

## i Frontpage

Institutt for datateknologi og informatikk

**Eksamensoppgave i TDT4145 Datamodellering og databasesystemer**

**Eksamensdato:** 6. juni 2023

**Eksamenstid (fra-til):** 09:00 - 11:00

**Hjelpemiddelkode/Tillatte hjelpemidler:**

**D:** Ingen trykte eller håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.

**Faglig kontakt under eksamen:**

**Svein Erik Bratsberg, Tlf.: 99539963**

**Faglig kontakt møter i eksamenslokalet:** JA

### **ANNEN INFORMASJON:**

**Skaff deg overblikk over oppgavesettet** før du begynner på besvarelsen din.

**Les oppgavene nøye**, gjør dine egne antagelser og presiser i besvarelsen hvilke forutsetninger du har lagt til grunn i tolkning/avgrensing av oppgaven. Faglig kontaktperson skal kun kontaktes dersom det er direkte feil eller mangler i oppgavesettet. Henvend deg til en eksamensvakt hvis du ønsker å kontakte faglærer. Noter gjerne spørsmålet ditt på forhånd.

**InsperaScan:** I oppgave 6 er det lagt opp til å besvare på ark. Andre oppgaver skal besvares direkte i Inspera. Nederst i oppgaven finner du en sjusifret kode. Fyll inn denne koden øverst til venstre på arkene du ønsker å levere. Det anbefales å gjøre dette underveis i eksamen. Dersom du behøver tilgang til kodene etter at eksamenstiden har utløpt, må du klikke «Vis besvarelse».

**Vekting av oppgavene:** Vektingen av hver oppgave er angitt i overskriften til oppgaven.

**Varslinger:** Hvis det oppstår behov for å gi beskjeder til kandidatene underveis i eksamen (f.eks. ved feil i oppgavesettet), vil dette bli gjort via varslinger i Inspera. Et varsel vil dukke opp som en dialogboks på skjermen. Du kan finne igjen varselet ved å klikke på bjella øverst til høyre.

**Trekk fra/avbrutt eksamen:** Blir du syk under eksamen, eller av andre grunner ønsker å levere blankt/avbryte eksamen, gå til "hamburgermenyen" i øvre høyre hjørne og velg «Lever blankt». Dette kan ikke angres selv om prøven fremdeles er åpen.

**Tilgang til besvarelse:** Etter eksamen finner du besvarelsen din i arkivet i Inspera. Merk at det kan ta én virkedag før eventuelle håndtegninger vil være tilgjengelige i arkivet.

## 1 Task 1 - Normalization (3 %)

Gitt  $R(A,B,C,D,E)$  og  $F=\{BC \rightarrow A; D \rightarrow C\}$ .

Velg de alternativene som er kandidatnøkler (candidate keys) i R

**Velg ett eller flere alternativer**

- ☐ BDE
- ☐ BCE
- ☐ BE
- ☐ BC
- ☐ Ingen av de andre alternativene
- ☐ DE
- ☐ BD

---

Maks poeng: 3

## 2 Task 2 - Normalization (3 %)

Gitt  $R(A,B,C,D,E, F)$  og  $G=\{C \rightarrow ABD; D \rightarrow CEF\}$

Anta at  $R$  er på 1. normalform (1NF). Velg alle normalformer som  $R$  oppfyller

**Velg ett eller flere alternativer**

- ☐ Tredje normalform (3NF)
- ☐ Boyce-Codd normalform (BCNF)
- ☐ Andre normalform (2NF)
- ☐ Fjerde normalform (4NF)
- ☐ Ingen av de andre alternativene (1NF er høyeste)

---

Maks poeng: 3

## 3 Task 3 - Normalization (3 %)

Gitt  $R(A,B,C,D,E,F)$  som dekomponeres i  $R_1(A,B,C)$  og  $R_2(D,E,F)$ . De avhengighetene som gjelder er  $G=\{C \rightarrow AB; D \rightarrow EF\}$ .

Har denne dekomponeringen tapsløst-join-egenskapen (lossless join)?

