# i Frontpage

Institutt for datateknologi og informatikk

Eksamensoppgave i TDT4145 Datamodellering og databasesystemer

Eksamensdato: 11. mai 2020

Eksamenstid (fra-til): 09.00-13.00

Hjelpemiddelkode/Tillatte hjelpemidler: A / Alle hjelpemidler tillatt

#### Faglig kontakt under eksamen:

**Tlf.:** Roger Midtstraum, mobil: 995 72 420 Svein Erik Bratsberg, mobil: 995 39 963

Teknisk hjelp under eksamen: NTNU Orakel

**Tlf**: 73 59 16 00

#### **ANNEN INFORMASJON:**

- 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.
- **Lagring**: Besvarelsen din i Inspera Assessment lagres automatisk. Jobber du i andre programmer husk å lagre underveis.
- **Juks/plagiat**: Eksamen skal være et individuelt, selvstendig arbeid. Det er tillatt å bruke hjelpemidler. Alle besvarelser blir kontrollert for plagiat.
- 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 i Inspera. Du kan finne igjen varselet ved å klikke på bjella øverst i høyre hjørne på skjermen. Det vil i tillegg bli sendt SMS til alle kandidater for å sikre at ingen går glipp av viktig informasjon. Ha mobiltelefonen din tilgjengelig.
- Vekting av oppgavene: Vektingen av oppgavene i prosent er oppgitt i overskriften.

#### **OM LEVERING:**

- Besvarelsen din leveres automatisk når eksamenstida er ute og prøven stenger, forutsatt at minst én oppgave er besvart. Dette skjer selv om du ikke har klikket «Lever og gå tilbake til Dashboard» på siste side i oppgavesettet. Du kan gjenåpne og redigere besvarelsen din så lenge prøven er åpen. Dersom ingen oppgaver er besvart ved prøveslutt, blir ikke besvarelsen din levert.
- Trekk fra eksamen: Ønsker du å levere blankt/trekke deg, gå til hamburgermenyen i øvre høyre hjørne og velg «Lever blankt». Dette kan ikke angres selv om prøven fremdeles er åpen.
- Tilgang til besvarelse: Du finner besvarelsen din i Arkiv etter at sluttida for eksamen er passert.

# <sup>1</sup> Problem 1 (15 %)

#### Situasjonsbeskrivelse:

VLDB er en internasjonal konferanse om databasesystemer som skal arrangeres i Trondheim i desember. For å administrere vurderingen av de vitenskapelige arbeidene (heretter kalt «paper») som sendes inn til konferansen, ønsker man å ta i bruk et egnet databasesystem.

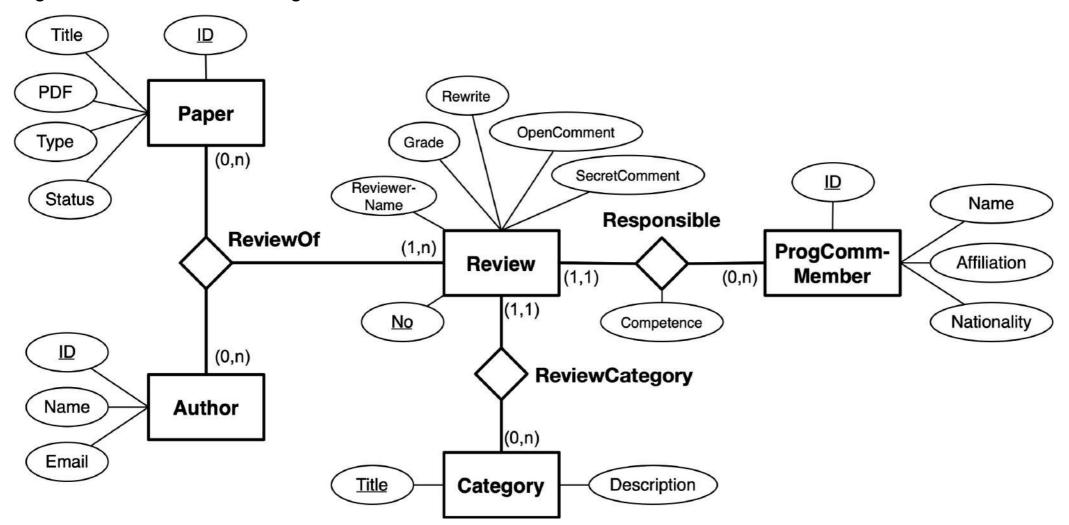
I god tid før konferansen sender man ut Call for Papers der fagmiljøet inviteres til å sende inn paper av ulike typer, f.eks. «ordinært paper», «kort paper» og «plakat» (eng: poster). Hvert mottatt paper får en unik

identifikator, og man må kunne lagre tittel, forfatter(e), e-postadresse til hver av forfatterne, PDF-fil og type paper. For å vurdere kvaliteten til de ulike paper-ene, setter man sammen en programkomité med kvalifiserte forskere innen det aktuelle fagområdet. Medlemmene i programkomiteen registreres med et unikt nummer, navn, tilknytning (universitet, forskningsorganisasjon, etc.) og e-postadresse. Hvert paper blir tilordnet tre medlemmer av programkomiteen som hver sørger for en uavhengig vurdering (review) av paper-et. Et medlem av programkomiteen er ansvarlig for de paper-ene han eller hun får tildelt, men trenger ikke å gjøre selve vurderingen selv. Dersom en annen person står for selve gjennomgangen og vurderingen av et paper, må man kunne registrere navnet på denne personen.

Hvert paper blir vurdert innen kategoriene "originality", "significance", "technical quality", "relevance", "presentation" og "overall rating". For hver kategori finnes det en tekst som beskriver kategorien og definerer hva som skal vektlegges i vurderingen. Innen hver kategori får paper-et en karakter fra 1 («Strong reject») til 7 («Strong accept») fra hver av de tre reviewerne. I tillegg kan hver reviewer skrive en kommentar til forfatterne og en (hemmelig) kommentar til programkomiteen. For å kunne vurdere ulike oppfatninger opp mot hverandre, må hver reviewer oppgi sin kompetanse (høy/middels/lav) innen det emnet som paper-et handler om. For hvert paper skal hver reviewer antyde hvor mye (ingen/litt/middels/mye) omskriving som er nødvendig før paper-et eventuelt kan aksepteres. Når alle paper er vurdert og bedømt, samles programkomiteen til et møte der de velger ut de paper-ene som skal presenteres på konferansen. Disse paper-ene får status som "accepted", mens øvrige innsendte paper får status som "rejected". Melding om programkomiteen sin beslutning, samt kopi av review-resultatene, sendes forfatterne pr e-post.

#### **Datamodell:**

I figuren under er vist et forslag til datamodell for VLDB.



