

Sensorveiledning TDT4145 eksamen 11. juni 2021

Versjon 2. mai 2022

Læringsutbyttebeskrivelser for TDT4145

Kunnskaper:

1. Databasesystemer: generelle egenskaper og systemstruktur.
2. Datamodellering med vekt på entity-relationship-modeller.
3. Relasjonsdatabasemodellen for databasesystemer, databaseskjema og dataintegritet.
4. Spørrespråk: Relasjonsalgebra og SQL.
5. Designteori for relasjonsdatabaser.
6. Systemdesign og programmering mot databasesystemer.
7. Datalagring, filorganisering og indeksstrukturer.
8. Utføring av databasespørringer.
9. Transaksjoner, samtidighet og robusthet mot feil.

Ferdigheter:

1. Datamodellering med entity-relationship-modellen.
2. Realisering av relasjonsdatabaser.
3. Databaseorientert programmering: SQL, relasjonsalgebra og database-programmering i Java.
4. Vurdering og forbedring av relasjonsdatabaseskjema med utgangspunkt i normaliseringsteori.
5. Analyse og optimalisering av ytelsen til databasesystemer.

Generell kompetanse:

1. Kjennskap til anvendelser av databasesystemer og forståelse for nytte og begrensninger ved slike systemer.
2. Modellering av og analytisk tilnærming til datatekniske problemer

Poenggrenser brukt:

Terskelverdier:

A: 88
B: 76
C: 64
D: 52
E: 37
F: 0

Dette er de endelige poenggrensene brukt ved sensur.

1. Miscellaneous (3 %)

Hvilken påstand er riktig?

Velg ett alternativ:

- ☐ En historie kan umulig havne i vranglås med "basic" tofaselåsing.
- ☐ En historie kan umulig havne i vranglås med "konservativ" tofaselåsing.
- ☐ En historie kan umulig havne i vranglås med "strict" tofaselåsing.
- ☐ En historie kan umulig havne i vranglås med "rigorous" tofaselåsing.

Konservativ 2PL.

2. Miscellaneous (3 %)

Hvilket alternativ er riktig?

Velg ett alternativ:

- ☐ Konfliktserialiserbarhet impliserer tofaselåsing
- ☐ Serialiserbarhet kan bare holdes ved låsing
- ☐ Tofaselåsing impliserer konfliktserialiserbarhet
- ☐ Serialiserbarhet krever serielle historier

Tofaselåsing impliserer konfliktserialiserbarhet.

3. Miscellaneous (3 %)

Hva er recovery-egenskapen til følgende historie: $w_1(A); r_2(A); w_1(B); w_2(B); c_1; c_2;$

Velg ett alternativ:

- ☐ Strict
- ☐ Recoverable (gjenopprettbar)
- ☐ Unrecoverable (ikke gjenopprettbar)
- ☐ ACA (unngå galopperende abort)

Recoverable.

4. Miscellaneous (3 %)

Hva er recovery-egenskapen til følgende historie: w1(A); w1(B); w2(A); c1; r2(B); r3(B); c2; c3;

Velg ett alternativ:

- ☐ Unrecoverable (ikke gjenopprettbar)
- ☐ Recoverable (gjenopprettbar)
- ☐ ACA (unngå galopperende abort)
- ☐ Strict

ACA.

5. Miscellaneous (3 %)

Hva er recovery-egenskapen til følgende historie: r1(A); w2(A); w1(A); r3(A); c1; c2; c3;

Velg ett alternativ:

- ☐ Unrecoverable (ikke gjenopprettbar)
- ☐ Recoverable (gjenopprettbar)
- ☐ ACA (unngå galopperende abort)
- ☐ Strict

Recoverable.

6. Miscellaneous (3 %)

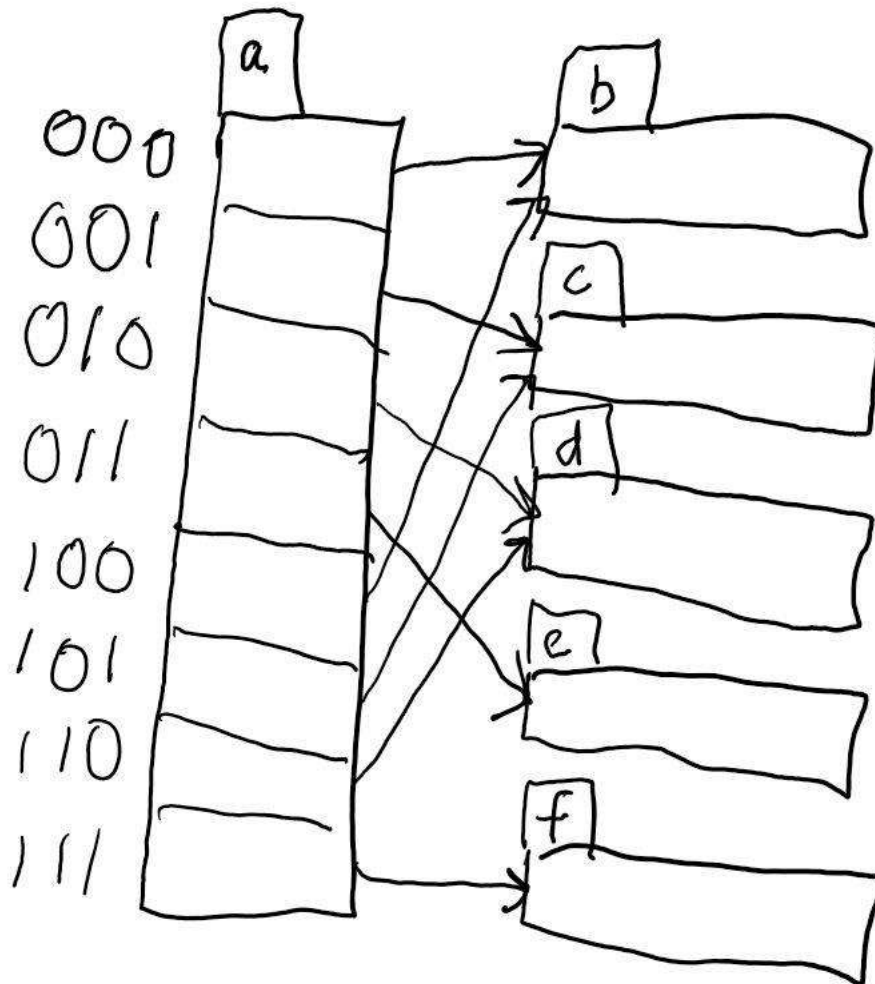
Hva er fordelen ved å bruke MVCC (multiversjon samtidighetskontroll)?