**Velg ett eller flere alternativer**

- ☐ Nei, join av  $R_1$  og  $R_2$  kan generere tupler som ikke er i  $R$
- ☐ Ja, fordi  $C$  er primærnøkkel i  $R_1$  og  $D$  er primærnøkkel i  $R_2$
- ☐ Ja, fordi  $C \twoheadrightarrow AB$  vil gjelde i  $R_1$  og  $D \twoheadrightarrow EF$  vil gjelde i  $R_2$
- ☐ Ja, fordi alle rader i  $R_1$  vil kombineres med alle rader i  $R_2$

---

Maks poeng: 3

## 4 Task 4 - SQL and Relational algebra (5 %)

Vi har tabellene:

**Photo**(ID, Title, PhotographerID)

-- PhotographerID is a foreign key against the Photographer table. Can have NULL value

**Photographer**(ID, Name)

Gitt spørringen:

```
select Photographer.ID, count(Photo.ID)
from Photo join Photographer on (Photo.PhotographerID = Photographer.ID)
group by Photographer.ID
```

Vi har tre andre spørringer:

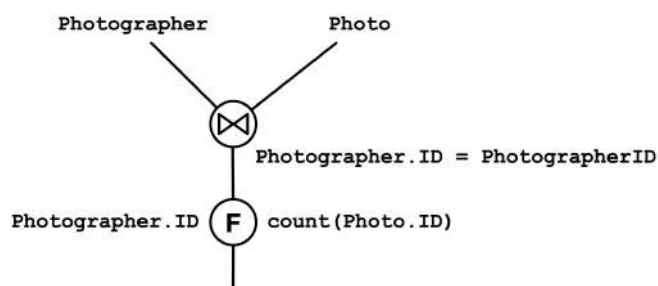
**A:**

```
select Photographer.ID, count(Photo.ID)
from Photo cross join Photographer
group by Photographer.ID
```

**B:**

```
select Photographer.ID, count(Photo.ID)
from Photo natural join Photographer
group by Photographer.ID
```

**C:**



Hvilke av disse spørringene vil alltid gi samme resultat som den oppgitte spørringen?

**Velg ett eller flere alternativer**

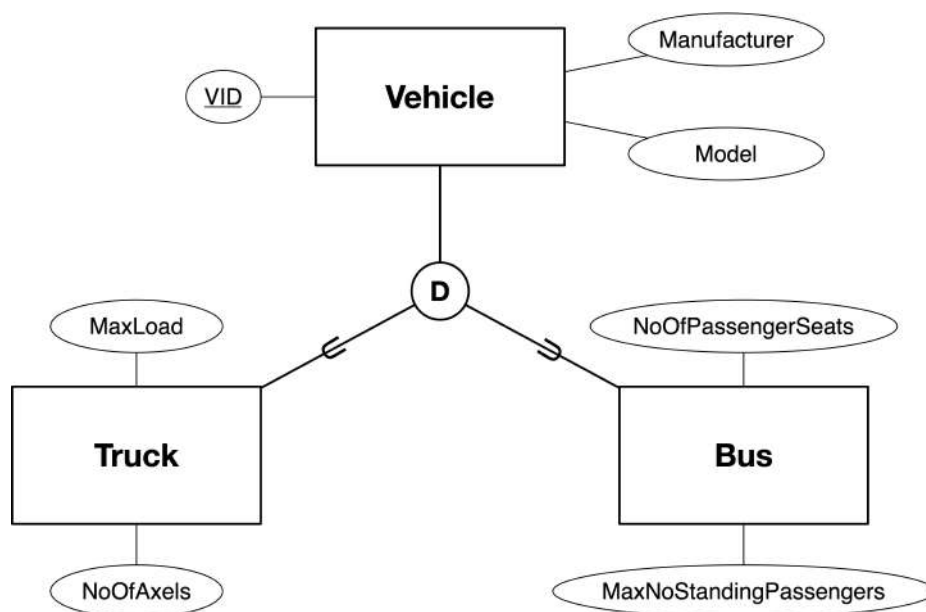
- ☐ C
- ☐ B
- ☐ A
- ☐ Ingen av de andre alternativene

---

Maks poeng: 5

## 5 Task 5 - Data Models (5 %)

Gitt datamodellen:



Oversett til et relasjonsdatabaseskjema. Spesifiser eventuelle forutsetninger som du legger til grunn. Hvilke andre alternativer ville vært mulig (du trenger ikke å vise alternativer i detalj, det er nok å forklare hovedtrekkene i det som ville vært annerledes)? Hvorfor mener du ditt forslag er det beste?

