

Tentamen i DATABASER

Svar:

Obs! Lärare-version, med lösningar

DAG: må, 19 okt 1998 **TID:** kl. 8.45 – 12.45 **SAL:** ML2-ML8

Ansvarig: Martin Weichert
Förfrågningar: Martin Weichert, tel. 772 10 68
Resultat: anslås den 8 nov 1998
Poängantal: sammanlagt maximalt 60 poäng.
Betygsgränser: CTH: 3:a 24 p., 4:a 36 p., 5:a 48 p.
GU: Godkänd 28 p., Väl godkänd 48 p.
Doktorander: Godkänd 28 p.
Hjälpmedel: utdraget ur *Oracle7 Server SQL Language*
resp. *Appendix B Syntax* ur referensmanualen för *Oracle SQL*
samt *SQL*Plus Quick Reference*.

Observera:

- Skriv tydligt och disponera pappret på ett lämpligt sätt.
- Börja varje uppgift på nytt blad. Skriv endast på en sida av pappret.
- Alla svar skall väl motiveras!
- Ange på tentan om du går på GU eller CTH. För CTH ange även vilken linje du går!

Lycka till!

Uppgift 1. (a) Vilka är de fyra krav som ställs på ett databassystem med **transaktioner**?

Svar:

- A. "Atomicity" - "Allt eller inget": Varje transaktion ska genomföras antingen helt eller inte alls.
- C. "Consistency" - "Se upp för konsistensvillkoren": En transaktion som börjar i ett konsistent tillstånd måste leda till ett konsistent tillstånd.
- I. "Isolation" - "Ingen Insyn": Inga (eventuellt inkonsistenta) mellantillstånd av en transaktion ska vara synliga till en annan.
- D. "Durability" - "Det som är gjort är gjort" En transaktions ändringar som har avslutats och bekräftats med COMMIT, gäller.

4 poäng.

(b) Varför är relationsalgebra viktig för databaser?

Svar: Hela teorin om relationsdatabaser bygger på relationsalgebra. Uppdelning av relationer i delrelationer är en relationsalgebra-operation (projektion, π); förening av dessa till en större är en relationsalgebra-operation (samkörning, \bowtie); alla **select**-satser uttrycker relationsalgebra-operationer. Relationsalgebra är det matematiska fundament som relationsdatabaser bygger på.

2 poäng.

6 poäng.

Uppgift 2. Givet är tabellerna $r(\underline{A}, B, C)$, $s(\underline{B}, D, E)$, $t(\underline{C}, E, F)$ och $u(\underline{E}, B)$ med följande innehåll:

$r:$	<table border="1"><thead><tr><th><u>A</u></th><th>B</th><th>C</th></tr></thead><tbody><tr><td>a_1</td><td>b_1</td><td>c_3</td></tr><tr><td>a_2</td><td>b_1</td><td>c_2</td></tr><tr><td>a_3</td><td>b_2</td><td>c_3</td></tr></tbody></table>	<u>A</u>	B	C	a_1	b_1	c_3	a_2	b_1	c_2	a_3	b_2	c_3
<u>A</u>	B	C											
a_1	b_1	c_3											
a_2	b_1	c_2											
a_3	b_2	c_3											

$s:$	<table border="1"><thead><tr><th><u>B</u></th><th>D</th><th>E</th></tr></thead><tbody><tr><td>b_1</td><td>d_1</td><td>e_2</td></tr><tr><td>b_2</td><td>d_1</td><td>e_1</td></tr><tr><td>b_3</td><td>d_2</td><td>e_3</td></tr><tr><td>b_4</td><td>d_2</td><td>e_3</td></tr></tbody></table>	<u>B</u>	D	E	b_1	d_1	e_2	b_2	d_1	e_1	b_3	d_2	e_3	b_4	d_2	e_3
<u>B</u>	D	E														
b_1	d_1	e_2														
b_2	d_1	e_1														
b_3	d_2	e_3														
b_4	d_2	e_3														