Velg ett alternativ:

- ☐ Man forhindrer vranglåser
- ☐ Det tillater at mange skriver samme dataelement samtidig
- ☐ Det tillater at transaksjoner kan lese endringene til andre transaksjonene før de andre commiter.
- ☐ Det tillater transaksjoner å lese dataelementer som andre transaksjoner skriver til samtidig.

Det tillater transaksjoner å lese data som andre transaksjoner skriver til samtidig.

For oppgave 7-9 var oppgaven å finne hvilket tall som skulle stå der a) d) og f) står i figuren. Randomisert mellom studentene.

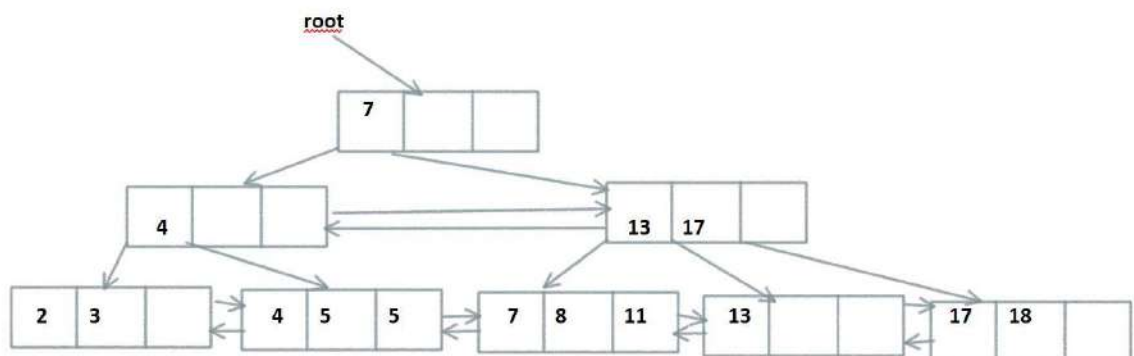


7. Extendible hashing (4 %)
Global dybde (a) = 3.
8. Extendible hashing (4 %)
Lokal dybde (d) = 2.
9. Extendible hashing (4 %)
Lokal dybde (f) = 3.

For oppgave 10-13 stod det i oppgaven de fire sekvensene med nøkler som skulle settes inn, og en figur. Så skulle studentene finne ut hvilken sekvens som ga B+-treet illustrert i oppgaven. Randomisert mellom studentene hvilken figur de fikk.

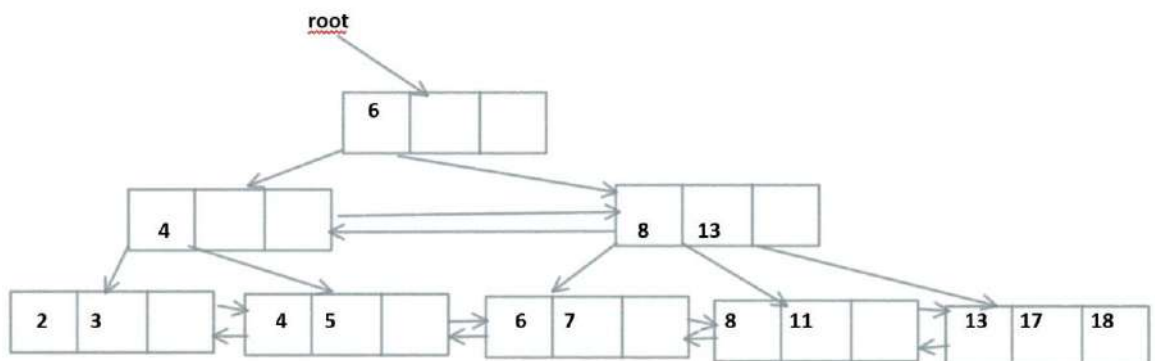
10. B+-tree (10 %)

2, 4, 7, 3, 5 17, 13, 11, 18, 6, 8 . Her var det en feil figuren (den siste 5 på løv nivå skal være 6). Opplyst med melding til alle under eksamen vha. Inspira. Svært mange studenter oppdaget feilen og ringte faglærer under eksamen. Et B+-tre skal ha unike søkenøkler.