Skriv ditt svar her

Format | **B** | *I* | U |  $x_2$  |  $x^2$  |  $I_x$  | | | | | | | | | | |



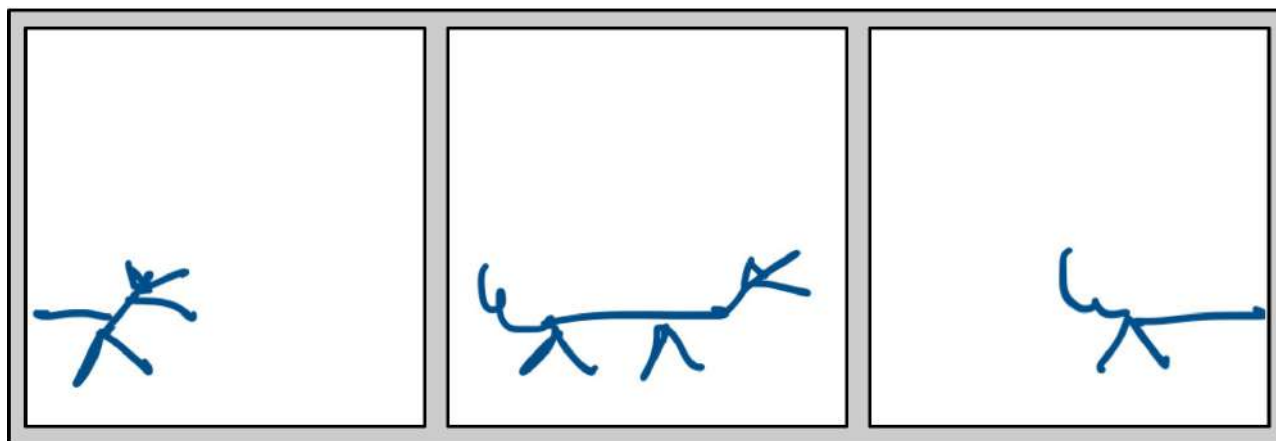
---

Maks poeng: 5

## 6 Task 6 - Data Models (15 %)

Du skal lage en ER-modell (du kan bruke alle modellelement som er med i pensum, også spesialisering og kategorier) ut fra denne situasjonen («miniverden»):

Vi skal holde oversikt over ulike tegneserier. Hver tegneserie har et navn som Asterix, Pondus eller Storefri. Hver tegneserie har en unik id, årstall da tegneserien oppsto og et opprinnelsesland. Innen en tegneserie kan det finnes et antall historier. Hver historie har et navn og et årstall. Historiene har et løpenummer innen tegneserien som viser publiseringsrekkefølgen av historiene. En historie består av et antall rammer som nummereres fra en og oppover. Under er vist et eksempel på en historie som består av tre rammer.



I en tegneserie finnes det vanligvis et antall karakterer. Karakterer har unik karakter-id og navn. En karakter kan delta i ulike tegneserier og må delta i minst en tegneserie. En tegneserie-historie skapes av forfattere som skriver teksten og tegnere som tegner rammene. Forfattere og tegnere registreres med unik person-ID, navn og fødselsår. Alle forfattere og alle tegnere må ha bidratt til minst en tegneserie. Dersom vi vet at en tegner har tegnet en ramme, skal vi kunne registrere dette. Tilsvarende kan en forfatter ha bidratt med tekst til en bestemt historie, dette skal kunne registreres. Det kan være flere tegnere på samme ramme og flere forfattere på samme historie. Karakterer kan inngå i en eller flere rammer, dette kan være som tegnet karakter (er vist i rammen) eller som omtalt karakter (i rammens tekst), eller begge deler. Det skal være mulig å finne alle rammer der en karakter vises, alle rammer der en karakter omtales, alle rammer der en karakter både vises og omtales, og alle rammer der en karakter vises, men ikke omtales.

Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

Denne oppgaven må løses på papir.

**Skriv ditt svar her**

Format
|
B
I
U
x<sub>2</sub>
x<sup>2</sup>
I<sub>x</sub>
|
📄
📁
|
↶
↷
↺
|
≡
≡
|
Ω
📊
|
✎
|
Σ
|

✖

Words: 0

Maks poeng: 15

## 7 Task 7 - Heapfile (6 %)

Vi har poster som er 100 bytes store. Vi har en tabell med 10 000 poster. Hvor mange blokker har vi i en heapfil når blokkene er 4096 bytes store? Anta det kun lagres hele poster i blokkene.