$t:$	<table border="1"><thead><tr><th><u>C</u></th><th>E</th><th>F</th></tr></thead><tbody><tr><td>c_1</td><td>e_3</td><td>f_1</td></tr><tr><td>c_2</td><td>e_1</td><td>f_2</td></tr><tr><td>c_3</td><td>e_4</td><td>f_3</td></tr></tbody></table>	<u>C</u>	E	F	c_1	e_3	f_1	c_2	e_1	f_2	c_3	e_4	f_3
<u>C</u>	E	F											
c_1	e_3	f_1											
c_2	e_1	f_2											
c_3	e_4	f_3											

$u:$	<table border="1"><thead><tr><th><u>E</u></th><th>B</th></tr></thead><tbody><tr><td>e_1</td><td>b_1</td></tr><tr><td>e_2</td><td>b_2</td></tr><tr><td>e_3</td><td>b_1</td></tr><tr><td>e_4</td><td>b_3</td></tr></tbody></table>	<u>E</u>	B	e_1	b_1	e_2	b_2	e_3	b_1	e_4	b_3
<u>E</u>	B										
e_1	b_1										
e_2	b_2										
e_3	b_1										
e_4	b_3										

Beräkna resultatet på följande relationsalgebra-uttryck (beakta att resultatet är **mängder**):

- (a) $\pi_B(r) \cap \pi_B(u)$
- (b) $\sigma_{F \neq f_2}(t)$
- (c) $r \bowtie u$

Svar: $\pi_B(r) \cap \pi_B(u)$

<u>B</u>
b_1
b_2

 $\sigma_{F \neq f_2}(t):$

<u>C</u>	E	F
c_1	e_3	f_1
c_3	e_4	f_3

 $r \bowtie u:$

<u>A</u>	B	C	E
a_1	b_1	c_3	e_1
a_2	b_1	c_2	e_1
a_1	b_1	c_3	e_3
a_2	b_1	c_2	e_3
a_3	b_2	c_3	e_2

En poäng avdrag om det finns dubletter.

– (a) **2p.** – (b) **1p.** – (c) **2p.** Kolumner får vara i annan ordning.

5 poäng.

Uppgift 3. Givet är tabellerna $r(\underline{A}, B, C)$, $s(\underline{B}, D, E)$, $t(\underline{C}, E, F)$ och $u(\underline{E}, B)$ med innehåll som i föregående uppgift. På tabellerna är följande referenser (främmande nycklar) definierade:

- från $r.B$ till $s.B$ med villkor "ON DELETE SET NULL"

- från $r.C$ till $t.C$ med villkor "RESTRICTED"
- från $s.E$ till $u.E$ med villkor "ON DELETE CASCADE ON UPDATE CASCADE"
- från $t.E$ till $u.E$ med villkor "ON DELETE SET DEFAULT ON UPDATE SET NULL"

Vad händer när man försöker:

- (a) ta bort tuppeln $\langle e_1, b_1 \rangle$ ur u ?

Svar: Ur s försvinner tuppeln $\langle b_2, d_1, e_1 \rangle$ (CASCADE) – **1p.**

I r ändras därmed tuppeln $\langle a_3, b_2, c_3 \rangle$ till $\langle a_3, NULL, c_3 \rangle$ (SET NULL) – **1p.**

I t ändras tuppeln $\langle c_2, e_1, f_2 \rangle$ till $\langle c_2, e_{default}, f_2 \rangle$ (SET DEFAULT) – **1p.**

I r påverkas varken $\langle a_2, b_1, c_2 \rangle$ eller någon annan tuppel av ändringen i t – ingen poäng för rätt svar här, men avdrag för fel svar.

- (b) ändra tuppeln $\langle e_3, b_1 \rangle$ i u till $\langle e_5, b_3 \rangle$?

Svar: I s ändras tupplerna $\langle b_3, d_2, e_3 \rangle$ och $\langle b_4, d_2, e_3 \rangle$ till $\langle b_3, d_2, e_5 \rangle$ och $\langle b_4, d_2, e_5 \rangle$ (CASCADE) – **1p.** Obs! Det är e -värdena som påverkas – b -värdena ändras inte!

I t ändras tuppeln $\langle c_1, e_3, f_1 \rangle$ till $\langle c_1, NULL, f_1 \rangle$ (SET NULL) – **1p.**

5 poäng.