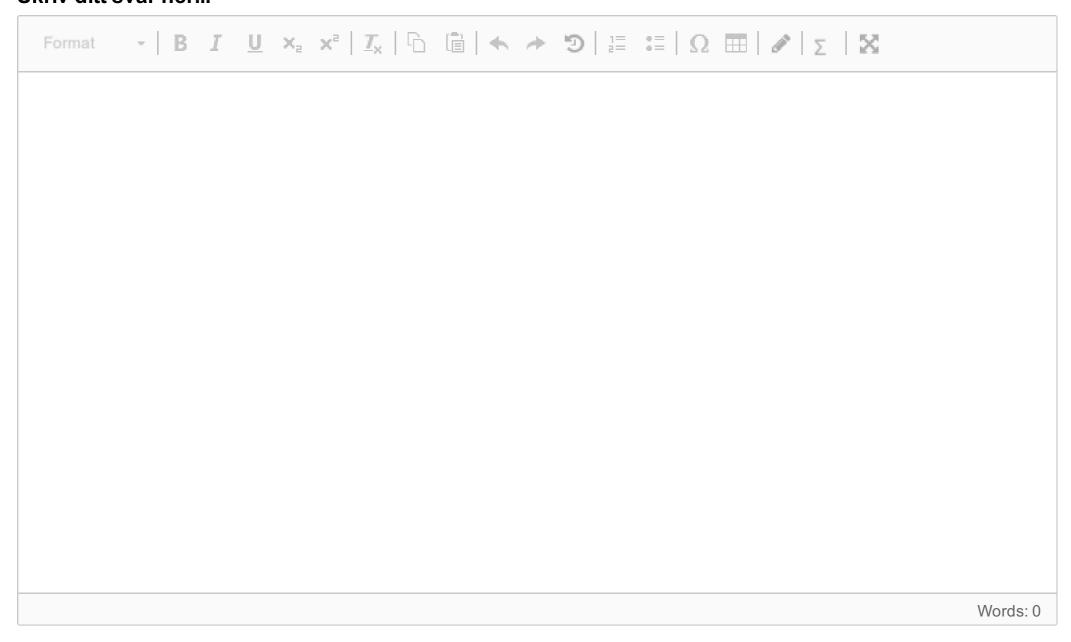
#### Oppgave:

Forslaget på datamodell har en del feil eller uheldige løsninger som gjør at modellen ikke er i overensstemmelse med situasjonsbeskrivelsen. Det er minst to slike problemer og ikke flere enn 7.

Det er din oppgave å finne disse problemene i den foreslåtte modellen. For hvert problem skal du forklare hva som er feil/uheldig og du skal beskrive hvordan problemet bør rettes opp.

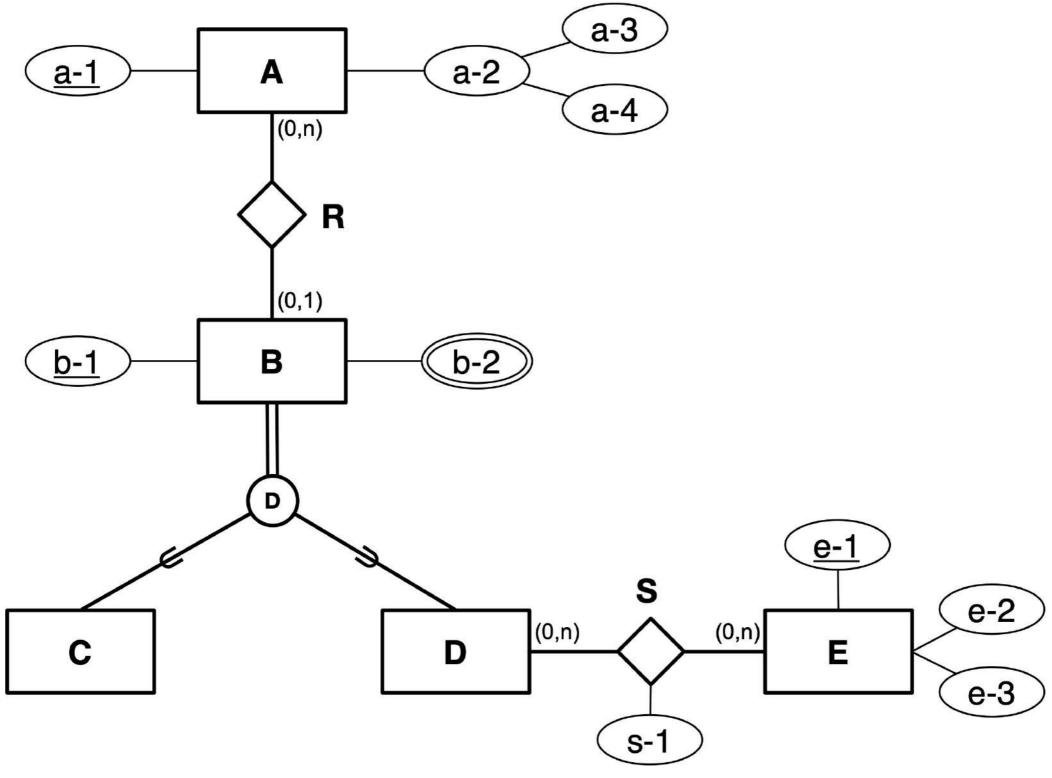
Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

## Skriv ditt svar her...



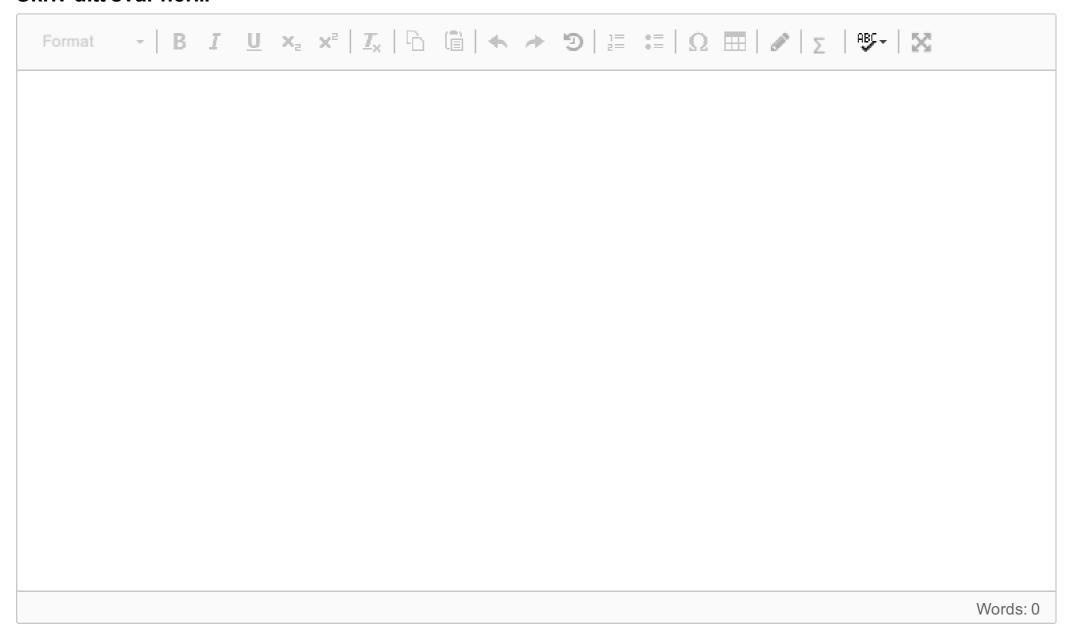
# <sup>2</sup> Problem 2 (10 %)

Ta utgangspunkt i følgende EER-modell.



Vis hvordan denne modellen kan oversettes til en tilsvarende relasjonsdatabasemodell. Du skal oppgi tabeller, attributter, primærnøkler og fremmednøkler. Diskuter alternative løsninger og forklar hvorfor du mener ditt forslag er det beste alternativet.

## Skriv ditt svar her...



Ta utgangspunkt i følgende skjema for en relasjonsdatabase som holder oversikt over Korona-pandemiens utvikling i Norge. Primærnøkler er understreket.

