

Institutt for datateknikk og informasjonsvitenskap

Eksamensoppgave i TDT4145 Datamodellering og databasesystemer

Faglig kontakt under eksamen:

Svein Erik Bratsberg: 995 39 963

Roger Midtstraum: 995 72 420

Eksamensdato: 1. juni 2015

Eksamenstid (fra-til): 09:00 - 13:00

Hjelpemiddelkode/Tillatte hjelpemidler:

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

Annen informasjon:

Målform/språk: Norsk bokmål

Antall sider: 6

Antall sider vedlegg: 0

Kontrollert av:

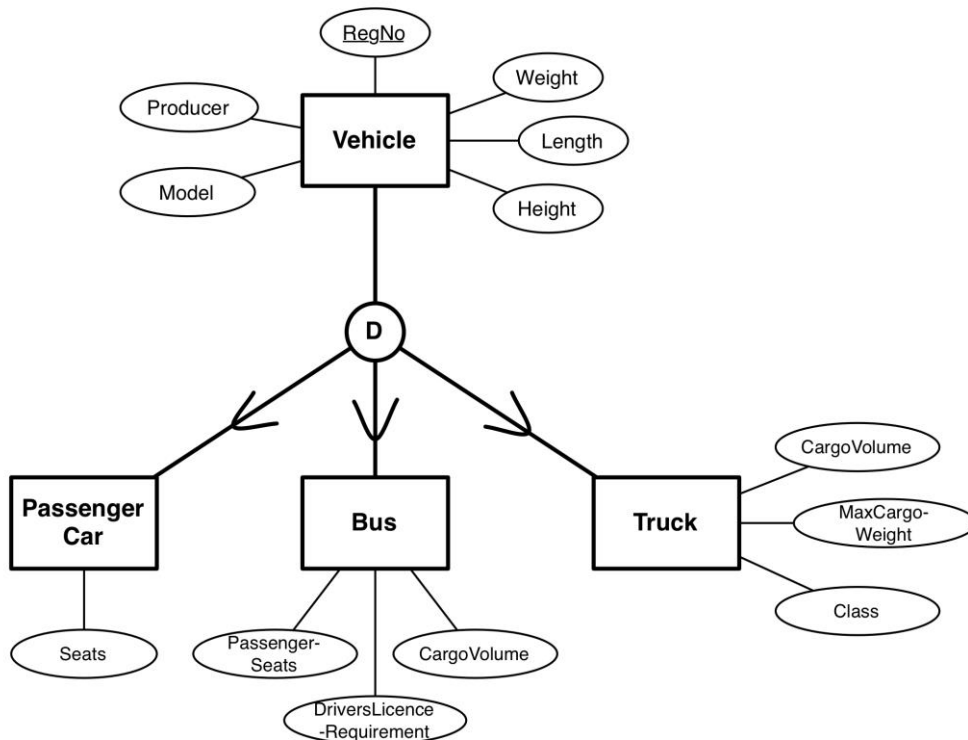
Svein-Olaf Hvasshovd (sign.)

Dato

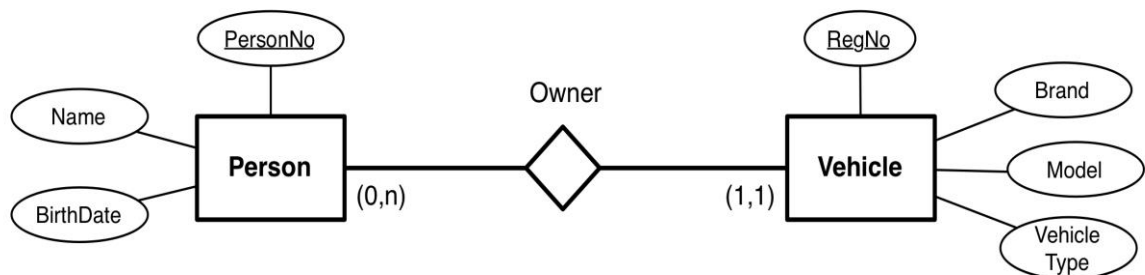
Sign.

Oppgave 1 – Datamodeller (20 %)

- a) (8 %) Under er vist et ER-diagram med spesialisering. Vis hvordan du vil oversette dette til en "ekvivalent" relasjonsdatabase (vis hvilke tabeller du ønsker å ha, med attributter, primær- og fremmednøkler). Forklar kort hvilke alternative løsninger som finnes og begrunn ditt valg.



- b) (12 %) Under er vist et ER-diagram for en enkel kjøretøydatabase.