Uppgift 4. Givet är tabellen $s(\underline{B}, D, E)$ från uppgift 2. Vilka av de följande flervärdeberoenden uppfyller s , vilka inte? Motivera!

- (a) $B \twoheadrightarrow D \mid E$

- (b) $D \twoheadrightarrow B \mid E$

Svar:

- (a) $B \twoheadrightarrow D \mid E$: ja; ty uppdelning i BD och BE ger inga falska tuppler.

– **1p.**

- (b) $D \twoheadrightarrow B \mid E$: nej; ty uppdelning i DB och DE ger falska tuppler:

B	D	E
b_1	d_1	e_1
b_2	d_1	e_2

– **1p.**

3 poäng.

Uppgift 5. En biograf vill hålla reda på sina filmer och filmvisningar och använder en databas med följande tabeller:

- $film(\underline{Film_Nr}, \underline{Titel}, \underline{Kategori}, \underline{Land})$

En lista över alla filmer, med ett nummer $Film_Nr$ som unikt identifierar varje film, $Titel$, $Kategori$ ('action', 'komedi' eller dyl.) och $Land$ (landet där filmen producerades).

- $med(\underline{Film_Nr}, \underline{Skådespelare})$

Lista över skådespelare som medverkar i någon film. $Film_Nr$ är referens till $film$.

- $sal(\underline{Sal_Nr}, \underline{Storlek})$

En lista över biografens alla salar, med deras $Storlek$ (antal platser).

- *visning*(*Visn_Nr*, *Datum*, *Tid*, *Sal_Nr*, *Film_Nr*, *Pris*)

En lista över alla filmvisningar. Varje visning har ett unikt visningsnummer *Visn_Nr*, vilket alltså är en nyckel till relationen, men även kombinationen (*Datum*, *Tid*, *Sal_Nr*) är en nyckel. *Tid* är klockslaget då visningen börjar och anges som heltal. *Sal_Nr* är referens till *sal*, och *Film_Nr* är referens till *film*.

Först skall databasen skapas:

- (a) Ange en fullständig SQL-sats som skapar tabellen *visning* enligt beskrivningen ovan. De andra tabellerna är redan skapade. Du kan använda datatypen DATE för datum.

För resten av uppgiften antar vi att databasen är skapad och är fylld med data. Skriv SQL-satser för följande uppgifter:

- (b) Lista nummer och titel på alla filmer där Sean Connery medverkar.
- (c) Lista (utan dubletter) alla skådespelare som spelar med i samma film som någon skådespelare som spelar med i samma film som Sean Connery.
- (d) Lista (utan dubletter) titel på alla filmer som visas den 01-AUG-98 i den största salen.
- (e) Lista (utan dubletter) alla datum där samma film visas i två olika salar samma dag.
- (f) Sänk priset på alla eftermiddagsvisningar (dvs. före klockan 7) med 20%.

Svar:

- (a) Ange en fullständig SQL-sats som skapar tabellen *visning*.

```
create table VISNING
(
  VISN_NR  char(8),
  DATUM    date,
  TID      decimal(2),
  SAL_NR   char(8),
  FILM_NR  char(8),
  PRIS     decimal(5,2),
  primary key (VISN_NR),
  unique   (DATUM,TID,SAL_NR),
  foreign key (FILM_NR) references FILM,
  foreign key (SAL_NR)  references SAL
);
```

– 5p.

- (b) Lista nummer och titel på alla filmer där Sean Connery med.

```
select FILM_NR, TITEL
  from FILM
 where FILM_NR in (select FILM_NR
                  from MED
                  where SKÅDESPELARE = 'Sean Connery');
```

eller

```

select FILM_NR, TITEL
  from FILM F, MED M
 where F.FILM_NR = M.FILM_NR and SKÅDESPELARE = 'Sean Connery';