RegisteredDates(<u>DateSeqNo</u>, DD, MM, YY, WeekNo)

County(CountyNo, Name, Population)

Municipality(MunNo, Name, Population, CountyNo)

• CountyNo er fremmednøkkel mot County, kan ikke være NULL

#### Tested(MunNo, DateSeqNo, Number)

- Hvis det ikke er testet noen, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

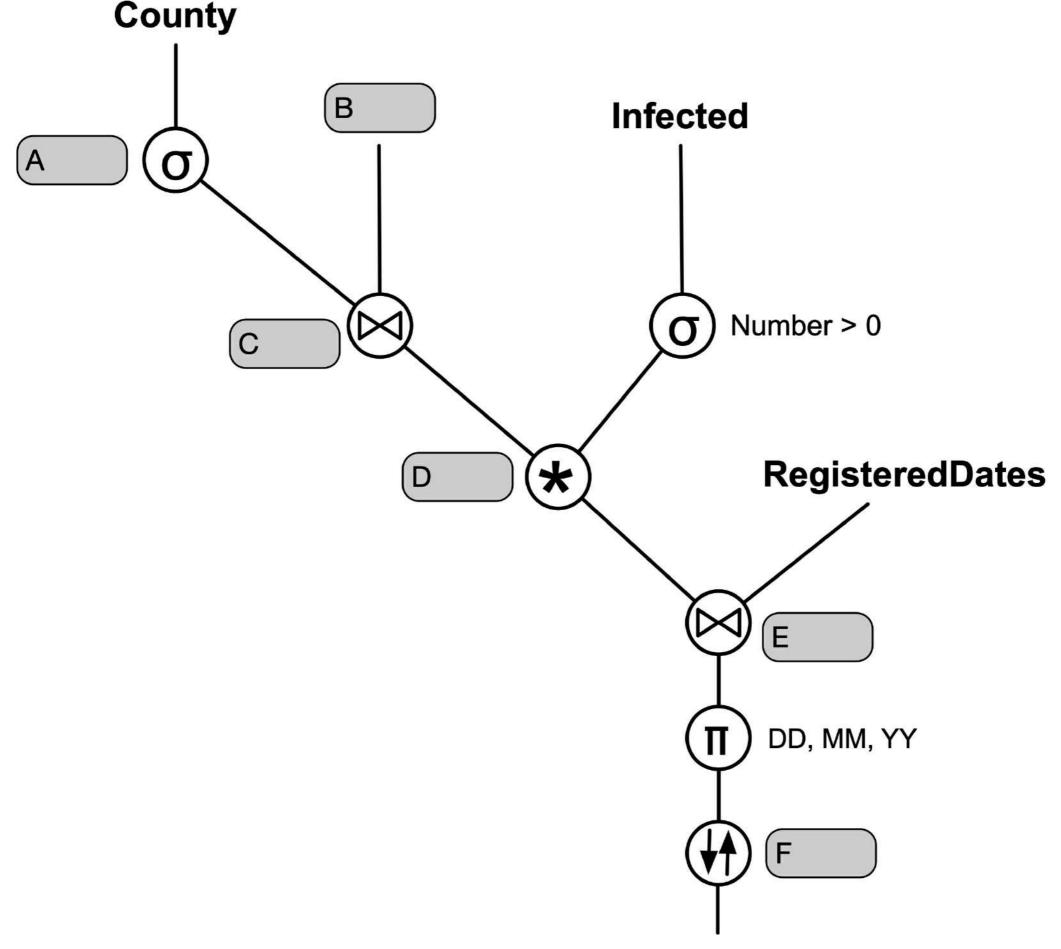
#### Infected(MunNo, DateSeqNo, Number)

- Hvis det ikke er registert ny smitte, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

### Hospitalized(MunNo, DateSeqNo, Number)

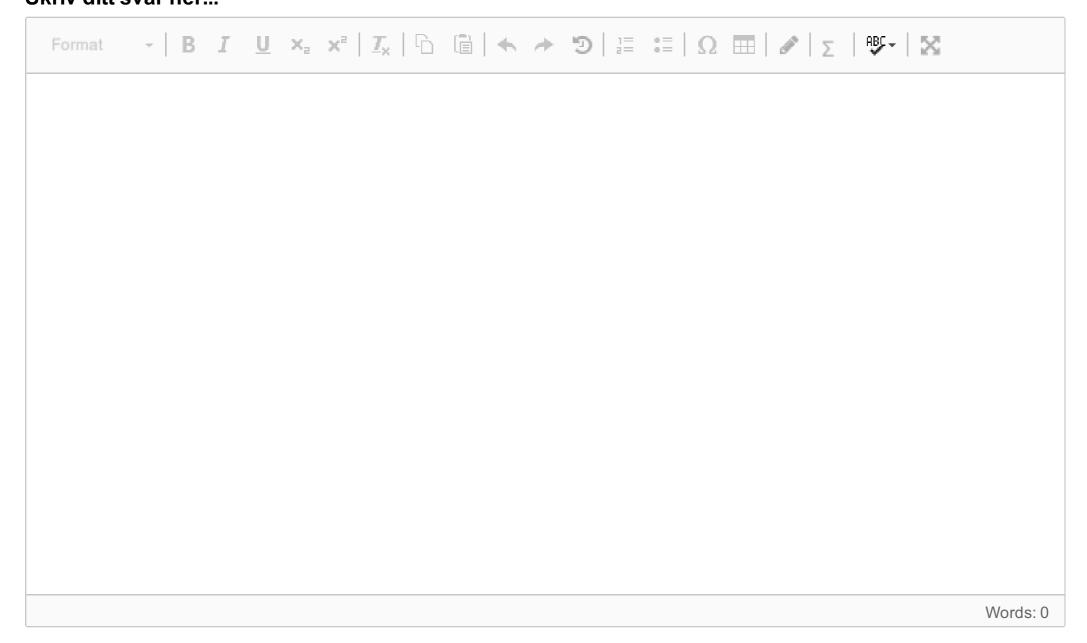
- Hvis det ikke er noen innlagt på sykehus, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

I figuren under har vi vist en relasjonsalgebra-spørring som skal finne datoer, representert med dag, måned og år, da det er registrert smitte i Trøndelag (county). Resultatet skal være sortert på år, måned og dag slik at de siste datoene kommer først.



Din oppgave er å finne ut hva som skal stå i de seks boksene merket med A-F.

## Skriv ditt svar her...



# <sup>4</sup> Problem 4 (5 %)

Ta utgangspunkt i følgende skjema for en relasjonsdatabase som holder oversikt over Korona-pandemiens utvikling i Norge. Primærnøkler er understreket.

RegisteredDates(<u>DateSeqNo</u>, DD, MM, YY, WeekNo) County(<u>CountyNo</u>, Name, Population)

Municipality(MunNo, Name, Population, CountyNo)

CountyNo er fremmednøkkel mot County, kan ikke være NULL

## Tested(MunNo, DateSeqNo, Number)

- Hvis det ikke er testet noen, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

#### Infected(MunNo, DateSeqNo, Number)

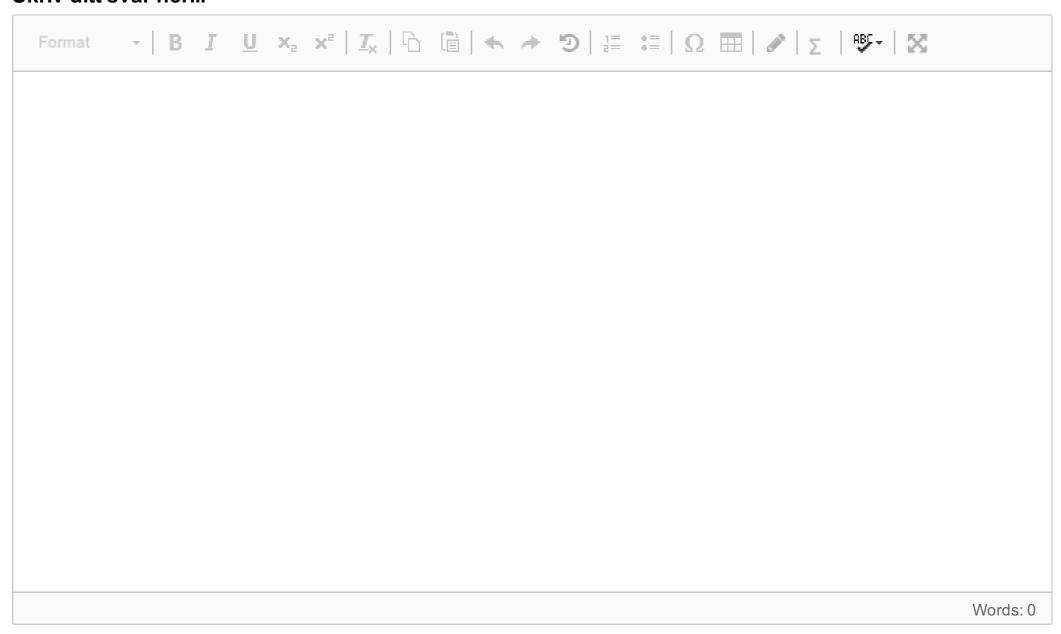
- Hvis det ikke er registert ny smitte, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

#### Hospitalized(MunNo, DateSeqNo, Number)

- Hvis det ikke er noen innlagt på sykehus, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

Skriv en SQL-spørring som finner ut hvor mange kommuner det er i de ulike fylkene i Norge. Resultatet skal inneholde fylkesnavn og antall kommuner, sortert etter synkende antall kommuner.