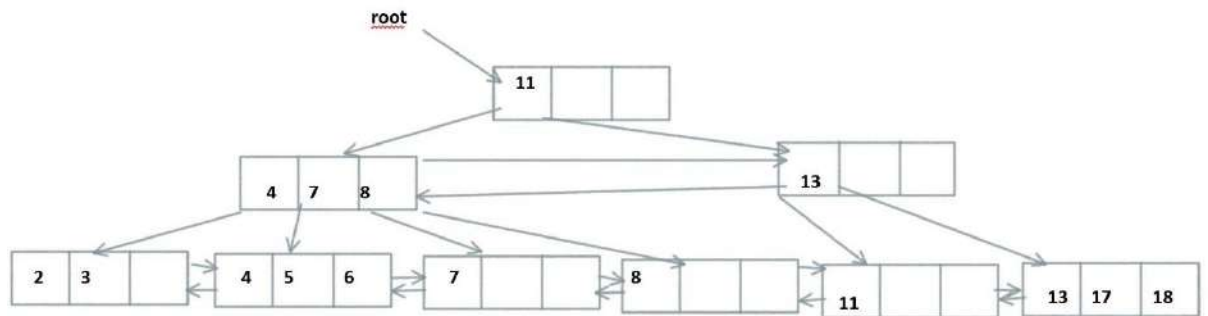
11. B+-tree (10 %)

2, 3, 4, 5, 6, 7, 8, 11, 13, 17, 18



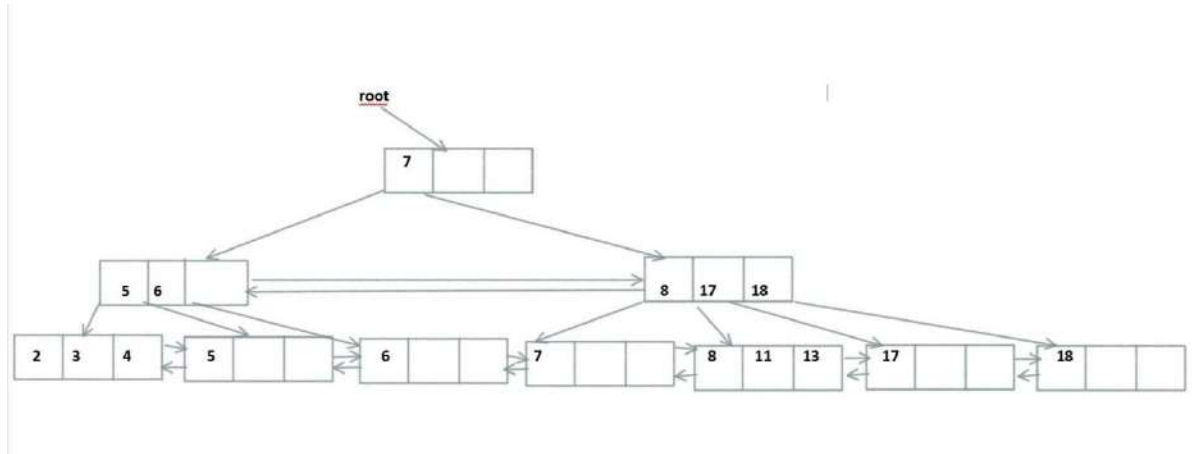
12. B+-tree (10 %)

11, 13, 2, 3, 17, 18, 4, 5, 8, 7, 6



13. B+-tree (10 %)

7, 8, 6, 4, 5, 17, 18, 2, 3, 11, 13



14. Recovery (ARIES) (6 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	<u>PrevLSN</u>	<u>TransId</u>	Type	<u>PageId</u>
101	100	T1	Update	C
102	null	T2	Update	B
103			<u>begin ckpt</u>	
104			<u>end ckpt</u>	
105	101	T1	<u>Commit</u>	
106	102	T2	Update	C
107	null	T3	Update	A
108	107	T3	Update	C
109	106	T2	<u>Commit</u>	

Anta end ckpt med LSN 104 har en Dirty Page Table (DPT) som inneholder (B, 102). Hvilke elementer hører med i DPT når analysen er fullført?

Velg ett eller flere alternativer

- ☐ (C, 101)
- ☐ (B, 102)
- ☐ (T1, Commit)
- ☐ (C, 106)
- ☐ (A, 107)
- ☐ (C, 108)
- ☐ (T2, Commit)

(B, 102), (C, 106) og (A, 107)

15. Recovery (ARIES) (6 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	PrevLSN	TransId	Type	Pageld
101			begin_ckpt	
102	null	T2	Update	B
103			end_ckpt	
104	99	T1	Update	B
105	104	T1	Commit	
106	102	T2	Update	C
107	null	T3	Update	A
108	107	T3	Update	C
109	106	T2	Commit	

Anta end_ckpt med LSN 103 har en Dirty Page Table (DPT) som inneholder (B, 102). Hvilke elementer hører med i DPT når analysen er fullført?

Velg ett eller flere alternativer

- ☐ (B, 102)
- ☐ (B, 104)
- ☐ (Commit, 105)
- ☐ (C, 106)
- ☐ (A, 107)
- ☐ (C, 108)
- ☐ (Commit ,109)

(B, 102), (C, 106) og (A, 107)

16. Recovery (ARIES) (6 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	PrevLSN	TransId	Type	Pageld
101	100	T1	Update	C
102	null	T2	Update	B
103			begin_ckpt	
104			end_ckpt	
105	101	T1	Commit	
106	102	T2	Update	C
107	null	T3	Update	A
108	107	T3	Update	B
109	106	T2	Commit	

Anta end_ckpt med LSN 104 har en Dirty Page Table (DPT) som inneholder (A, 97). Hvilke elementer hører med i DPT når analysen er fullført?

Velg ett eller flere alternativer

- ☐ (A, 97)
- ☐ (C, 101)
- ☐ (B, 102)
- ☐ (Commit, 105)
- ☐ (C, 106)
- ☐ (A, 107)
- ☐ (B, 108)

(A, 97), (C, 106) og (B, 108)

17. Access methods (3 %)

Anta vi har en tabell Birdwatcher(bwid, name, address, age, ...) med 10000 rader. Attributtet bwid er primærnøkkel for tabellen. Tabellen er lagret i et clustered B+-tre med bwid som søkenøkkel, der løvnodene (level=0) i B+-treet har 750 blokker. Det er tre nivåer med blokker i B+-treet. Hvor mange blokkaksesser får vi med SQL-setningen "SELECT * FROM Birdwatcher where bwid=2001;"?