**Velg ett alternativ:**

- ☐ Ingen av de andre alternativene stemmer
- ☐ 750
- ☐ 250
- ☐ 125
- ☐ 500

Maks poeng: 6



## 8 Task 8 - Clustered B+-tre (6 %)

Vi har poster som er 120 byte store.

Vi har en tabell med 8 000 poster. Hvor mange blokker har vi på level=0 i et clustered B+-tre når blokkene er 4096 bytes store og vi antar det ikke lagres noe annet i blokkene enn poster? Anta fyllingsgrad er  $2/3$ , dvs. blokkene vil ikke være mer enn  $2/3$  fulle og at vi lagrer kun hele poster i blokkene.

**Velg ett alternativ:**

- ☐ 235
- ☐ 364
- ☐ 1130
- ☐ 728
- ☐ 240
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 6

## 9 Task 9 - Extendible hashing (7 %)

Vi har en extendible hashing-struktur hvor vi starter med 4 blokker og 4 pekere fra directory. Det er plass til 3 nøkler i hver blokk. Hvor mange blokker med lokal dybde=3 får vi når vi har satt inn følgende sekvens med nøkler:

1, 2, 4, 5, 11, 12, 6, 7, 13, 16, 17

Du kan bruke hashfunksjonen  $h(K) = K \text{ MOD } 8$ .

**Velg ett alternativ:**

- ☐ 0
- ☐ 1
- ☐ 4
- ☐ 3
- ☐ 2
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 7

**10 Task 10 - Join (7 %)**

Vi skal joine to tabeller **Class** og **Student** med nested loop-join.

**Class** består av 30 blokker.

**Student** består av 1000 blokker.

Hvor mange blokker leses når vi skal joine disse to tabellene og vi har plass til 7 blokker i buffer?

**Velg ett alternativ:**

- ☐ 3090
- ☐ 4120
- ☐ 1030
- ☐ 6030
- ☐ 2060
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 7

**11 Task 11 - Sorting (7 %)**

Vi har en usortert fil bestående av 512 blokker med poster og et buffer ned plass til 5 blokker. Hvor mange blokker leses og skrives til sammen når postene skal sorteres ved hjelp av flettesortering? En lesing og en skriving av en blokk blir til sammen 2.

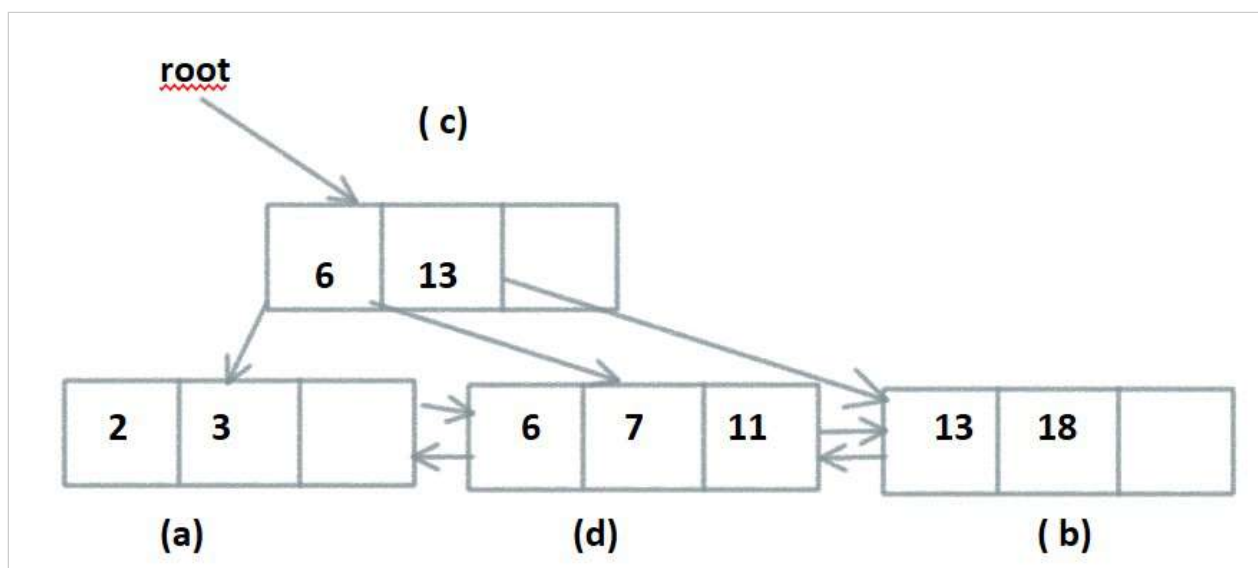
**Velg ett alternativ:**

- ☐ 2048
- ☐ 5120
- ☐ 2560
- ☐ Ingen av de andre alternativene stemmer
- ☐ 10240
- ☐ 1024

---

Maks poeng: 7

## 12 Task 12 - Access paths (2 %)



Tabellen heter 'Tab' og søkenøkkelen heter 'mykey'. Den er lagret i et clustered B+-tre som vist i figuren.

