

INF3100 V2018

Obligatorisk oppgave nr. 1

Oppgavesettet skal løses og leveres individuelt.

Gjennomføring og innlevering av oppgaven skal skje i henhold til gjeldende retningslinjer ved Institutt for informatikk, se

www.uio.no/studier/admin/obligatoriske-aktiviteter/mn-ifi-oblig.html

Enhver innlevering av besvarelse på en obligatorisk oppgave tas som en bekreftelse på at retningslinjene er lest og forstått.

Innleveringsfrist: Fredag 23. mars kl. 23.59.

Fristen er absolutt, og det blir ikke gitt utsettelse. Alle spørsmålene må besvares for å få godkjent besvarelsen, med unntak av oppgave 5C, der det er tilstrekkelig å besvare ett av punktene C1–C3.

Oppgave 1 Oppvarming

Det begrepsmessige skjemaet for en relasjonsdatabase for plassreservering i tog inneholder (blant annet) følgende tabeller:

$Tog(\underline{togNr}, startSt, endeSt, ankomstTid)$

$TogTabell(\underline{togNr}, \underline{avgangstid}, stasjon)$

$Plass(\underline{dato}, \underline{togNr}, \underline{vognNr}, \underline{plassNr}, vindu, ledig)$

Primærnøkler er understreket. $togNr$ er fremmednøkkel til Tog både i $TogTabell$ og $Plass$.

- A. Skriv en passende CREATE TABLE statement for å opprette tabellen $Plass$. Gjør fornuftige antagelser angående datatyper og constraints, og forklar disse kort.
- B. Skriv en spørring som for hvert tog finner den lengste perioden mellom togets ankomsttid til endestasjonen og togets avgangstid derfra. Anta at tidene har datatype TIME (altså ingen dato, bare timer, minutter, og sekunder). Skriv ut de $togNr$ som har den lengste slike perioden.
- C. Uttrykk i relasjonsalgebra at $(togNr, stasjon)$ er kandidatnøkkel i relasjonen $TogTabell$. Hint: Husk at vi i relasjonsalgebraen kan ha booleske uttrykk av formen $E_1 \subseteq E_2$ der E_1 og E_2 er relasjonsuttrykk.

Oppgave 2 FDer og MVDer

Gitt relasjonen $\mathcal{R}(A, B, C, D, E, F, G)$.

La $\mathcal{Q} = \{CDE \rightarrow B, AF \rightarrow B, B \rightarrow A, BCF \rightarrow DE, D \rightarrow G\}$ være de integritetsreglene som gjelder for \mathcal{R} .

- A. Hvilke kandidatnøkler har \mathcal{R} ?
- B. Finn den høyeste normalformen som \mathcal{R} tilfredsstiller.
- C. La $\mathcal{D} = \{ABF, ACF, BCDE, DG\}$ være en dekomposisjon av \mathcal{R} . Avgjør om \mathcal{D} er tapsfri med hensyn på \mathcal{Q} .
- D. (i) Dekomponer \mathcal{R} tapsfritt til BCNF. Start dekomposisjonen ved å ta utgangspunkt i FDen $CDE \rightarrow B$.
(ii) Er dekomposisjonen FD-bevarende?
(iii) Kan dekomposisjonen ha støyinstanser?

E. Vis at $CDF \rightarrow B$ ikke holder.

F. Utvid \mathcal{Q} med MVDen $DG \twoheadrightarrow AC$. Vis at $CDF \rightarrow B$ nå følger fra \mathcal{Q} .

Oppgave 3 Ekteskap

Nedenfor følger skjemaet for en liten relasjonsdatabase som holder rede på navn og navneendringer. Man kan skifte fornavn eller etternavn (eller begge deler) ved å sende en søknad til Folkeregisteret. Dessuten kan man skifte etternavn i forbindelse med at man gifter seg; en eller begge ektefeller kan skifte etternavn i forbindelse med vielsen.

Person(fnr, etternavn, fornavn, adresse)

Ekteskap(dato, fnr1, fnr2, etternavn1, etternavn2)

ForrigeNavn(dato, fnr, etternavn, fornavn)

Person inneholder nåværende navn. *Ekteskap* inneholder etternavnene slik de ser ut etter vielsen. *ForrigeNavn* inneholder forrige etternavn og fornavn. Primærnøkler er understreket. I tillegg inneholder *Ekteskap* kandidatnøgkelen (dato, fnr2). Hver av fnr1 og fnr2 i *Ekteskap* er fremmednøkler til fnr i *Person*. Hvis *ForrigeNavn* inneholder et tuppel (d, f, x, y) , så inneholder *Ekteskap* ett av tuplene $(d, f, _, _, _)$ og $(d, _, f, _, _)$ der ‘_’ representerer en vilkårlig verdi.