Velg ett alternativ:

- ☐ 10 000
- ☐ 750
- ☐ 3
- ☐ 1,5

3.

18. Access methods (3 %)

Anta vi har følgende nøkler som skal settes inn i en statisk hash-struktur:

10, 11, 12, 13, 14, 15, 16, 17 og 18.

Hash-strukturen er tom fra før og har 4 blokker der hver

blokk kan inneholde to poster. Vi bruker separat lenket overløp hvis en blokk blir full.

Bruk hashfunksjonen $h(k) = k \bmod 4$ til å sette inn nøklene i rekkefølgen gitt over.

Hvor mange blokkaksesser får vi gjennomsnittlig her på aksess med en tilfeldig nøkkel over? Rund av svaret til to desimaler.

Velg ett alternativ:

☐ 1.01

☐ 1.11

☐ 1.22

☐ 1.33

19. Access methods (3 %)

Anta vi har en tabell Birdwatcher(bwid, ...) med 10000 rader. Attributtet bwid er primærnøkkel for tabellen. Tabellen er lagret i en clustered, statisk hash-struktur med bwid som søkenøkkel, der alle fugletitterne får plass i 500 blokker. Gjennomsnittlig aksesseres 1.25 blokker per søk på bwid. Hvor mange blokkaksesser får vi med SQL-setningen "SELECT * FROM Birdwatcher;"?

Velg ett alternativ:

- ☐ 10 000
- ☐ 12 500
- ☐ 500
- ☐ 625

500.

20. 2PL Execution (5 %)

Databasesystemet får inn følgende sekvens av operasjoner. Vi innfører tofaselåsing (rigorous). I hvilken rekkefølge committer transaksjonene?

r2(A); r1(A); r3(A); w2(B); r1(B); w1(C); r2(C); c3; c2; c1;

Hvis flere transaksjoner blir vekt opp etter en låsventing, blir de vekt opp i den rekkefølgen de la seg til å vente, dvs. en kø.

Velg ett alternativ:

- ☐ T1; T2; T3;
- ☐ T1; T3; T2;
- ☐ Vranglås mellom T1 og T2.
- ☐ T2; T1; T3;
- ☐ T2; T3; T1;
- ☐ T3; T2; T1;
- ☐ T3; T1; T2;

T3; T2; T1;

21. 2PL Execution (5 %)

Databasesystemet får inn følgende sekvens av operasjoner. Vi innfører tofaselåsing (rigorous). I hvilken rekkefølge committer transaksjonene?

r1(A); r2(A); w1(A); r2(B); r3(B); w3(B); c1, c2; c3;

Hvis flere transaksjoner blir vekt opp etter en låsventing, blir de vekt opp i den rekkefølgen de la seg til å vente, dvs. en kø.

Velg ett alternativ:

- ☐ T1; T2; T3
- ☐ T1; T3; T2;
- ☐ T2; T1; T3;
- ☐ T3; T2; T1;
- ☐ T2; T3; T1;
- ☐ Vranglås mellom T1, T2 og T3.
- ☐ T3; T1; T2;

T2; T1; T3;

(noen studenter ble forvirret av komma mellom c1 og c2, men det skal være semikolon).

22. 2PL Execution (5 %)

Databasesystemet får inn følgende sekvens av operasjoner. Vi innfører tofaselåsing (rigorous). I hvilken rekkefølge committer transaksjonene?

r1(A); r2(A); r3(A); w3(B); r2(B); w3(C); r1(C); c2; c3; c1;

Hvis flere transaksjoner blir vekt opp etter en låsventing, blir de vekt opp i den rekkefølgen de la seg til å vente, dvs. en kø.

Velg ett alternativ:

- ☐ T1; T2; T3;
- ☐ T1; T3; T1;
- ☐ T2; T1; T3;
- ☐ T2; T3; T1;
- ☐ T3; T1; T2;
- ☐ Vranglås mellom T3, T2 og T1.
- ☐ T3; T2; T1;

T3; T2; T1;

23. Join (5 %)

Vi har følgende to tabeller:

Employee(ssn, first_name, last_name, bdate, sex, salary, super_ssn, dno)

Department (dname,dnumber,mgr_ssn,mgr_start_date)

Anta Department har 1000 poster lagret i 12 diskblokker og Employee har 120 000 poster i 4000 diskblokker.

Gitt det følgende queryet:

```
SELECT e.last_name, e.first_name, d.dname  
FROM Department d, Employee e  
WHERE e.dno=d.dnumber;
```

Anta du har plass for 8 diskblokker tilgjengelig i buffer. Hvor mange I/Oer (reads) får du ved å bruke nested-loop-join her for joinen `e.dno=d.number`?

Velg ett alternativ:

- ☐ 11988
- ☐ 8012
- ☐ 4006
- ☐ 12012
- ☐ 4012

8012(minste tabell har alltid kun *ett* gjennomløp)

24. Join (5 %)

Vi har følgende to tabeller

Employee (ssn, first_name, last_name, bdate, sex, salary, super_ssn, dno)

Department (dname, dnumber, mgr_ssn, mgr_start_date)

Anta Department har 800 poster lagret i 16 diskblokker og Employee har 800000 poster i 3000 diskblokker.

Gitt det følgende queryet:

```
SELECT e.last_name, e.first_name, d.dname  
FROM Department d, Employee e  
WHERE e.dno=d.dnumber;
```

Anta du har plass for 18 diskblokker tilgjengelig i buffer. Hvor mange I/Oer (reads) får du ved å bruke en nested-loop-join her for joinen `e.dno=d.number`?