Hvilke sekvens av blokker aksesseres ved følgende query?

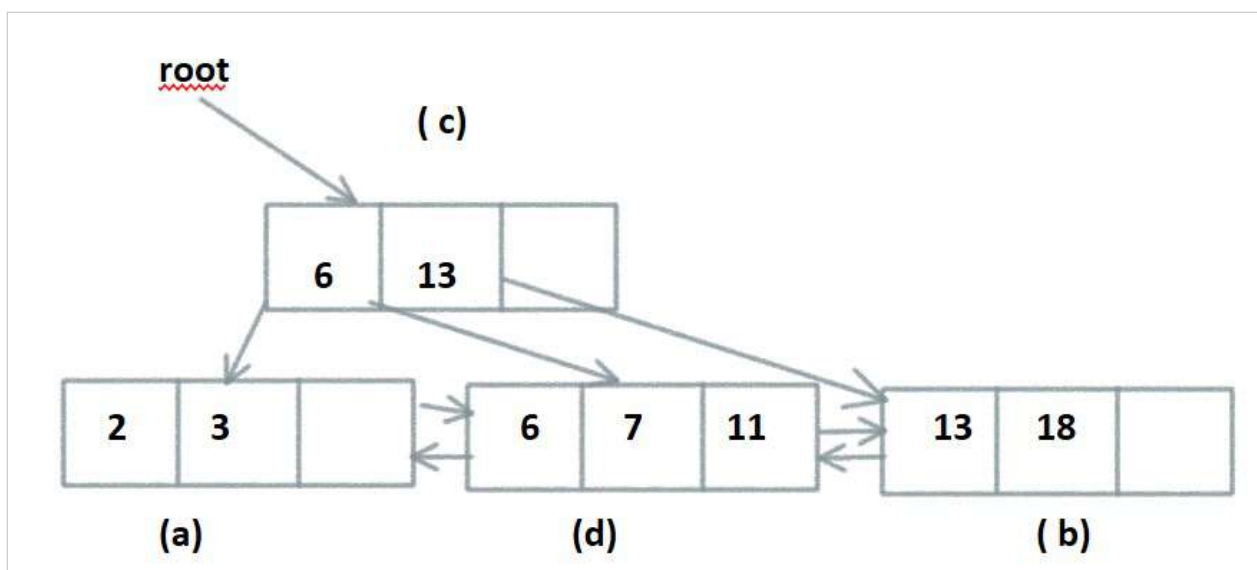
SELECT mykey FROM Tab WHERE mykey=7;

Velg ett alternativ:

- ☐ c, b, d
- ☐ c, a, d
- ☐ Ingen av de andre alternativene stemmer
- ☐ c, d
- ☐ c, b
- ☐ a, d

Maks poeng: 2

## 13 Task 13 - Access paths (2 %)



Tabellen heter 'Tab' og søkenøkkelen heter 'mykey'. Den er lagret i et clustered B+-tre som vist i figuren.

Hvilken sekvens av blokker aksesseres ved følgende query?