Du skal ta utgangspunkt i denne datamodellen og utvide den slik at den også kan ta vare på data om ulykker der ett eller flere kjøretøy er involvert. For hver ulykke må vi kunne lagre en tekstlig beskrivelse av ulykken, GPS-posisjon, dato, tid og ulykkestype ("materiell skade", "personskade", "dødsulykke"). Vi ønsker å kunne holde rede på alle involverte personer og kjøretøy. For involverte kjøretøy ønsker vi å vite hvem som var fører, hvem som eventuelt var passasjerer og det skadeomfanget ("ingen", "liten", "omfattende", "totalvrak") kjøretøyet ble påført i ulykken. For alle personer som var involvert i ulykken, skal det kunne registreres skadeomfang ("uskadet", "lett skadet", "alvorlig skadet", "livstruende skadet", "dødelig skadet"). For sjåførere skal det kunne registreres om føreren hadde gyldig førerkort eller ikke og om vedkommende var edru eller påvirket av rusmidler da ulykken skjedde ("ukjent", "upåvirket", "påvirket").

I ER-modellen din kan du bruke alle virkemidler som er med i pensum. Du kan bruke norske navn på entitetsklasser, relasjonsklasser og attributter. Gjør kort rede for eventuelle forutsetninger som du finner det nødvendig å gjøre.

Oppgave 2 – Relasjonsalgebra og SQL (20 %)

Ta utgangspunkt i følgende relasjonsdatabase (primærnøkler er understreket) for en enkel fugleobservasjoner-database:

BirdGroups(BGID, GroupName)

BirdSpecies(BSID, SpeciesName, PrevalenceStatus, Prevalence, BirdGroupID)

BirdGroupID er fremmednøkkel mot BirdGroups-tabellen. BirdGroupID kan ikke ha NULL-verdi.

Observation(ObsNo, ObsDay, ObsMonth, ObsYear, SpeciesID, LocationNo, BirdCount)

SpeciesID er fremmednøkkel mot BirdSpecies-tabellen. LocationNo er fremmednøkkel mot Location-tabellen. SpeciesID og LocationNo kan ikke ha NULL-verdi.

Location(LNo, LocName, Description, MunicipalityNo)

MunicipalityNo er fremmednøkkel mot Municipality-tabellen. MunicipalityNo kan ikke ha NULL-verdi.

Municipality(Mno, MunName)

Relasjonsalgebra kan formuleres som tekst eller grafer. Hvis du behersker begge notasjonene foretrekker vi at du svarer med grafer, men du blir ikke trukket for å svare med tekst.

- (3 %) Lag en spørring i *relasjonsalgebra* som finner navn på alle fuglearter (eng: bird species) som er trekkfugler (PrevalenceStatus har verdien migratory).
- (4 %) Lag en spørring i *relasjonsalgebra* som finner navnet på alle fuglearter som er observert i Trondheim kommune (eng: Trondheim municipality) i 2015.
- (4 %) Lag en spørring i *SQL* som finner navn på alle fuglearter i fuglegruppen Thrushes (trostefugler). Resultatet skal være ordnet i alfabetisk rekkefølge.
- (4 %) Lag en spørring i *SQL* som finner artsnavn, antall observasjoner av arten i 2015 og samlet antall observerte individer av arten i 2015 (BirdCount), for alle fuglearter som er observert i 2015. Resultatet skal ordnes etter synkende antall individer.
- (5 %) Lag en spørring i *SQL* som finner navn på alle fuglearter som er observert i Trondheim kommune, men ikke i Trondheim i 2015.

NB! Oppgavesettet fortsetter på neste side.

Oppgave 3 – Teori (20 %)

- a) (3 %) Ta utgangspunkt i tabellen **Birds**(SpeciesID, SpeciesName, BirdGroup, Status) og følgende tabellforekomst:

SpeciesID	SpeciesName	BirdGroup	Status
1	Redwing	Thrushes	Migratory
2	Blackbird	Thrushes	Resident
3	Raven	Crows	Resident
4	Magpie	Crows	Resident
5	Chaffinch	Finches	Resident

Hvilke funksjonelle avhengigheter (eng: functional dependencies) er det rimelig å anta vil gjelde for denne tabellen? Forklar de forutsetningene du legger til grunn. Du trenger ikke å ta med trivielle funksjonelle avhengigheter eller funksjonelle avhengigheter som kan utledes ut fra funksjonelle avhengigheter i svaret ditt.