Velg ett alternativ:

- ☐ 6016
- ☐ 12032
- ☐ 3016
- ☐ 6032
- ☐ 9016

3016 (minste tabell har alltid kun *ett* gjennomløp)

25. Join (5 %)

Vi har følgende to tabeller

Employee (ssn, first_name, last_name, bdate, sex, salary, super_ssn, dno)

Department (dname, dnumber, mgr_ssn, mgr_start_date)

Anta Department har 500 poster lagret i 12 diskblokker og Employee har 70000 poster i 2500 diskblokker.

Gitt det følgende queryet:

```
SELECT e.last_name, e.first_name, d.dname  
FROM Department d, Employee e  
WHERE e.dno=d.dnumber;
```

Anta du har plass for 12 diskblokker tilgjengelig i buffer. Hvor mange I/Oer (reads) får du ved å bruke en nested-loop-join her for joinen e.dno=d.number?

Velg ett alternativ:

- ☐ 2512
- ☐ 5012
- ☐ 2506
- ☐ 7512
- ☐ 75 000

5012. (minste tabell har alltid kun *ett* gjennomløp)

26. Miscellaneous (6 %)

Hvilke påstander er sanne?

Velg ett eller flere alternativer

- ☐ Heapfiler er gode for å søke etter poster med ett gitt mobilnr. som søkenøkkel.
- ☒ Heapfiler er gode til å skrive data som ikke trenger en spesiell sortering.
- ☒ Statisk hashing er gode for å søke etter poster gitt mobilnr. som søkenøkkel.
- ☐ B+-trær er den raskeste metoden for å søke etter data.
- ☐ Unclustered B+-trær er nyttige hvis det du søker etter utgjør en stor del av datamengden.
- ☒ Clustered B+-trær er fine hvis du trenger sorterte datamengder.

Blått kryss er rett svar.

27. Miscellaneous

Hvilke utsagn er sanne?

Velg ett eller flere alternativer

- ☐ Vranglåser unngås ved å innføre MVCC.
- ☒ Vranglåser kan løses ved å gi transaksjoner en timeout, før de automatisk aborteres.
- ☐ Overflyt i statisk hashing gjør ytelsen kjempebra
- ☒ En hashfunksjon som gir god spredning er viktig i hashing.
- ☒ I flettesortering (merge-sort) er det gunstig at tabellene er sortert på forhånd
- ☐ I flettesortering (merge-sort) er antall flettepass (merge passes) uavhengig av størrelsen på buffer tilgjengelig.

Blått kryss er rett svar.

28. Miscellaneous (6 %)

Hvilke utsagn er sanne?

Velg ett eller flere alternativer

- ☐ SQL dictionary / Catalog brukes for å sjekke reserverte ord i SQL queries.
- ☒ Optimalisatoren velger mellom forskjellige utføringsplaner for et query basert på en kostnadsmodell.
- ☒ SERIALIZABLE gjør at transaksjoner ikke merker noe til såkalte "fantomer".
- ☐ READ COMMITTED gjør at alle samtidighetsproblemer blir borte.
- ☐ Write-Ahead Logging sørger for at data må skrives før loggen skrives.
- ☒ Force / No-Steal gjør at man ikke trenger logging

Blått kryss er rett svar. «ikke merker noe til» kan også leses som «unngår».

29. Miscellaneous (3 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	PrevLSN	TransId	Type	PageId
99	null	T1	Update	A
100	99	T1	Update	B
101	100	T1	Update	C
102	null	T2	Update	B
103			begin_ckpt	
104			end_ckpt	
105	101	T1	Commit	
106	102	T2	Update	B
107	null	T3	Update	A
108	107	T3	Update	D
109	106	T2	Commit	

Anta end_ckpt med LSN 104 har en Dirty Page Table (DPT) som inneholder (B, 102), (A, 99) og (C, 101). Ved hvilken loggpost (LSN) må REDO-scannet starte?

Velg ett alternativ:

- ☐ 99
- ☐ 100
- ☐ 101
- ☐ 105
- ☐ 106
- ☐ 107
- ☐ 108
- ☐ 109

LSN =99. Eldste RecLSN i DPT.

30. Miscellaneous (3 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	PrevLSN	TransId	Type	Pageld
99	null	T1	Update	A
100	99	T1	Update	B
101	null	T3	Update	C
102	null	T2	Update	B
103			begin_ckpt	
104			end_ckpt	
105	100	T1	Commit	
106	null	T4	Update	B
107	101	T3	Update	A
108	107	T3	Update	D
109	102	T2	Commit	

Transaksjonstabellen i loggposten end_ckpt inneholder transaksjonstabellen med aktive transaksjoner (T1, T2 og T3). Hvor mange Update-loggposter vil bli undoet i løpet av recovery?

Velg ett alternativ:

- ☐ 0
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7

4 loggposter.

31. Miscellaneous (3 %)

Følgende logg ble funnet etter et krasj med ARIES-type logging og recovery:

LSN	PrevLSN	TransId	Type	PageId
101	100	T1	Update	C
102	null	T2	Update	B
103			begin_ckpt	
104			end_ckpt	
105	101	T1	Commit	
106	102	T2	Update	B
107	null	T3	Update	A
108	107	T3	Update	D
109	106	T2	Commit	

Anta end_ckpt med LSN 104 har en Dirty Page Table (DPT) som inneholder (B, 102). Basert på loggen vi ser, hvilken Page er det vi med sikkerhet kan si er skrevet ut før krasjet.

Velg ett alternativ:

- ☐ A
- ☐ B
- ☐ C
- ☐ D
- ☐ Ckpt
- ☐ T1
- ☐ T2
- ☐ T3

Page C. I og med at den ikke er funnet i DPT i sjekkpunktloggposten.

32. Serializability (6 %)

Hvilke historier er konfliktserialiserbare?