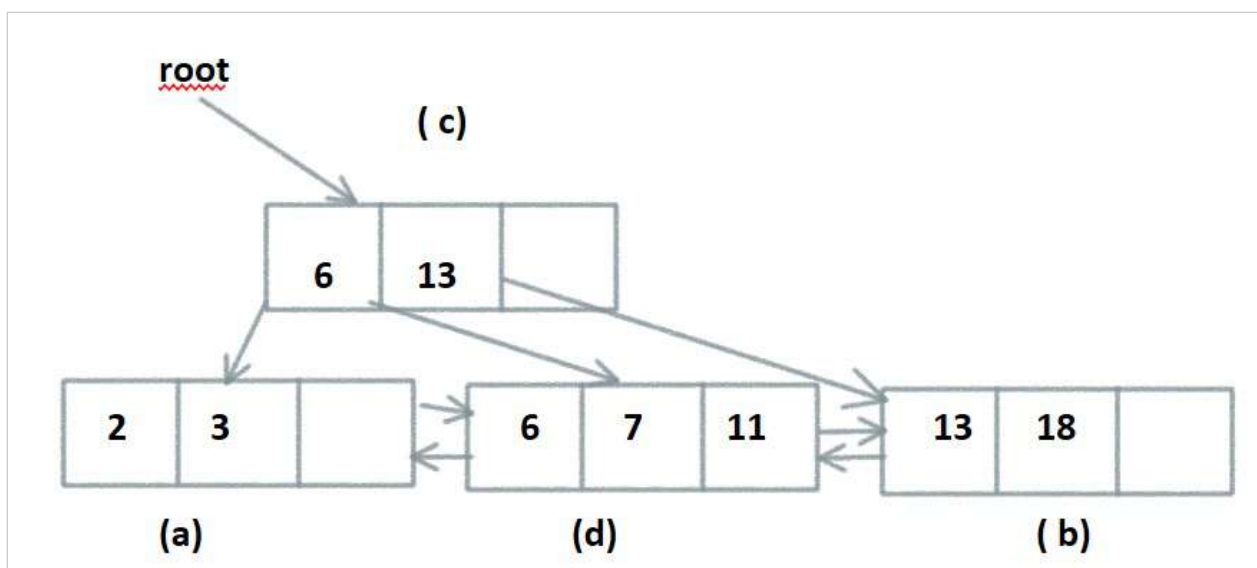
SELECT mykey FROM Tab WHERE mykey < 7;

Velg ett alternativ:

- ☐ a,d
- ☐ c, a, d
- ☐ c, d,a
- ☐ Ingen av de andre alternativene stemmer
- ☐ a, d, b
- ☐ c, b, d, a

Maks poeng: 2

# 14 Task 14 - Access paths (2 %)



Tabellen heter 'Tab' og søkenøkkelen heter 'mykey'. Den er lagret i et clustered B+-tre som vist i figuren.

Hvilken sekvens av blokker aksesseres ved følgende query?

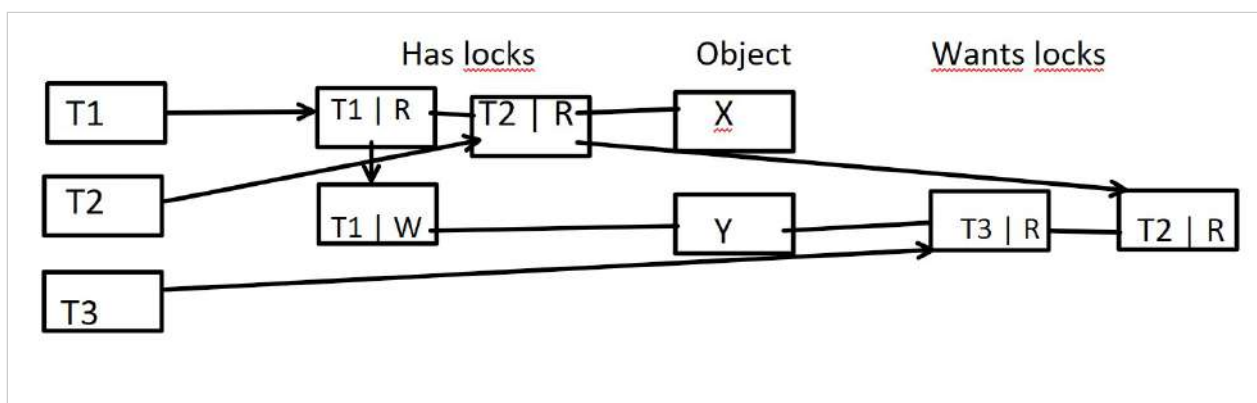
SELECT mykey FROM Tab ORDER BY mykey ASC;.

**Velg ett alternativ:**

- ☐ b, d, a
- ☐ c, d, a, b
- ☐ a, d, b
- ☐ Ingen av de andre alternativene stemmer
- ☐ c, b, d, a
- ☐ c, a, d, b

Maks poeng: 2

# 15 Task 15 - Locking 2PL (7 %)



Vi har følgende låser (i figuren) etter en sekvens av operasjoner mot databasen.

