con fan-out medio 10 (trascurare il riempimento parziale)

Considerare le tre interrogazioni seguenti (supporre che ci sia una ennupla con $B=15$ e $C=20$)

	<code>select *</code>		<code>select *</code>		<code>select *</code>
1.	<code>from R</code>	2.	<code>from R</code>	3.	<code>from R</code>
	<code>where B=15</code>		<code>where C=20</code>		<code>where B=15 AND C=20</code>

Indicare, nei riquadri sottostanti, il costo per ciascuna delle tre interrogazioni, con un breve commento esplicativo, se lo si ritiene necessario.

Basi di dati II — Prova parziale — 7 maggio 2020

Compito B

Domanda 1 (15%)

Considerare una tabella T appena creata (vuota), con le seguenti ipotesi (molto semplificative, al fine di evitare calcoli complessi)

- T è definita su due campi, A e di lunghezza $a = 18$ byte e B variabile, massimo $b = 20$ byte, senza vincoli 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 **variabile** (in cui supponiamo che, servano 2 ulteriori byte per la memorizzazione; quindi un record occupa fra 20 e 40 byte) e in cui si marciano come liberi gli spazi dei record eliminati, **senza riutilizzarli per successivi inserimenti** (se non dopo una **riorganizzazione** che ricompatti i blocchi); **poiché la lunghezza è variabile, se, in caso di modifica, la lunghezza del campo variabile aumenta, allora l'operazione viene gestita come un'eliminazione seguita da un inserimento**
- il sistema utilizza blocchi di dimensione $D = 200$ byte.

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. **inserimento** di $L = 100$ ennuple con il campo B di **lunghezza nulla**
2. **aggiornamento** di $L/2 = 50$ ennuple (sulla base di una condizione verificabile durante la scansione), con modifica del valore di B **costituito questa volta di una stringa $b = 20$ byte**,
3. **riorganizzazione** del file con **ricompattazione** dei blocchi

Indicare, negli spazi sottostanti, il numero di blocchi occupati dalla relazione, dopo ciascuna delle serie di operazioni (nella casella 1, indicare il numero di blocchi dopo le operazioni di cui al punto 1, e così via; è sufficiente indicare il valore numerico)

Numero dei blocchi occupati da T dopo le operazioni di cui al punto 1:

Numero dei blocchi occupati da T dopo le operazioni di cui al punto 2:

Numero dei blocchi occupati da T dopo le operazioni di cui al punto 3:

Basi di dati II — 7 maggio 2020 — Compito B

Domanda 2 (20%)

Si supponga di voler ordinare una relazione che occupa $N = 400$ blocchi, con un algoritmo di merge-sort a più vie, e considerare due casi:

(A) sono disponibili $A = 10$ pagine di buffer

(B) sono disponibili $B = 30$ pagine di buffer

Per ciascuno dei due casi, indicare nel corrispondente riquadro sottostante il numero di passate utilizzato e, per ciascuna passata (sia essa di ordinamento o di fusione), indicare il numero di pagine di buffer utilizzate (rendere evidente il senso della risposta, con frasi, anche sintetiche, del tipo “X passate, b1 pagine di buffer nella prima, b2 nella seconda, b3 nella terza e così via”)

Domanda 3 (25%)

Considerare un sistema i cui dischi abbiano le seguenti caratteristiche:

- tempo medio di posizionamento della testina (tempo di seek) più tempo medio di latenza (attesa dovuta alla rotazione) $T_0 = 10$ millisecondi
- tempo minimo di lettura di un blocco $T_B = 10$ microsecondi

Rispondere, negli spazi sottostanti, alle seguenti domande mostrando formula e valore numerico (entrambi approssimati)

1. Qual è il tempo medio **in secondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi non contigui, non letti di recente?
2. Qual è il tempo medio **in millisecondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi contigui, non letti di recente?
3. Qual è il tempo **in millisecondi** che si può ipotizzare necessario per eseguire $n = 10$ accessi diretti a record di un file attraverso un indice che abbia profondità $p = 4$ e fan-out (fattore di blocco dell'indice) $f = 50$, non usato di recente?
4. Qual è il tempo **in secondi** che si può ipotizzare necessario per eseguire $K = 50.000$ accessi diretti in tempi ravvicinati a record di un file (molto grande) attraverso un indice che abbia profondità $p = 4$, fan-out $f_I = 50$, con disponibilità di circa $P = 4000$ pagine di buffer?

Domanda 4 (30%)

Si consideri una base di dati con le relazioni (**entrambe con indice sulla chiave primaria ed entrambe con la chiave primaria costituita da interi consecutivi**)

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

Considerare le due interrogazioni seguenti

- | | |
|---|---|
| 1. <code>select *</code>
<code>from R1 join R2 on C=D</code> | 2. <code>select *</code>
<code>from R1 join R2 on C=D</code>
<code>where A>=21 AND A<=25</code> |
|---|---|

Indicare, per ciascuna delle due interrogazioni, il costo di un piano di esecuzione con hash join e di uno con nested loop join e utilizzo dell'indice per l'accesso alla tabella interna; per l'interrogazione 2 si deve ovviamente tenere conto della selezione. Supporre che

- le relazioni abbiano $L_1=20.000$ ed $L_2=10.000$ ennuple, (con fattore di blocco $f_1=20$ e $f_2=10$)
- entrambi gli indici abbiano $i=3$ livelli (radice e foglie incluse) e fattore di blocco massimo $f_i=40$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=200$.