Velg ett eller flere alternativer

- ☐ w1(X); r1(Y); w2(Y); r3(Y); w1(Y); r3(X); r2(X);
- ☒ w2(X); w3(Y); r3(Y); r1(X); r2(X); r1(Y); w1(Y);
- ☒ r1(X); r2(Y); r3(X); w1(X); w2(Y); w3(Z);
- ☐ r2(X); w2(X); r1(Y); w2(Y); w1(Y); r3(X); w3(Z);
- ☒ r2(X); w2(X); r1(X); w2(Y); w1(Y); r3(X); w3(Z);
- ☐ r2(Z); w2(X); r1(Y); w2(Y); r3(X); w3(Z); r1(X);

Blått kryss er rett svar.

33. Serializability (6 %)

Hvilke historier er konfliktserialiserbare?

Velg ett eller flere alternativer

- ☐ r1(X); r2(X); w2(X); r3(X); w3(Z); r1(Z);
- ☒ r2(X); r3(X); w2(X); r3(Y); w3(Y); r1(Y); w2(Y);
- ☐ r1(X); w1(X); r2(Y); w3(Y); r3(X); w3(Z); w2(X);
- ☒ r1(X); w1(X); r2(Y); w2(Y); r3(Z); w3(X);
- ☒ r2(X); r1(Y); w2(X); r3(X); w1(Y); w3(X); r2(Y); w2(Y);
- ☐ r2(X); w3(X); w1(Y); r2(Y); w3(Z); w2(Z);

Blått kryss er rett svar.

34. Serializability (6 %)

Hvilke historier er konfliktserialiserbare?

Velg ett eller flere alternativer

☒ r2(X); r1(X); w2(X); r3(X); r2(Z); w3(Z); r1(Y);

☐ r1(X); r1(Y); w1(X); w2(Y); w3(Y); w1(X); r2(Y);

☒ w2(X); w3(Y); r3(Y); r1(X); r2(X); r1(Y); w1(Y);

☐ r1(X); w1(X); r2(Y); w3(Y); r3(X); w3(Z); w2(X);

☒ r1(X); w1(X); r2(Y); w2(Y); r3(X); w3(Z); r2(Z);

☐ r3(X); w1(X); r2(Z); w2(X); r3(Z); w3(Z); r1(X);

Blått kryss er rett svar.

35. (34 %)

BestPris har informasjon om prisutviklingen for produkter, som vaskemaskiner eller mobiltelefoner. Et produkt registreres med en unik produkt-id, produktnavn og en kort spesifikasjon av produktet. BestPris har oversikt over leverandører som selger produkter som er registrert i databasen. Hver leverandør er registrert med en unik leverandør-id, leverandørnavn og en nettside til leverandørens nettsider. Alle produkter har minst en leverandør. Produktene er organisert i ulike produktkategorier. Produktkategoriene har en unik kategori-id og et kategorinavn. Kategoriene er organisert i et hierarki med to nivåer, som for eksempel kategorien fotoutstyr med underkategorier som kamerahus, objektiv og blits. Et produkt kan være i mange kategorier. For et produkt kan man registrere produsent. Produsentene har en unik produsent-id samt navn og nettsiden til produsentens nettsider. Et produkt trenger ikke å være registrert med produsent og har aldri flere enn en produsent. En produsent kan være registrert uten å være knyttet til noe produkt.

Det skal være mulig å registrere tester av et produkt, dersom det finnes tilgjengelige tester på nett. En test registreres med en unik test-id, overskrift, og nettsiden til testen. En test kan omfatte flere produkter, for eksempel flere vaskemaskiner. BestPris registrerer bare tester som gjelder ett eller flere produkter i BestPris sin database. Det er mulig å registrere seg som bruker (e-postadresse og navn) hos BestPris. Brukere får tilgang til å legge inn vurderinger av produkt. En vurdering består av en karakter (1-10) og en kort begrunnelse. En bruker kan ikke ha flere vurderinger av samme produkt, men det er mulig for brukeren å endre karakter eller begrunnelse.

BestPris holder oversikt over prisutviklingen for et produkt. For hver leverandør som selger et produkt, registreres prisen på produktet. Priser kan endres maks en gang per døgn. På den måten kan BestPris lage oversikt over gjeldende priser for et produkt og finne de laveste prisene. BestPris

kan også lage oversikter over den historiske prisutvikling til et produkt; for både laveste pris og for en særskilt leverandør.

Lag en ER-modell (du kan bruke alle virkemidler som er med i pensum) for denne databasen. Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

36. (34 %)

Helsedirektoratet ønsker å utvikle en nasjonal database for å holde oversikt over vaksinasjoner. Vaksinasjonene organiseres av kommunene (kommunenummer og kommunenavn) som har et antall vaksinasjonssenter (senter-id og navn). Et vaksinasjonssenter hører til en kommune og senter-id er unikt innenfor denne kommunen. Vaksinasjonssenter kan være alt fra en stor «vaksinasjonsfabrikk» til et helsesykepleierkontor, det omfatter alle steder der det foretas vaksinasjoner.

Det nasjonale vaksinasjonsprogrammet har et antall vaksiner som registreres i systemet. Hver vaksine har en unik vaksine-id, navn, produsent og en kort beskrivelse. Vaksiner kan bare settes av registrerte vaksinesettere. En vaksinesetter kan være tilknyttet ett eller flere vaksinasjonssenter. En vaksinesetter har bare lov til å sette de vaksinene som vedkommende er autorisert til å sette. Vaksinesettere er registrert med en unik vaksinesetter-id, navn og mobilnummer.

De som får satt vaksiner registreres med en unik person-id, navn, fødselsår, kjønn og et mobilnummer. Alle vaksinasjoner blir registrert, med informasjon om personen som fikk satt vaksinen, vaksinasjonssenter, vaksinesetter og vaksine. I tillegg registreres dato og tid. Om det oppsto avvik ved vaksinasjonen registreres dette, i tillegg til en kort beskrivelse av avviket. En vaksinasjon identifiseres med et unikt vaksinasjon-nr.