## Skriv ditt svar her...



# <sup>5</sup> Problem 5 (5 %)

Ta utgangspunkt i følgende skjema for en relasjonsdatabase som holder oversikt over Korona-pandemiens utvikling i Norge. Primærnøkler er understreket.

RegisteredDates(<u>DateSeqNo</u>, DD, MM, YY, WeekNo) County(<u>CountyNo</u>, Name, Population)

Municipality(MunNo, Name, Population, CountyNo)

• CountyNo er fremmednøkkel mot County, kan ikke være NULL

## Tested(MunNo, DateSeqNo, Number)

- Hvis det ikke er testet noen, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

#### Infected(MunNo, DateSeqNo, Number)

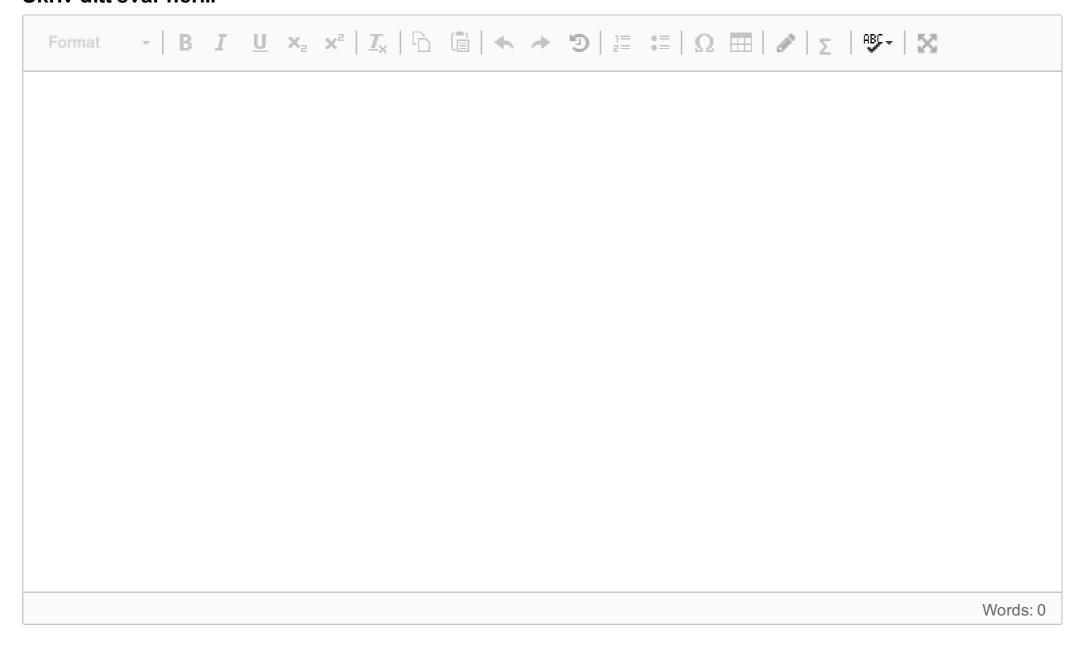
- Hvis det ikke er registert ny smitte, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

#### Hospitalized(MunNo, DateSeqNo, Number)

- Hvis det ikke er noen innlagt på sykehus, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

Skriv en SQL-spørring som finner alle datoer som er registrert i RegisteredDates, der det er registrert 0 innlagte på sykehus i Trondheim kommune. I resultatet skal du bare ta med DateSeqNo.

## Skriv ditt svar her...



# <sup>6</sup> Problem 6 (5 %)

Ta utgangspunkt i følgende skjema for en relasjonsdatabase som holder oversikt over Korona-pandemiens utvikling i Norge. Primærnøkler er understreket.

RegisteredDates(<u>DateSeqNo</u>, DD, MM, YY, WeekNo) County(<u>CountyNo</u>, Name, Population)

Municipality(MunNo, Name, Population, CountyNo)

CountyNo er fremmednøkkel mot County, kan ikke være NULL

#### Tested(MunNo, DateSeqNo, Number)

- Hvis det ikke er testet noen, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

## Infected(MunNo, DateSeqNo, Number)

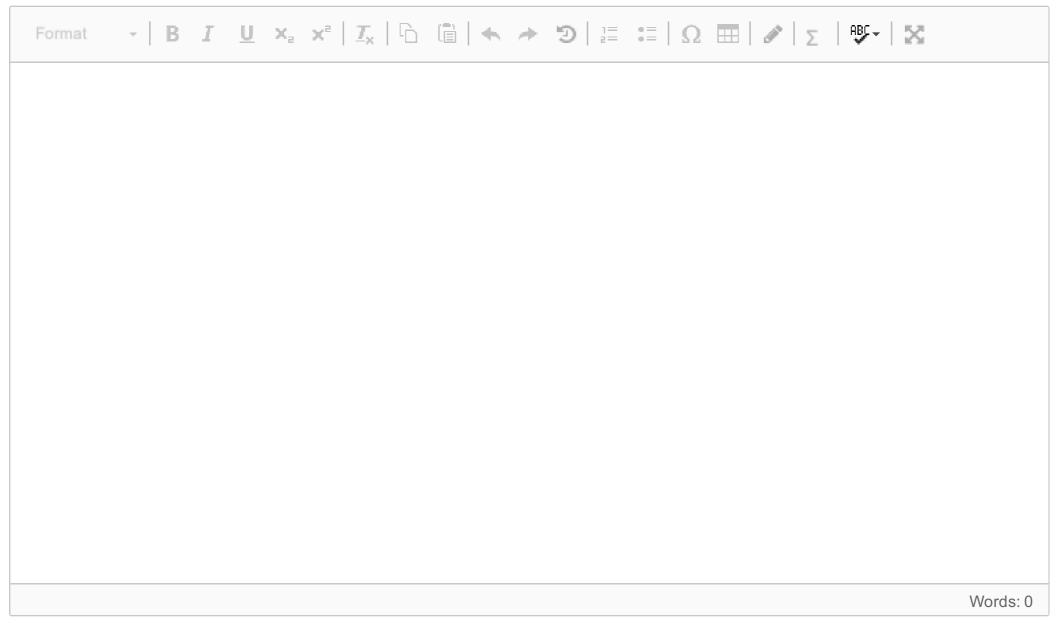
- Hvis det ikke er registert ny smitte, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

#### Hospitalized(MunNo, DateSegNo, Number)

- Hvis det ikke er noen innlagt på sykehus, settes Number til 0.
- MunNo er fremmednøkkel mot Municipality, kan ikke være NULL
- DateSeqNo er fremmednøkkel mot RegisteredDates, kan ikke være NULL

Skriv SQL-setningene som skal til for å registrere data for Trondheim kommune 6. april 2020. Du kan gå ut fra at Trondheim er registrert i Municipality-tabellen med kommunenummer 1601 (MunNo). Datoen 6. april 2020 må registreres med DateSecNo=41, DD=6, MM=4, YY=2020 og WeekNo=15. På den aktuelle datoen ble det i Trondheim testet 64 personer, det var 3 positive tester og 12 personer var innlagt på sykehus.