Basi di dati II — 7 maggio 2020 — Compito B

Domanda 5 (30%)

Si consideri una relazione $R(\underline{A}, B, C)$, con record di $L=10$ byte su un sistema con blocchi da $P=100$ byte. Supporre che

- la relazione abbia 10.000 ennuple, l'attributo B abbia 100 valori diversi uniformemente distribuiti e l'attributo C abbia 1000 valori diversi pure uniformemente distribuiti
- la relazione abbia un indice su ciascun attributo, con fan-out medio 10 (trascurare il riempimento parziale)

Considerare le tre interrogazioni seguenti (supporre che ci sia una ennupla con $B=15$ e $C=20$)

	<code>select *</code>	<code>select *</code>	<code>select *</code>
1.	<code>from R</code>	2. <code>from R</code>	3. <code>from R</code>
	<code>where C=20</code>	<code>where B=15</code>	<code>where B=15 AND C=20</code>

Indicare, nei riquadri sottostanti, il costo per ciascuna delle tre interrogazioni, con un breve commento esplicativo, se lo si ritiene necessario.

Basi di dati II — Prova parziale — 7 maggio 2020

Compito A

Cenni sulle soluzioni

(solo Compito A, le varianti del testo sono in rosso)

Domanda 1 (15%)

Considerare una tabella **R** appena creata (vuota), con le seguenti ipotesi (molto semplificative, al fine di evitare calcoli complessi)

- **R** è definita su due campi, **A** e di lunghezza $a = 8$ byte e **B** variabile, massimo $b = 10$ byte, senza vincoli 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 **variabile** (in cui supponiamo che, servano 2 ulteriori byte per la memorizzazione; quindi un record occupa fra **10** e **20** byte) e in cui si marciano come liberi gli spazi dei record eliminati, **senza riutilizzarli per successivi inserimenti** (se non dopo una **riorganizzazione** che ricompatti i blocchi); **poiché la lunghezza è variabile, se, in caso di modifica, la lunghezza del campo variabile aumenta, allora l'operazione viene gestita come un'eliminazione seguita da un inserimento**
- il sistema utilizza blocchi di dimensione $D = 100$ byte.

In tale contesto, supporre che vengano eseguite le seguenti operazioni

1. **inserimento** di $N = 100$ ennuple con il campo **B** di **lunghezza nulla**
2. **aggiornamento** di $N/2 = 50$ ennuple (sulla base di una condizione verificabile durante la scansione), con modifica del valore di **B** **costituito questa volta di una stringa $b = 10$ byte**,
3. **riorganizzazione** del file con **ricompattazione** dei blocchi

Indicare, negli spazi sottostanti, il numero di blocchi occupati dalla relazione, dopo ciascuna delle serie di operazioni (nella casella 1, indicare il numero di blocchi dopo le operazioni di cui al punto 1, e così via; è sufficiente indicare il valore numerico)

Numero dei blocchi occupati da **R** dopo le operazioni di cui al punto 1:

Per il compito A: $N/(D/(a+2)) = 100/(100/10) = 10$

Per il compito B: $L/(D/(a+2)) = 100/(200/20) = 10$

Numero dei blocchi occupati da **R** dopo le operazioni di cui al punto 2:

N.B. Lo spazio libero non viene riutilizzato e serve nuovo spazio per i record modificati;

Per il compito A: spazio aggiuntivo $(N/2)/(D/(a+b+2)) = 50/(100/20) = 10$; totale $10 + 10 = 20$

Per il compito B: analogo

Numero dei blocchi occupati da **R** dopo le operazioni di cui al punto 3:

N.B. Lo spazio viene recuperato, in effetti ci potrebbe essere sfrido oppure record divisi su due blocchi

Per il compito A: spazio in Byte $(N/2) * (a+2) + (N/2) * (a+b+2) = 1500$ cioè 15 blocchi

Per il compito B: stesso ragionamento e stesso risultato

Domanda 2 (20%)

Si supponga di voler ordinare una relazione che occupa $N = 100$ blocchi, con un algoritmo di merge-sort a più vie, e considerare due casi:

(A) sono disponibili $A = 12$ pagine di buffer

(B) sono disponibili $B = 6$ pagine di buffer

Per ciascuno dei due casi, indicare nel corrispondente riquadro sottostante il numero di passate utilizzato e, per ciascuna passata (sia essa di ordinamento o di fusione), indicare il numero di pagine di buffer utilizzate (rendere evidente il senso della risposta, con frasi, anche sintetiche, del tipo “X passate, b1 pagine di buffer nella prima, b2 nella seconda, b3 nella terza e così via”)

Per il compito A:

caso A: due passate (una di sort e una di merge), 10 pagine nella prima e 10 nella seconda

caso B: tre passate (sort e poi due merge), 5 pagine in ciascuna passata (bastano 4 nella terza)