Dersom det oppstår bivirkninger etter en vaksinasjon, kan dette registreres i systemet. Det finnes en oversikt over alle leger med (unik) lege-id, navn og mobilnummer. En bivirkning registreres av en lege og gjelder en person som har fått satt en eller flere vaksiner. En registrert bivirkning har et løpenummer som er unikt for personen som har fått bivirkningen. Ved registrering av en bivirkning kan legen legge inn en eller flere vaksinasjoner som kan være årsak til bivirkningen. I tillegg registreres dato, tid og en kort beskrivelse av problemene som har oppstått. Det er ingen restriksjoner på hvor mange bivirkninger som kan registreres.

Lag en ER-modell (du kan bruke alle virkemidler som er med i pensum) for denne databasen. Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

37. (34 %)

Hotellkjeden SovGodt har hoteller som registreres med unikt navn, adresse, nettsideadresse og telefonnummer. Kjeden klassifiserer standarden på hotellene fra 1 til 5 stjerner. Hotellkjeden klassifiserer rommene som tilbys i et antall romklasser. Hver romklasse har et unikt klassenavn og en beskrivelse som definerer standarden på rommene i denne klassen. Et hotell trenger ikke å ha rom av alle romklasser.

Hvert hotell i kjeden har et antall hotellrom som leies ut til kunder. En kunde er registrert med et unikt kundennummer, navn, adresse og mobilnummer. Hotellkjeden kan ha ett eller flere fordelsprogram, en kunde kan være medlem i ett av disse. Fordelsprogram har et unikt navn og en beskrivelse. Alle hotellrom har et romnummer som er unikt for det aktuelle hotellet, kan ha et navn, har antall soveplasser og et areal i kvadratmeter. Hotellrom er registrert med romklasse, rompris og om rommet er røykfritt, handicap-vennlig, har utsikt eller er spesielt stille.

Kunder kan gjøre bestillinger på et hotell. En bestilling har en unik bestilling-id og bestillingsdato og bestillingstid. En bestilling vil bestå av en eller flere romreservasjoner, der en romreservasjon gjelder fra en ankomstdato til en avreisedato. En romreservasjon har et løpenummer som er unikt innenfor bestillingen. Det er selvsagt viktig å unngå dobbeltbooking av rom. For en romreservasjon registreres hvor mange personer som skal bo på hvert rom. Hotellkjeden er åpen for prisforhandlinger, det skal derfor registreres en totalpris for en (samlet) bestilling, denne prisen kan være rabattert i forhold til prisen på de rommene som inngår i bestillingen.

Kunder kan også legge inn vurderinger av kvaliteten på et hotell. En slik vurdering vil bestå av en tittel, en samlet vurdering fra 1 til 6 (best), vurderingsdato og en beskrivende tekst. Det er ingen begrensninger på hvor mange vurderinger en kunde kan gjøre, verken totalt eller for et bestemt hotell. Spesielt verdifulle kunder kan få VIP-status og vil få ekstraordinær god behandling ved bestilling og under opphold ved et av kjedens hoteller.

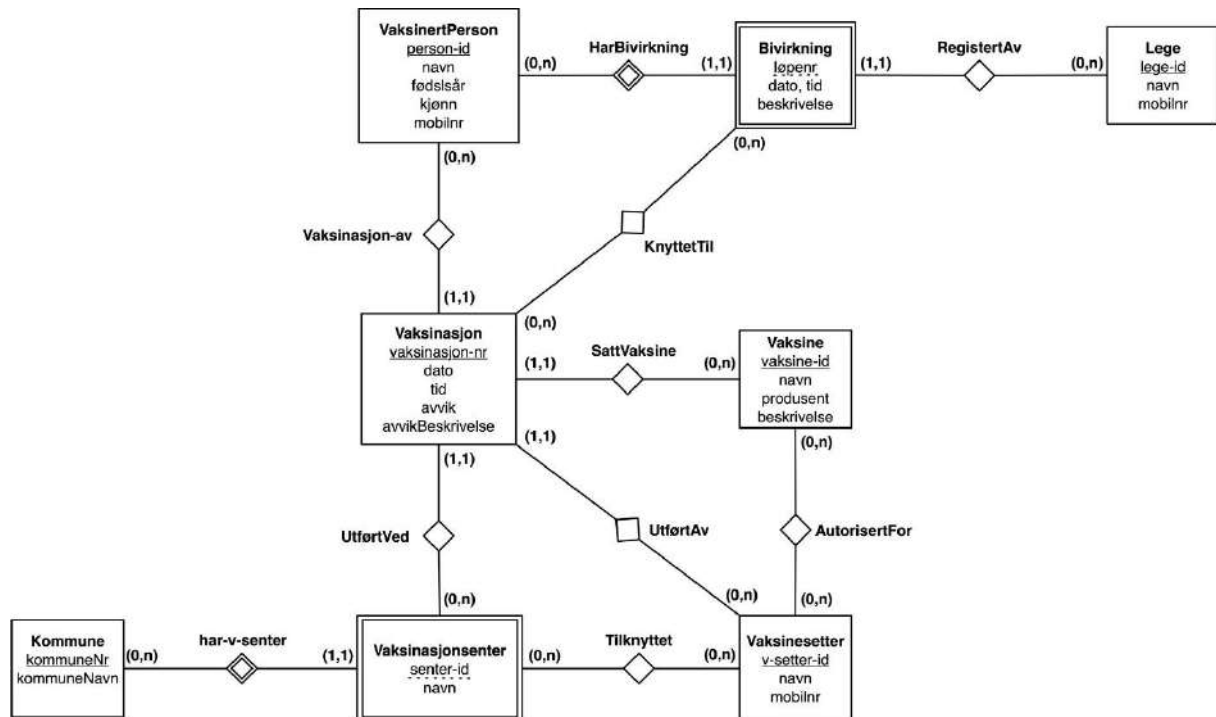
Lag en ER-modell (du kan bruke alle virkemidler som er med i pensum) for denne databasen. Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

Datamodelleringsoppgaven

Under er det vist utkast til løsning for de tre ulike modelleringsoppgavene. Det skal legges vekt på at de ulike modell-virkemidlene brukes på riktig måte. God (overordnet) «struktur» i datamodellen tillegges større vekt enn mer ubetydelige feil og mangler. Det finnes en del alternative modelleringsvalg og alternative forutsetninger som kan være like riktige som de som er vist i løsningsskissene.