### Skriv ditt svar her...



# <sup>7</sup> Problem 7 (6 %)

Oppgaven består av tre deloppgaver som teller like mye.

## Oppgave a:

Lag et originalt eksempel på en tabell som er på andre normalform (2NF), men ikke på tredje normalform (3NF). Forklar hvorfor det er slik.

I denne oppgaven skal du ikke kopiere et eksempel fra en annen kilde, det vil gi null uttelling.

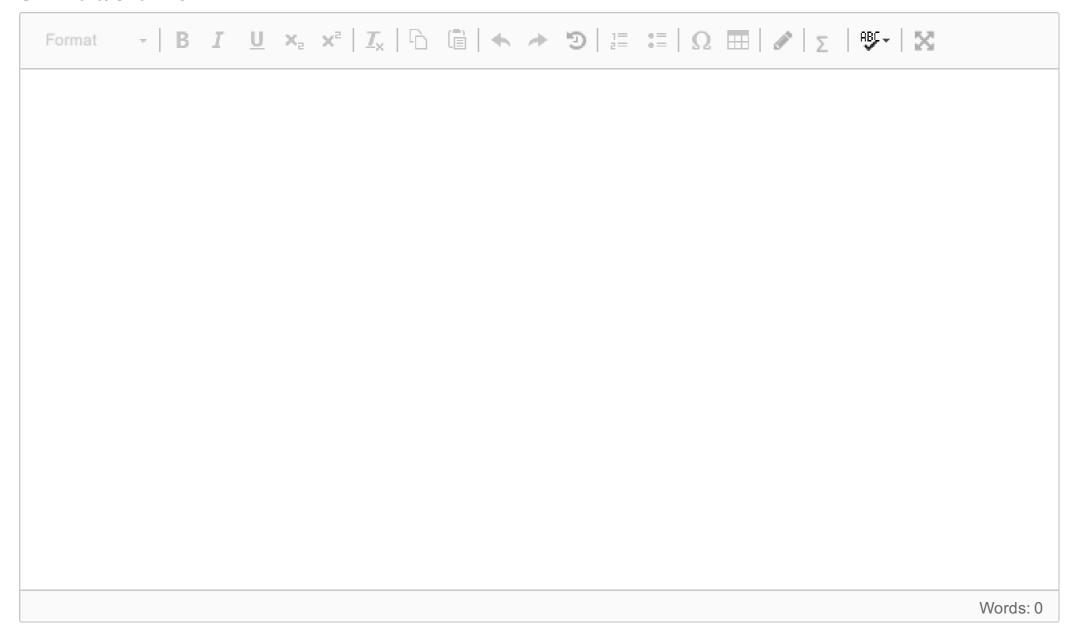
## Oppgave b:

Grei ut om problemene som kan oppstå på grunn av at tabellen du foreslår i deloppgave a ikke er på tredje normalform.

## Oppgave c:

Vis hvordan tabellen du foreslår i deloppgave a kan splittes opp slik at vi oppnår tredje normalform. Forklar hvorfor forslaget ditt er en god løsning.

#### Skriv ditt svar her...

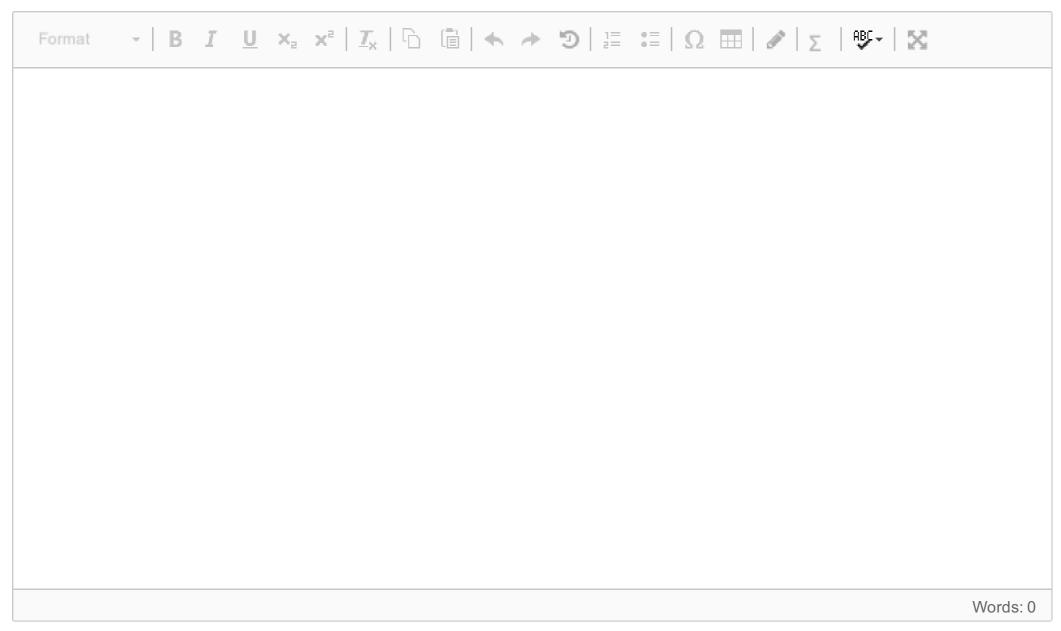


# <sup>8</sup> Problem 8 (3 %)

Ta utgangspunkt i en tabell R(A, B, C, D, E). Attributtene (A-E) er alle definert over en datatype med heltallene fra og med 1 til og med 10.

Hva er det maksimale antall rader som kan finnes i en tabellforekomst av R? Hvilken primærnøkkel vil R ha i dette tilfellet? Du *må* begrunne svarene.

## Skriv ditt svar her...

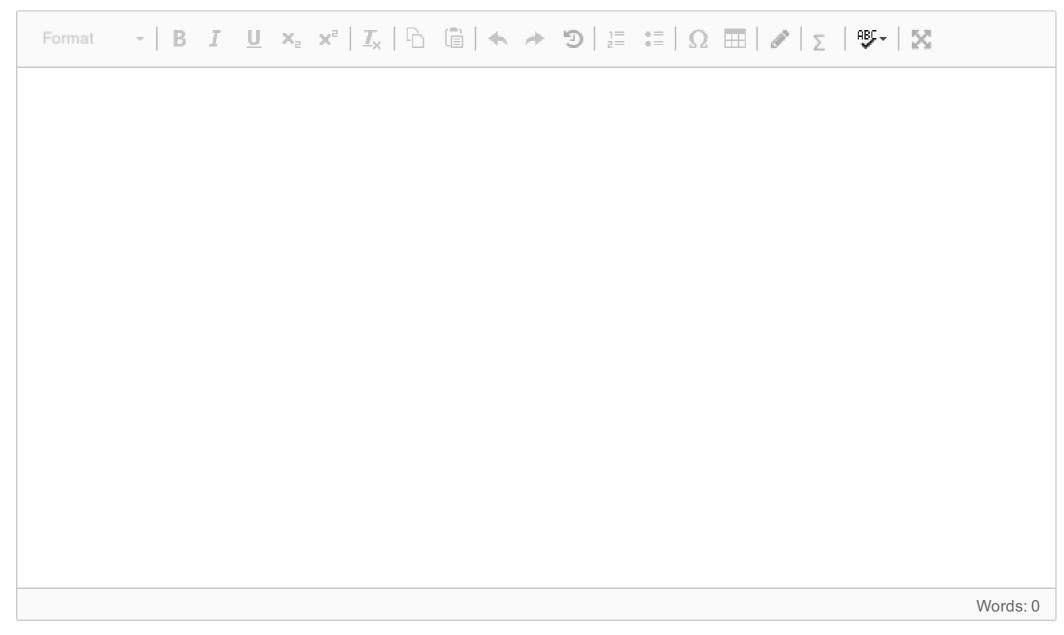


# <sup>9</sup> Problem 9 (2 %)

Ta utgangspunkt i en tabell R(A, B, C, D, E). Attributtene (A-E) er alle definert over en datatype med heltallene fra og med 1 til og med 10.