- A. Finn ved hjelp av en SQL-spørring navn og adresse til alle personer som ved en vielse i perioden 2000–2010 skiftet etternavn til et navn som er forskjellig fra ektefellens.
- B. Det er blitt ganske populært for ektefeller å ha “reverserte etternavn”. Et eksempel er SV-politikeren Heikki Eidsvoll Holmås og hans ektefelle Sunniva Holmås Eidsvoll; før de giftet seg, het de henholdsvis Heikki Holmås og Sunniva Eidsvoll. Finn ved hjelp av en SQL-spørring navn og adresse til alle personer som ved en vielse har reversert etternavnene sine.

Her er noen eksempler på formen

$\text{forrigeNavn}_1, \text{forrigeNavn}_2 \rightarrow \text{navnEtterVielsen}_1, \text{navnEtterVielsen}_2$

og om eksempelet oppfyller/ikke oppfyller betingelsen. (Det kan være andre tolkninger som også gir mening.)

A. (Nei/ja gjelder den første av ektefellene.)

Andersen, Hansen → Andersen-Hansen, Andersen-Hansen	nei
Andersen, Hansen → Andersen-Hansen, Hansen	ja
Andersen, Hansen → Hansen-Andersen, Andersen-Hansen	ja

B. (Tolkning: Ektefelles forrige navn stilles foran eget etternavn, med eller uten bindestrek mellom de to etternavnene.)

Andersen, Hansen → Hansen, Hansen	nei
Andersen, Hansen → Hansen, Andersen	nei
Andersen, Hansen → Hansen Andersen, Andersen Hansen	ja
Andersen, Hansen → Hansen-Andersen, Andersen-Hansen	ja

Oppgave 4 Rekursiv SQL

Relasjonen under inneholder informasjon om aksjeselskaper mm. og hvilke personer som har hvilke roller i selskapene.

Selskapsinfo(selskap, rolle, person)

Rolle er ‘styreleder’, ‘nestleder’, ‘styremedlem’, ‘vamedlem’, ‘daglig leder’, ‘kontaktperson’, mm. Informasjonen i *Selskapsinfo* kan tolkes som en graf der hver node representerer et selskap eller en person. Det er en kant mellom et selskap og en person hvis personen har en rolle i selskapet. Langs hver sti i grafen er nodene vekselvis selskaper og personer.

A. Finn ved hjelp av en rekursiv SQL-spørring hvor mange personer det er på den korteste stien mellom Olav Thorsen og Celina Monsen i grafen.

(Hvis Olav Thorsen og Celina Monsen har roller i samme selskap, er svaret ‘1’. Hvis ikke, og det finnes en person p og selskaper $s_1 \neq s_2$ slik at Olav Thorsen og p begge har roller i s_1 , og p og Celina Monsen begge har roller i s_2 , er svaret ‘2’. Osv. Hvis det ikke er noen sti mellom dem, så kan du velge om du vil returnere svaret ‘0’ eller et “tomt” svar, dvs. et svar med 0 rader.)

Skattemyndighetene er interessert i konstellasjoner hvor det er sykliske avhengigheter mellom personer i viktige roller. Eksempel:

Person A er daglig leder i selskap X og styreleder i selskap Y.

Person B er daglig leder i selskap Y og styremedlem i selskap Z.

Person C er daglig leder i selskap Z og nestleder i selskap X.

- B. Finn ved hjelp av en rekursiv SQL-spørring alle sykler som inneholder 3, 4 eller 5 personer, og der hver person har rollen ‘daglig leder’ i ett selskap og en av rollene ‘styreleder’, ‘nestleder’ eller ‘styremedlem’ i neste selskap i sykkelen. Skriv for hver slik sykkel ut personene og selskapene i sykkelen.

Du kan lage din egen testdatabase ved å kopiere innholdet i

www.uio.no/studier/emner/matnat/ifi/INF3100/v16/undervisningsmateriale/obliger/selskapsinfo.sql

En grafisk fremstilling av testdatabasen finner du her:

www.uio.no/studier/emner/matnat/ifi/INF3100/v16/undervisningsmateriale/obliger/selskapsinfo.pdf

Oppgave 5 Filmdatabasen

I denne oppgaven skal dere i utgangspunktet bruke den *fulle* versjonen av filmdatabasen¹ til å besvare spørsmålene under (men se også spesifikke kommentarer under de enkelte deloppgavene). Besvarelsen skal for hvert punkt inneholde (i) en SQL-spørring og (ii) resultatet fra en kjøring av SQL-spørringen.