Hvilken sekvens av operasjoner vil gi denne tilstanden til låsesystemet (tofaselåsing – rigorous)?

«T1 | R» betyr leselås fra T1 og «T1 | W» betyr skrivelås fra T1.

**Velg ett alternativ:**

- ☐ w1(X); r2(X); w1(Y); r3(Y); r2(Y);
- ☐ r1(X); r2(X); w1(Y); r3(Y); r2(Y);
- ☐ r1(X); r3(X); w1(Y); r3(Y); r2(Y);
- ☐ Ingen av de andre alternativene stemmer.
- ☐ r1(X); r2(X); w1(Y); r2(Y); w3(Y);

Maks poeng: 7



**16 Task 16 - Locking 2PL (7 %)**

Vi skal sette tofaselåser (2PL) ved hjelp av metoden rigorous. I hvilken rekkefølge committer de tre transaksjonene T1, T2 og T3?

Følgende sekvens av operasjoner kommer inn til databasen:

r1(X); r2(X); w1(X); r3(Y); c1; w2(X); w2(Y); c2; r3(Z); c3;

**Velg ett alternativ:**

- ☐ T3; T2; T1;
- ☐ T3; T1; T2;
- ☐ T1; T2; T3;
- ☐ T3; Vi får vranglås mellom T1 og T2.
- ☐ T2; T3; T1;
- ☐ T2; T1; T3;
- ☐ T1; T3; T2;
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 7

**17 Task 17 - Recovery REDO (7 %)**

Vi har en logg som ser ut som følger etter en krasj.

LSN	PrevLSN	TransactionID	Operation	PageID
101	0	T1	Update	A
102	101	T1	Update	B
103			Ckpt_start	
104			Ckpt_end	
105	0	T2	Update	A
106	105	T2	Commit	
107	102	T1	Update	B
108	0	T3	Update	C

Og en DPT som ser ut som følger i loggpost 104 (Ckpt\_end):

(Page B, RecLSN=102)

Blokkene har følgende verdier for PageLSN:

A har PageLSN=101

B har PageLSN=102

C har PageLSN=98

Hvilke loggposter blir det gjort REDO på (LSN er oppgitt i alternativene)?

**Velg ett alternativ:**

- ☐ 108
- ☐ 105, 107, 108
- ☐ 107, 108
- ☐ 101, 102, 105, 107, 108
- ☐ 102, 105, 107, 108
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 7

## 18 Task 18 - Recovery UNDO (6 %)

Vi har en logg som ser ut som følger etter en krasj. Hvilke loggposter vil genereres under undo?

LSN	PrevLSN	TransactionID	Operation	PageID
101	0	T1	Update	A
102	101	T1	Update	B
103			Ckpt_start	
104			Ckpt_end	
105	0	T2	Update	A
106	105	T2	Commit	
107	102	T1	Update	B
108	0	T3	Update	C

Formatet på loggpostene under er (LSN, PrevLSN, TransactionID, Operation, PageID)

**Velg ett alternativ:**

- ☐ (109, 108, T3, CLR, C), (110, 109, T3, Abort,), (111, 107, T1, CLR, B), (112, 111, T1, CLR, B), (113, 112, T1, CLR, A), (114, 113, T1, Abort,)
- ☐ Ingen loggposter genereres
- ☐ (109, 106, T2, CLR, A), (110, 109, T2, Abort,)
- ☐ (109, 108, T3, CLR, C), (110, 109, T3, Abort,)
- ☐ (109, 107, T1, CLR, A), (110, 109, T1, CLR, B), (111, 110, CLR, A), (112, 111, T1, Abort,)
- ☐ Ingen av de andre alternativene stemmer

---

Maks poeng: 6

## 19 Task 19 - Comments (0 %)

Denne "oppgaven" er en mulighet for å informere om *omstendigheter* som du tenker er **helt nødvendige** å kommunisere til sensor, for at din besvarelse skal bli riktig vurdert. Dette kan for eksempel gjelde antakelser som det var tvingende nødvendig å gjøre.

Du skal *ikke* bruke dette feltet til å gi *generelle kommentarer* til eksamen, det kan gjøres i Piazza eller i e-post til faglærer.

Skriv ditt svar her

Format

**B**


*I*


U


$x_e$


$x^e$


$I_x$
























$\Sigma$



Words: 0

Maks poeng: 0

20/20