Anta at den funksjonelle avhengigheten C->BD gjelder for R. Hva er primærnøkkelen i dette tilfellet? Hvor mange rader kan det maksimalt være i en tabellforekomst av R? Du *må* begrunne svarene.

## Skriv ditt svar her...



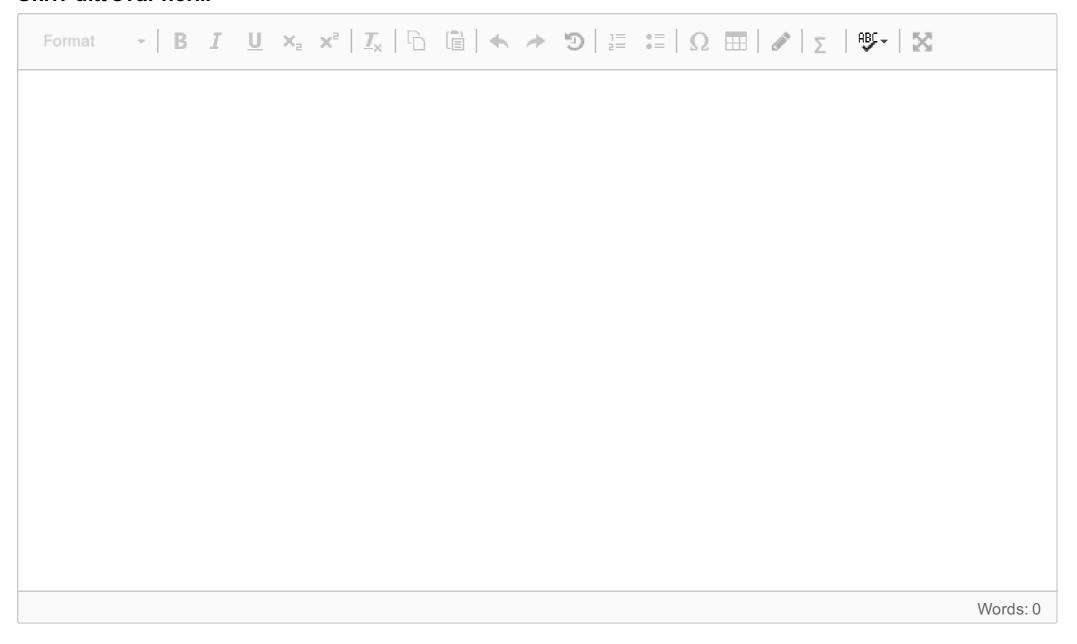
# <sup>10</sup> Problem 10 (4 %)

Ta utgangspunkt i tabellen Bil(modell, drivstoff, fremdrift). Et eksempel på en tabellforekomst er vist under.

Model	Fuel	Drive
RAV4	Petrol	2WD
RAV4	Petrol	4WD
RAV4	Diesel	2WD
RAV4	Diesel	4WD

Tabellen er på Boyce-Codd normalform siden det ikke er noen ikke-trivielle funksjonelle avhengigheter som gjelder for tabellen. Hvilke forutsetninger om miniverdenen må vi gjøre for at tabellen Bil skal være på fjerde normalform (4NF)? Du må begrunne svaret.

### Skriv ditt svar her...



# <sup>11</sup> Problem 11 (5 %)

Vi har en extendible hashingstruktur hvor vi starter med 4 blokker (binært: 00, 01, 10, 11), hvor hver blokk har plass til to nøkler.

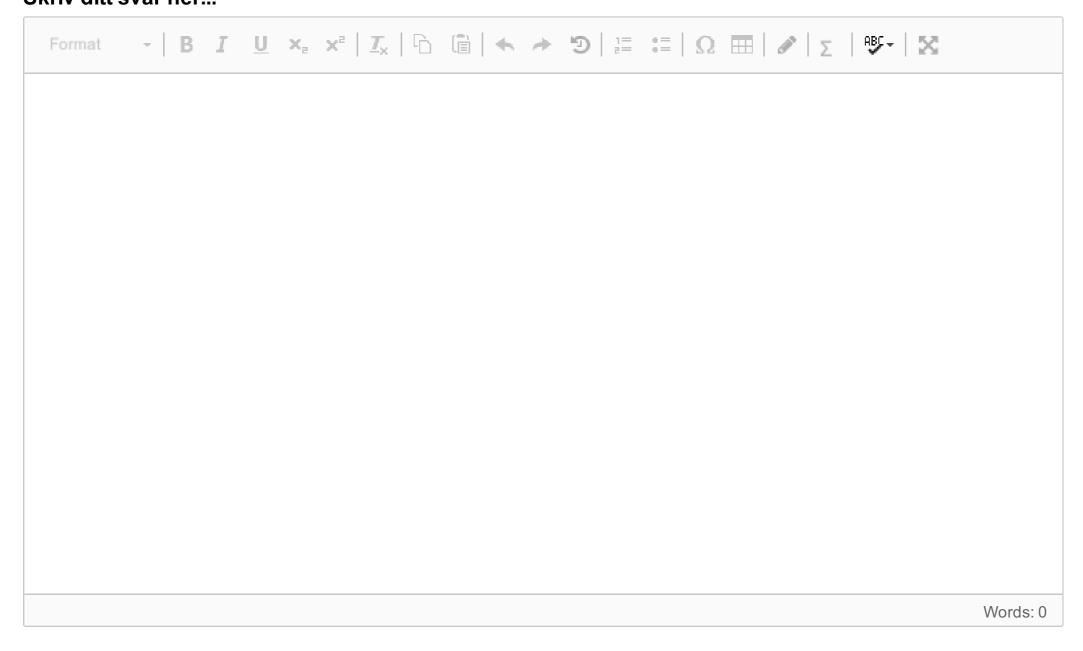
Når vi hasher inn de følgende nøklene: 7, 2, 14, 13, 11, 6, 1, 27.

Hvilken blokk er den første som må splittes?

Hva er lokal dybde for de to blokkene (den gamle og den nye) etter splitten?

Husk å svare på begge spørsmålene.

Skriv ditt svar her...