- b) (3 %) Gitt $R = \{A, B, C, D, E\}$ og $F = \{A \rightarrow B, B \rightarrow C, CD \rightarrow E\}$. Finn alle kandidatnøkler i R . Svaret må begrunnes.
- c) (4 %) Gitt $R = \{A, B, C, D, E\}$ og $F = \{A \rightarrow B, B \rightarrow C, CD \rightarrow E\}$. Gå ut fra at R oppfylles 1. normalform. Bestem den høyeste normalformen som oppfylles av R . Svaret må begrunnes.
- d) (6 %) Gitt $R = \{A, B, C, D, E\}$ og $F = \{A \rightarrow B, B \rightarrow C, CD \rightarrow E\}$. En mulig dekomponering av R er $R_1(A, B)$, $R_2(B, C)$ og $R_3(C, D, E)$. Er dette en god dekomponering? Svaret må begrunnes.
- e) (4 %) Ta utgangspunkt i tabellen **Teaching**(Lecturer, Subject, Campus). Gå ut fra at fler-verdi avhengighetene (eng: multi-value dependency) $\text{Lecturer} \twoheadrightarrow \text{Subject}$ og $\text{Lecturer} \twoheadrightarrow \text{Campus}$ gjelder og at følgende tupler (rader) er lagret i tabellen:

(svein, db, gløs), (rune, db, gjøvik), (tore, db, kalvskinnet) og (mads, os, gløs).

Tegn tabellen med rader og kolonner slik at du viser alle tupler (rader) som nødvendigvis må være med i tabellforekomsten.

NB! Oppgavesettet fortsetter på neste side.

Oppgave 4 – Lagring og skalering (5 %)

Du jobber i en bedrift som lanserer en webbutikk hvor antall kunder og kjøp er usikkert. Databasen består av tabellene Kunde, Ordre, Produkt og OrdreLinje. Kunder identifiseres ved hjelp av epost-adresse, og nesten alle aksesser av tabellene skjer direkte via primærnøkkel, dvs. epostadresse, kundenummer, ordrenummer og produktnummer. Hvilke(n) lagrings/indeksstruktur(er) mener du det er lurt å bruke for tabellene når du er usikker på om det blir 1000 eller 1000000 kunder? Gi en begrunnelse for ditt valg.

Oppgave 5 – B+-trær og queryutføring (15 %)

Anta følgende tabell:

```
CREATE TABLE Employee (empno INT PRIMARY KEY, lastname CHAR(30), firstname CHAR(30), email CHAR(30), startyear INT, salary INT);
```

Anta tabellen er lagret i et clustered B+-tre med 1500 blokker på løvnivå. Empno er søkenøkkel i B+-treet. Vi antar B+-treet har høyde 3.

- a) (10 %) Gjør et estimat på hvor mange blokker som leses og skrives ved de følgende SQL-setningene:
- i) INSERT INTO Employee VALUES (123123,'Hansen','Hans','hans@email.org',2015,100000);
 - ii) SELECT lastname, firstname, email, startyear, salary FROM Employee WHERE empno = 123123;
 - iii) SELECT * FROM Employee;
 - iv) SELECT COUNT(*) FROM Employee WHERE empno > 100000;
- Anta at første ansatte hadde empno = 10000 og at alle nummer er brukt fram til 123123.

Gi en begrunnelse/forklaring for hvert svar.

- b) (5 %) Hvis queriet 'SELECT * FROM Employee WHERE startyear=2015' kjøres veldig ofte, ville du ha endret lagringen/indekseringen? Hvorfor/hvorfor ikke?

Oppgave 6 – Transaksjoner - gjenopprettbarhet (10 %)

Anta følgende historier:

H1: r1(A); w1(A); r2(B); c1; r2(A); w2(A); c2;

H2: r1(A); w1(A); w2(A); c2; c1;

H3: r1(A); r2(C); r1(C); r3(A); r3(B); w1(A); w3(B); r2(B); w2(C); w2(B); c1; c2; c3;

Bestem recoveryegenskapene til historiene (unrecoverable, recoverable, ACA, strict). Begrunn svarene.

Oppgave 7 – Transaksjoner - recovery (10 %)

Anta at vi bruker ARIES-recovery og la A, B, og C være dataelementer. Etter en krasj ble følgende logg funnet med en loggpost i hver rad av tabellen:

LSN	Last_lsn	Transaction	OpType	Page_id	Other_info
101	0	T1	Update	C	...
102	0	T2	Update	B	...
103	101	T1	Commit		...
104			Begin_ckpt		
105			End_ckpt		
106	0	T3	Update	A	...
107	102	T2	Update	C	...
108	107	T2	Commit		...

- a) Anta T1, T2 og T3 er alle transaksjonene som finnes og at transaksjonstabellen som finnes i loggpost 105 ser slik ut:

Transaction	Last_lsn	Status
T1	103	Commit
T2	102	In progress

Dirty Page Table som finnes i loggpost 105 ser slik ut:

Page_id	Rec_lsn
C	101
B	102

Hvordan ser transaksjonstabellen og Dirty Page Table ut etter analysefasen av recovery?

- b) Hvilke loggposter vil legges til loggen under **undofasen** av recovery her?