Per il compito B:

caso A: tre passate (una di sort e due di merge), 8 pagine in ogni passata (bastano 7 nella terza)

caso B: due passate (una di sort e una di merge), 20 pagine in ciascuna passata

Domanda 3 (25%)

Considerare un sistema i cui dischi abbiano le seguenti caratteristiche:

- tempo medio di posizionamento della testina (tempo di seek) più tempo medio di latenza (attesa dovuta alla rotazione) $T0 = 10$ millisecondi
- tempo minimo di lettura di un blocco $TB = 10$ microsecondi

Rispondere, negli spazi sottostanti, alle seguenti domande mostrando formula e valore numerico (entrambi approssimati)

1. Qual è il tempo medio **in secondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi non contigui, non letti di recente?
2. Qual è il tempo medio **in millisecondi** necessario per la scansione sequenziale di un file costituito da $F = 100$ blocchi contigui, non letti di recente?
3. Qual è il tempo **in millisecondi** che si può ipotizzare necessario per eseguire $n = 10$ accessi diretti a record di un file attraverso un indice che abbia profondità $p = 4$ e fan-out (fattore di blocco dell'indice) $f = 50$, non usato di recente?
4. Qual è il tempo **in secondi** che si può ipotizzare necessario per eseguire $K = 40.000$ accessi diretti in tempi ravvicinati a record di un file (molto grande) attraverso un indice che abbia profondità $p = 4$, fan-out $f_I = 50$, con disponibilità di circa $P = 4000$ pagine di buffer?

Per il compito A:

- 1: $F \times (T0 + TB) = \text{ca. } 1 \text{ secondo}$
- 2: $(T0 + TB) + (F - 1) \times TB = \text{ca. } 11 \text{ millisecondi}$
- 3: $((p + 1) + 9 \times (p - 1 + 1)) \times (T0 + TB) = \text{ca } 410 \text{ millisecondi}$
- 4: $(p - 3 + 1) \times K \times (T0 + TB) = \text{ca } 800 \text{ secondi}$

Per il compito B: cambia solo

- 4: $(p - 3 + 1) \times K \times (T0 + TB) = \text{ca } 1000 \text{ secondi}$

Domanda 4 (30%)

Si consideri una base di dati con le relazioni (**entrambe con indice sulla chiave primaria ed entrambe con la chiave primaria costituita da interi consecutivi**)

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

Considerare le due interrogazioni seguenti

- | | |
|---|---|
| 1. <code>select *</code>
<code>from R1 join R2 on C=D</code> | 2. <code>select *</code>
<code>from R1 join R2 on C=D</code>
<code>where A>=21 AND A<=25</code> |
|---|---|

Indicare, per ciascuna delle due interrogazioni, il costo di un piano di esecuzione con hash join e di uno con nested loop join e utilizzo dell'indice per l'accesso alla tabella interna; per l'interrogazione 2 si deve ovviamente tenere conto della selezione. Supporre che

- le relazioni abbiano $N_1=10.000$ ed $N_2=20.000$ ennuple, (con fattore di blocco $f_1=10$ e $f_2=20$)
- entrambi gli indici abbiano $p=3$ livelli (radice e foglie incluse) e fattore di blocco massimo $f_i=40$
- l'operazione possa contare su un numero di pagine di buffer pari a circa $q=200$.

Per il compito A:

1 hash: 6000
1 NL: 21.000
2 hash: trascurabile la selezione, poi la scansione di R1, cosro 1000
2 NL: $3+5 + 5 \times 3$

Per il compito B, l'unica differenza:

1 NL: 41.000

Domanda 5 (30%)

Si consideri una relazione $R(\underline{A}, B, C)$, con record di $L=10$ byte su un sistema con blocchi da $P=100$ byte. Supporre che

- la relazione abbia 10.000 ennuple, l'attributo B abbia 100 valori diversi uniformemente distribuiti e l'attributo C abbia 1000 valori diversi pure uniformemente distribuiti
- la relazione abbia un indice su ciascun attributo, con fan-out medio 10 (trascurare il riempimento parziale)

Considerare le tre interrogazioni seguenti (supporre che ci sia una ennupla con $B=15$ e $C=20$)

	<code>select *</code>	<code>select *</code>	<code>select *</code>
1.	<code>from R</code>	2. <code>from R</code>	3. <code>from R</code>
	<code>where B=15</code>	<code>where C=20</code>	<code>where B=15 AND C=20</code>

Indicare, nei riquadri sottostanti, il costo per ciascuna delle tre interrogazioni, con un breve commento esplicativo, se lo si ritiene necessario.

Per il compito A:

1. Si visita l'indice e poi si accede (attraverso 10 foglie) ai 100 record con $B=15$; costo $3 + 10 + 100$
2. Si visita l'indice e poi si accede ai 10 record con $B=15$; costo $4 + 10$
3. Si visitano i due indici e poi si fa l'intersezione degli indirizzi nelle foglie e infine il record;
costo $3 + 1 + 4 + 1$

Per il compito B: come sopra, invertendo 1 e 2