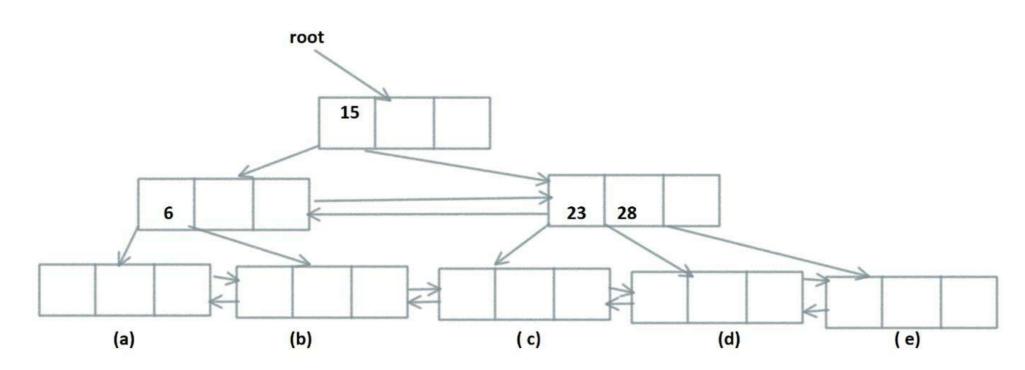


# <sup>12</sup> Problem 12 (8 %)

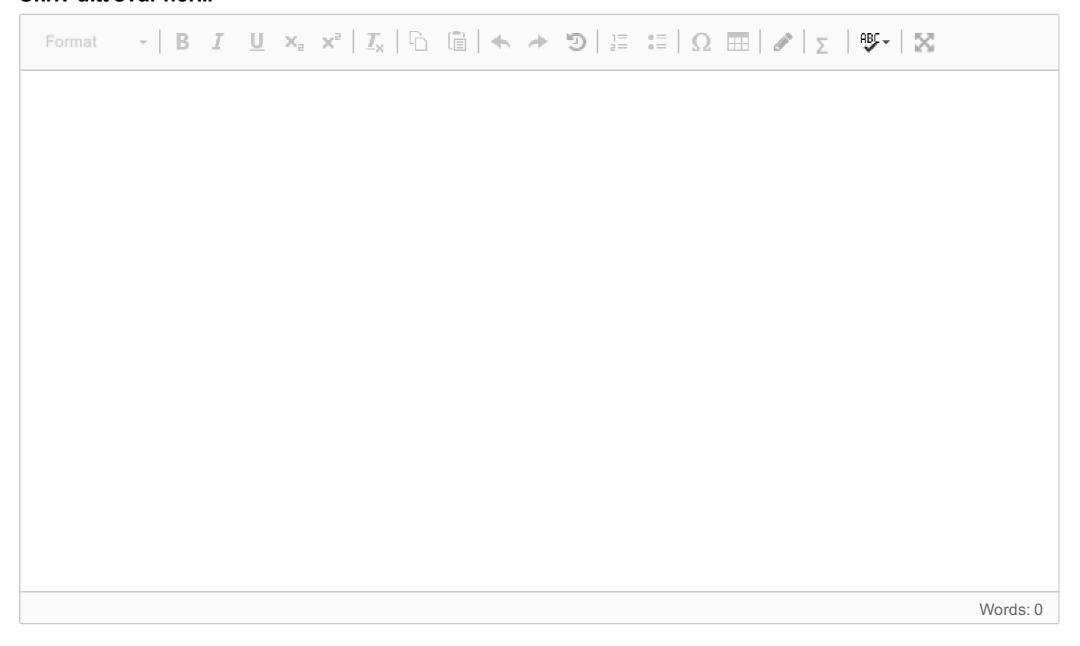
Vi skal sette inn nøkler i et B+-tre med plass til tre nøkler i hver blokk. Treet under skal inneholde nøkler på løvnivå (level=0). De er satt inn i den rekkefølgen de står i under. For hver blokk på løvnivå, dvs. (a)-(e), skriv hvilke nøkler som skal inn der. Du skal ikke tegne, bare skriv det som tekst: f.eks. (a): nøkkel 1, nøkkel 2, ... (b): (nøkkel), ...

Følgende nøkler er satt inn:

3, 6, 15, 23, 28, 34, 4, 8, 17 og 25.



## Skriv ditt svar her...



# <sup>13</sup> Problem 13 (10 %)

Vi har en tabell Student (studnr, pnr, fornavn, etternavn, studieprogram, epost)

Tabellen lagres i en heapfil med 2000 blokker. Det er 20 studentposter i hver blokk.

For å tvinge gjennom at **studnr** er primærnøkkel, brukes et B+-tre med **studnr** som søkenøkkel og Recordld som peker til posten i heapfila. B+-treet har 3 nivåer og 1000 blokker på løvnivå.

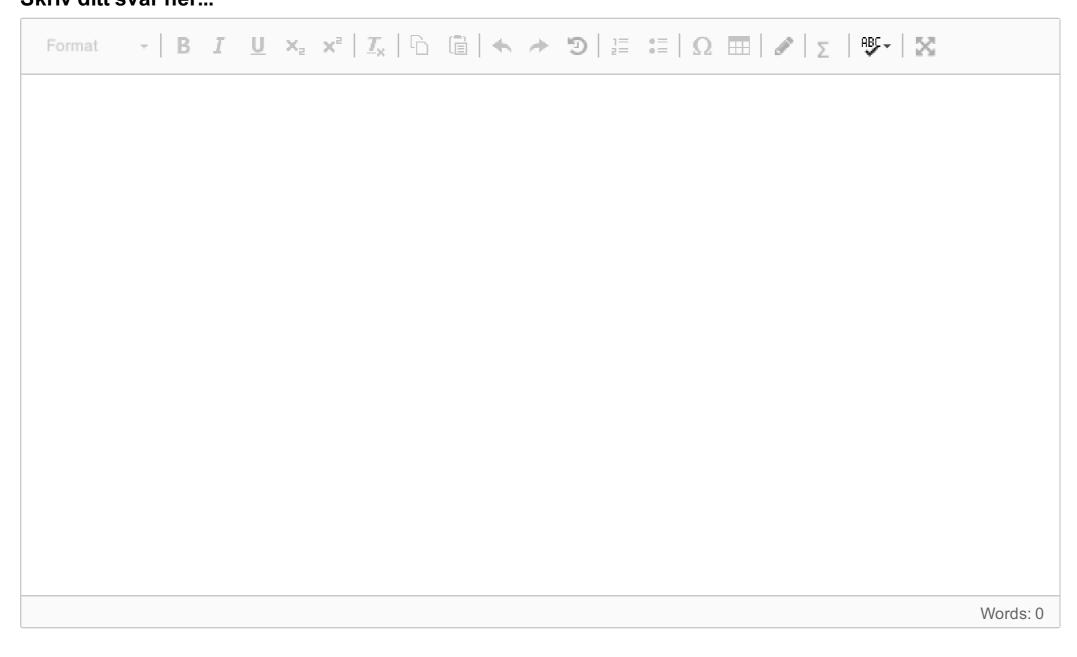
Vi har også et annet B+-tre med søkenøkkel **etternavn**. Dette B+-treet har også 3 nivåer, men 1500 blokker på løvnivå.

Hvor mange blokker aksesseres ved de følgende queryene. Husk å begrunne hvert svar.

- a) SELECT \* FROM Student WHERE studnr = 123456;
- b) SELECT \* FROM Student WHERE fornavn='Jon';
- c) SELECT \* FROM Student WHERE etternavn='Hansen';
- d) SELECT DISTINCT etternavn FROM Student ORDER BY etternavn;

Husk å svare på alle 4 delspørsmålene.

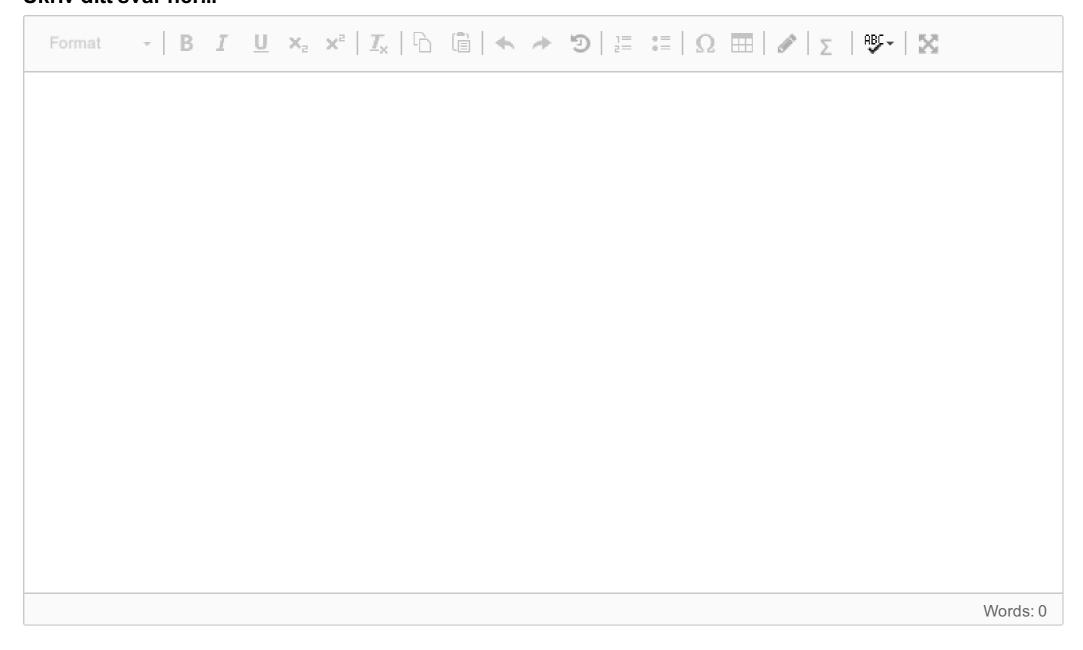
#### Skriv ditt svar her...