- A. Skriv ut serietittel, produksjonsår og antall episoder for de yngste TV-seriene i filmdatabasen (dvs. de med størst verdi i attributtet *firstprod-year*). (Det er mellom 5 og 10 slike TV-serier.)
- B. Lag en liste over alle deltakelsestyper og hvor mange personer som prosentvis faller inn under hver deltakelsestype. Listen skal være sortert etter fallende prosentpoeng. Ta med ett siffer etter desimaltegnet.
- C. Gjør minst én av følgende tre oppgaver:

- C1. Finn navnet på personer som har “byttet” deltakelsestype med en annen person fra en film til en annen, og hvor dette gjelder minst 50 slike andre personer. (Det er mellom 1 og 10 personer i resultatmengden.) Du finner et eksempel på neste side.

Denne spørringen tar noe tid (anslagsvis mellom en halv og en time mot den fulle databasen), og vi har i skrivende stund ikke funnet noen god måte å få ned tiden på. Derfor godtar vi at du bare kjører spørringen mot det nedskalerte segmentet av filmdatabasen, men

¹Se informasjon om filmdatabasen og om bruk av Postgres på kursets hjemmesider.

da skal du i stedet finne dem hvor det er minst 15 slike andre personer (da blir det mellom 1 og 10 personer i resultatmengden også her).

C2. Finn for- og etternavn på alle kvinnelige skuespillere som har deltatt i mer enn 50 kinofilmer, og det for hver film er slik at etternavnet deres kommer først i alfabetet blant de kvinnelige skuespillerne i filmen. (Det er mellom 5 og 20 slike personer.)

C3. Finn for- og etternavn på alle mannlige regissører som har laget mer enn 50 filmer, og der det er minst én kvinnelig skuespiller som har vært med i samtlige filmer av denne regissøren. (Det er mellom 1 og 5 slike personer.)

Denne spørringen tar potensielt så lang tid² at vi godtar at du bare kjører spørringen mot det nedskalerte segmentet av filmdatabasen som ligger på ditt lokale arbeidsområde, men da må du i stedet finne dem som har lagd mer enn 5 slike filmer for å få ut et ikke-tomt svar. (Da blir det mellom 1 og 5 personer i resultatmengden også her.)

Eksempel på bytte av deltakelsestyper (se C1): I filmen ‘Les folies d’Élodie’ er Claude Chabrol skuespiller, mens Paul Gégauff er manusforfatter. I filmen ‘La ligne de démarcation’ er det motsatt: der er Paul Gégauff skuespiller og Claude Chabrol manusforfatter (i tillegg til at han er regissør). Så Claude Chabrol er med i resultatmengden hvis det i tillegg til Paul Gégauff finnes minst 49 andre slike personer. (I filmene ‘Les godelureaux’ og ‘La ligne de démarcation’ er Jean-Marie Arnoux og Mario David begge skuespillere, men dette er ikke noe bytte av deltakelsestype.)

²Vi kjenner til én spørring som tar mindre enn ett minutt mot den fulle databasen, mens andre, tilsynelatende fornuftig formulerte spørringer, har vist seg å ta mangfoldige timer. (Noen spørringer avbrøt vi fordi databasesystemet etter ganske mange timer fortsatt ikke hadde produsert noe resultat.) Også mot det nedskalerte segmentet kan noen av spørringene i verste fall ta flere timer.

title	lastname	firstname	parttype
Godelureaux, Les	Arnoux	Jean-Marie	cast
Godelureaux, Les	Chabrol	Claude	writer
Godelureaux, Les	Chabrol	Claude	director
Godelureaux, Les	Chabrol	Claude	cast
Godelureaux, Les	David	Mario	cast
Godelureaux, Les	Gégauff	Paul	writer
Folies d'Élodie, Les	Chabrol	Claude	cast
Folies d'Élodie, Les	Gégauff	Paul	writer
Ligne de démarcation, La	Arnoux	Jean-Marie	cast
Ligne de démarcation, La	Chabrol	Claude	director
Ligne de démarcation, La	Chabrol	Claude	writer
Ligne de démarcation, La	David	Mario	cast
Ligne de démarcation, La	Gégauff	Paul	cast

Slutt på obligatorisk oppgave 1