Dersom det gjøres hensiktsmessige forutsetninger, skal disse legges til grunn ved vurderingen av løsningen.

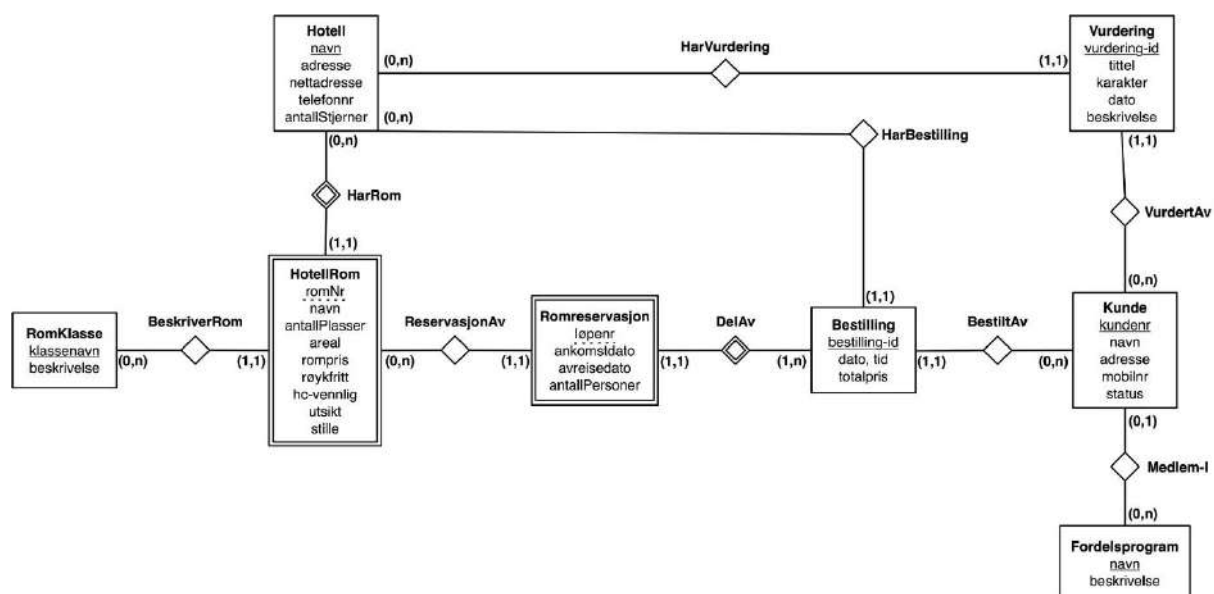
Oppgave om vaksinerings



I løsningen har vi lagt til grunn følgende presiseringer av opplysningene i oppgaveteksten:

- Valg som er nødvendige for å spesifisere min-maks-restriksjoner for relasjonsklassene

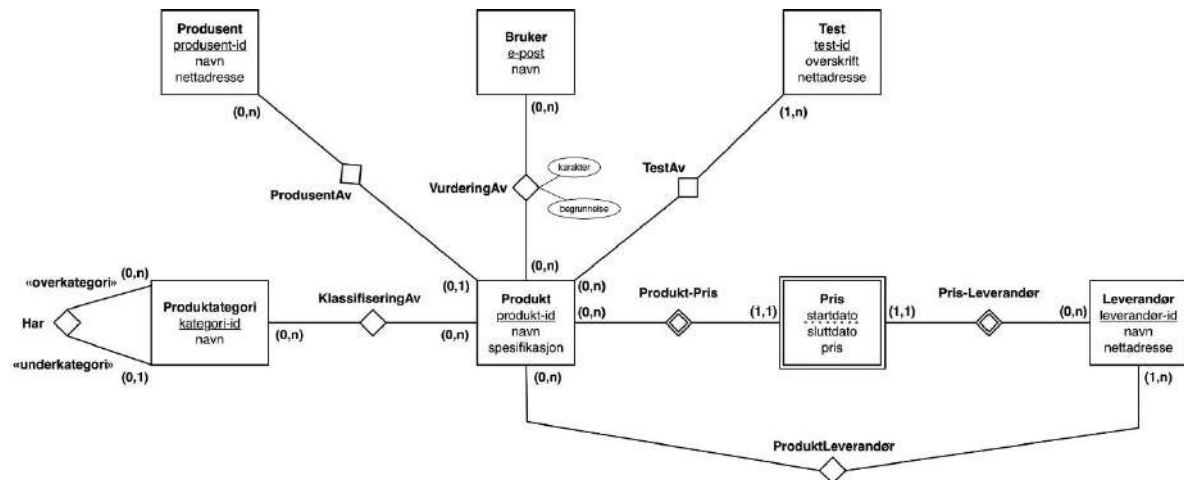
Oppgave om hotellbestillinger



I løsningen har vi lagt til grunn følgende presiseringer av opplysningene i oppgaveteksten:

- Valg som er nødvendige for å spesifisere min-maks-restriksjoner for relasjonsklassene

Oppgave om produktpriser



I løsningen har vi lagt til grunn følgende presiseringer av opplysningene i oppgaveteksten:

- Valg som er nødvendige for å spesifisere min-maks-restriksjoner for relasjonsklassene