

# UNIVERSITETET I OSLO

## Det matematisk-naturvitenskapelige fakultet

Eksamen i                    INF3100 — Databasesystemer

Eksamensdag:            8. juni 2009

Tid for eksamen:        14.30–17.30

Oppgavesettet er på 3 sider.

Vedlegg:                 Ingen

Tillatte hjelpemidler: Kalkulator og ordbok

Kontroller at oppgavesettet er komplett før  
du begynner å besvare spørsmålene.

*Les oppgavene nøye, og lykke til!*

### Oppgave 1 Relasjonsdatabaser (55%)

I Uqbar, Ruritania, foregår søkningen til videregående skoler ved at hver elev oppgir inntil fem skoler og inntil fem linjer som de ønsker seg til (egentlig i prioritert rekkefølge, men det skal vi ikke bry oss om her). Elevene søker altså *ikke* konkrete linjer på de ulike skolene, blant annet fordi det på søknadstidspunktet ikke nødvendigvis er klart hvilke skoler som kommer til å tilby de ulike linjene.

Alle søknadene lagres i en relasjonsdatabase som blant annet inneholder relasjonene

```
Skolesøknad(søknadsNr, skole)
Linjesøknad(søknadsNr, linje)
Søknad(fødselsNr, søknadsNr)
Elev(fødselsNr, navn, karaktersnitt)
```

Skolesøknad har primærnøkkel (søknadsNr, skole).

Linjesøknad har primærnøkkel (søknadsNr, linje).

Søknad har primærnøkkel fødselsNr. Dessuten er søknadsNr kandidatnøkkel i Søknad.

Elev har primærnøkkel fødselsNr.

En oversikt over hvilke skoler som tilbyr hvilke linjer, vil etterhvert lagres i følgende relasjon:

```
Tilbud(skole, linje, antallPlasser)
```

Her er (skole, linje) primærnøkkel.

*(Fortsettes på side 2.)*

**1a FDer og MVDer (20%)**

Et alternativt design for søknadene er

```
Elev(fødselsNr, søknadsNr, navn, karaktersnitt)
Søknad(søknadsNr, skole, linje)
```

- (i) Hvis disse relasjonene skal ha samme instanser som de opprinnelige, hvilke FDer og MVDer må da gjelde?
- (ii) Hvilken normalform er relasjonen `Elev` på? Begrunn svaret.
- (iii) Hvilken normalform er relasjonen `Søknad` på? Begrunn svaret.

**1b SQL (25%)**

Ta utgangspunkt i de opprinnelige relasjonene fra innledningsteksten.

- (i) Bruk SQL til å lage en liste over hvor populære de ulike `(skole, linje)`-kombinasjonene er, basert på antall søknader. De med størst søkning skal komme først.
- (ii) Bruk SQL til å finne `(skole, linje)`-kombinasjoner som tilbys, men som ikke er søkt av en eneste elev.
- (iii) Bruk SQL til å finne fødselsnummer og navn på de elevene som bare har søkt linjer som ikke tilbys på noen av skolene de har søkt (og som derfor ikke kan tilbys plass på noen av sine skolealternativer).

**1c Relasjonsalgebra (10%)**

Besvar oppgave 1b punkt (i) med relasjonsalgebra.

**Oppgave 2 Dekomposisjon (15%)**

- (i) Hva er en tapsfri dekomposisjon?

La  $\mathcal{R}$  være en relasjon med de åtte attributtene A, B, C, D, E, F, G og H. La mengden av FDer som gjelder, være

$$\mathcal{F} = \{AC \rightarrow H, BE \rightarrow F, DH \rightarrow FG, EF \rightarrow D, FG \rightarrow AE\}$$

og sett dekomposisjonen til  $\mathcal{D} = \{ABCE, ACFGH, BCDFG, BDEG\}$ .

- (ii) Gi et begrunnet svar på om  $\mathcal{D}$  er en tapsfri dekomposisjon av  $\mathcal{R}$  med hensyn på  $\mathcal{F}$ .

(Fortsettes på side 3.)

### Oppgave 3 Transaksjonshåndtering (30%)

Gitt tre transaksjoner  $T_1$ ,  $T_2$  og  $T_3$  samt en eksekveringsplan  $S$ :

$$S: \quad r_1(a); r_1(b); r_2(c); w_1(b); r_2(b); r_3(d); w_2(c); r_1(c); \\ w_1(c); r_3(c); w_2(b); w_3(c); w_1(a); w_3(d)$$

#### 3a Serialiserbarhet og låser (15%)

- (i) Avgjør om  $S$  er konfliktserialiserbar. Begrunn svaret ditt.
- (ii) Vi skal så benytte eksklusive låser i transaksjonshåndteringen. La  $l_i(x)$  betegne at  $T_i$  tar en lås på elementet  $x$  og  $u_i(x)$  at  $T_i$  frigir låsen på  $x$ . Sett inn låser i transaksjonene i henhold til tofaselås-protokollen (2PL) og beskriv hva som skjer hvis lese- og skriveoperasjonene så langt som mulig utføres i rekkefølgen angitt i  $S$ .

#### 3b Distribuerte transaksjoner (15%)

- (i) Beskriv prinsippene bak protokollen tofasecommit (2PC).

Anta at vi har et distribuert system med tre noder  $K$ ,  $M$  og  $N$ , hvor  $M$  har en kopi (replikat) av elementet  $a$  og  $N$  en kopi av hvert av elementene  $b$ ,  $c$  og  $d$ .  $K$  har ingen kopier.  $K$  initierer transaksjonen  $T_1$  og  $N$  transaksjonene  $T_2$  og  $T_3$ . Transaksjonene  $T_2$  og  $T_3$  kan kjøres lokalt på node  $N$ . Transaksjon  $T_1$  må imidlertid utføres distribuert på de tre nodene.

- (ii) Anta at node  $K$  gjennomfører 2PC-protokollen på vegne av transaksjon  $T_1$  og at node  $M$  fullfører sin deltransaksjon av  $T_1$ , mens node  $N$  må abortere sin del av  $T_1$  fordi den kommer i konflikt med  $T_2$  og  $T_3$ . Beskriv hvordan den resulterende meldingsutvekslingen mellom  $K$ ,  $M$  og  $N$  forløper.
- (iii) Anta at nettverket mellom nodene  $K$  og  $M$  går ned under fase 2 i 2PC-protokollen slik at  $M$  aldri får noen fase 2-melding fra koordinatoren  $K$ . Senere kommer nettverket opp igjen. Hvordan kan  $M$  få avsluttet sin del av protokollen?