# <sup>14</sup> Problem 14 (2 %)

Hvilket isolasjonsnivå for transaksjoner må vi bruke hvis vi ønsker å unngå at en annen transaksjon setter inn poster mellom de postene som en transaksjon ønsker å scanne og lese? Begrunn svaret ditt.

## Skriv ditt svar her...



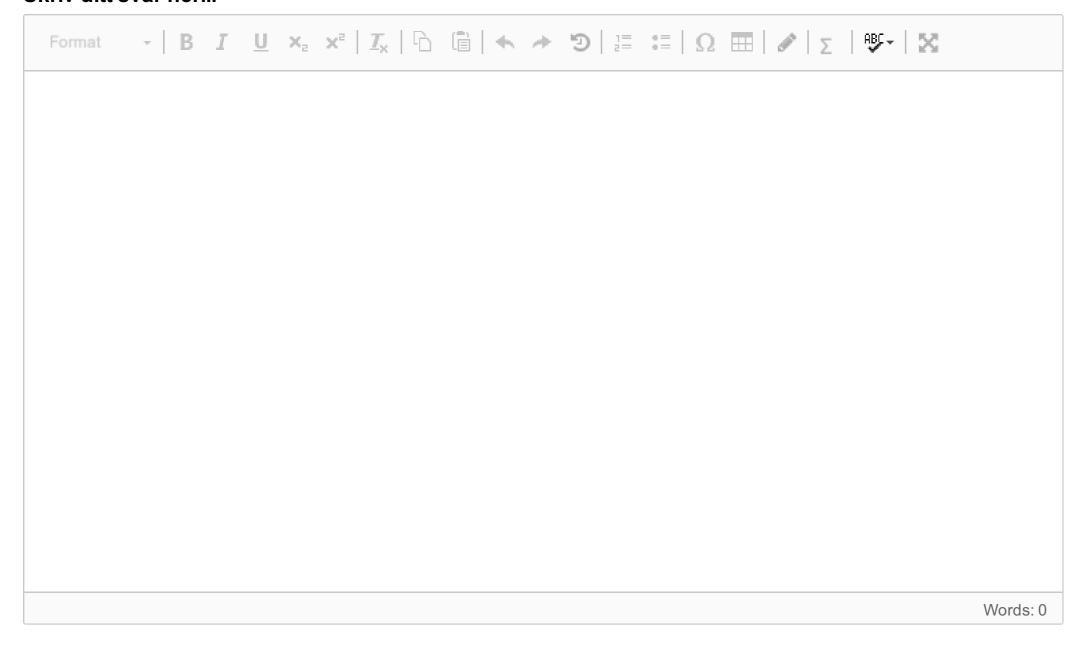
# <sup>15</sup> Problem 15 (5 %)

Vi har en historie som vi prøver å få utført ved at transaksjonene setter lese- og skrivelåser. Vi innfører tofaselåsing (2PL) av typen rigorous for denne historien.

r1(X);r2(X);w1(X);c1;r2(Y);r3(Z);w3(Z);c3;w2(Y);c2;

I hvilken rekkefølge committer de tre transaksjonene T1, T2 og T3 når vi forutsetter rigorous 2PL? Du skal ikke tegne historien, bare skriv rekkefølgen av transaksjonenes commit.

#### Skriv ditt svar her...



# <sup>16</sup> Problem 16 (5 %)

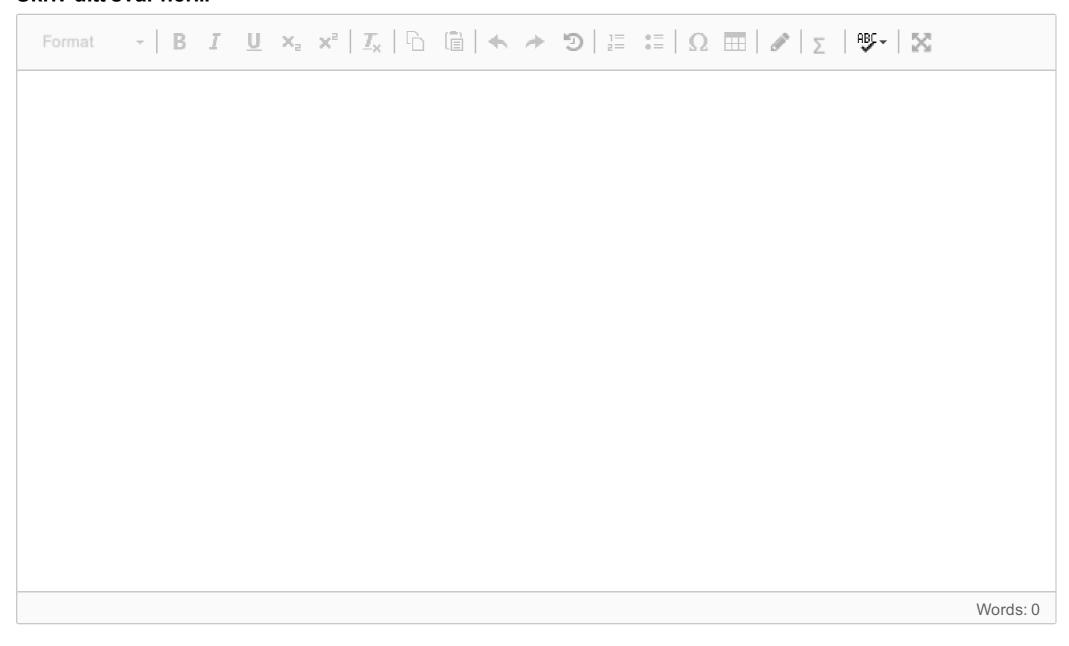
Vi har følgende logg som er funnet under recovery etter et krasj (vi bruker ARIES):

(104, T3, 103, Commit) (105, StartCkpt) (106, EndCkpt) (107, T4, NIL, Update, Page A, beflmage, aftImage) (108, T4, 107, Update, Page B, beflmage, aftImage) (109, T5, NIL, Update, Page C, beflmage, aftImage) (110, T5, 109, Update, Page A, beflmage, aftImage) (111, T4, 108, Commit)

Anta at DPT og Transaksjonstabellen er tomme i sjekkpunktloggposten med LSN 106 (EndCkpt). Loggpostene for update har formatet (LSN, Transaction, PrevLSN, Operation, Page, beflmage, aftlmage). Hvis det er den første loggede operasjonen for en transaksjon, vil PrevLSN være NIL. For Commit og Abort er formatet (LSN, Transaction, PrevLSN, Operation).

Skriv opp innholdet av Dirty Page Table (DPT) og Transaksjonstabellen etter at analysen har kjørt seg ferdig. Du skal ikke tegne disse, bare skriv de som tekst.

#### Skriv ditt svar her...



# <sup>17</sup> Problem 17 (5 %)

Anta ARIES-recovery er gjort for loggen i oppgave 16.

Skriv opp PageLSN for datablokkene A, B, C etter redo-recovery. Skriv også opp PageLSN for datablokkene A, B, C etter undo-recovery.

Husk å begrunne og å svare på begge spørsmål.

Skriv ditt svar her...