```

Obs – villkoret F.FILM_NR = M.FILM_NR får inte glömmas!– **2p.**

- (c) Lista alla skådespelare (utan dubletter) som spelar i samma film som någon som spelar i samma film som Sean Connery.

```

select distinct SKÅDESPELARE
  from MED
 where FILM_NR in
       (select FILM_NR
        from MED
        where SKÅDESPELARE in
              (select SKÅDESPELARE
               from MED
               where FILM_NR in
                     (select FILM_NR
                      from MED
                      where SKÅDESPELARE = 'Sean Connery'))));

```

eller

```

select distinct A.SKÅDESPELARE
  from MED A, MED B, MED C, MED D
 where A.FILM_NR=B.FILM_NR
       and B.SKÅDESPELARE=C.SKÅDESPELARE
       and C.FILM_NR=D.FILM_NR
       and D.SKÅDESPELARE = 'Sean Connery';

```

Tabellen MED måste användas fyra gånger!

Obs! att även Sean Connery själv, samt dessa som spelar i samma film som han, uppfyller det angivna villkoret och därmed ska vara med i resultatet. Om någon har valt ut välja bort dem ur svaret, då räknas det dock inte som fel, förutsatt att det har gjorts på ett korrekt sätt.

– **3p.**

- (d) Lista (utan dubletter) titel på alla filmer som visas den 01-AUG-98 i den största salen.

```

select distinct TITEL
  from VISNING V, FILM F
 where F.FILM_NR = V.FILM_NR
       and DATUM='01-AUG-98'
       and SAL_NR in (select SAL_NR
                     from SAL
                     where STORLEK = (select max(STORLEK)
                                       from SAL));

```

Obs! Tabellen SAL måste användas två gånger: en gång med gruppfunktion **max** för att beräkna vilken storlek som är den största, och en gång för att selektera den sal (eller de salar) som har denna största storlek.

– **3p.**

- (e) Lista alla datum där någon film visas i två olika salar.

```
select distinct DATUM
  from VISNING
  group by DATUM, FILM_NR
having count(distinct SAL_NR) > 1;
```

eller

```
select distinct DATUM
  from VISNING
  group by DATUM
having count(distinct SAL_NR) > count(distinct FILM_NR);
```

eller

```
select distinct V1.DATUM
  from VISNING V1, VISNING V2
 where V1.DATUM = V2.DATUM
       and V1.FILM_NR = V2.FILM_NR
       and V1.SAL_NR <> V2.SAL_NR
```

– 2p.

(f) Sänk priset på alla visningar före klockan 7 med 20%.

```
update VISNING
  set PRIS = 0.8 * PRIS
  where TID < 7;
```

– 2p.

17 poäng.

Uppgift 6. Givet relation $r(A, B, C, D, E)$ med funktionella beroenden

1. $A \rightarrow BC$
2. $CD \rightarrow E$
3. $B \rightarrow D$
4. $E \rightarrow A$

(a) Bestäm alla nycklar till relationen r .

Svar: $\{A\}$, $\{B, C\}$, $\{C, D\}$ och $\{E\}$. Icke $\{B\}$! – 3p.

(b) Uppfyller r Boyce-Codd-normalformen? Motivera!

Svar: Nej. Beroendet 3. $B \rightarrow D$ bryter mot BCNF, eftersom $\{B\}$ inte är någon (över)nyckel. – 1p.

(c) Uppfyller r tredje normalformen? Motivera!

Svar: Ja. Beroendet 3. $B \rightarrow D$ uppfyller 3NF, eftersom D är ett nyckelattribut (ingår i nyckeln $\{C, D\}$).

(De andra beroendena uppfyller 3NF eftersom de redan uppfyller det starkare BCNF.) – 1p.

- (d) Om vi delar upp relationen r för att uppfylla en bättre (d.v.s. strängare) normalform, vilka delrelationer får vi? Vad förlorar vi? Vilka felaktiga data är möjliga att mata in i tabellen? (Med "felaktiga data" avses data som uppfyller alla nyckel- och referensvillkor men ändå bryter mot något av de givna funktionella beroendena; inte data som är felaktiga på något annat sätt.)

Svar: Vi delar upp med beroendet 3. $B \rightarrow D$, och vi får relationer $r_1(\underline{A}, \underline{B}, \underline{C}, \underline{E})$. och $r_2(\underline{B}, \underline{D})$ Vi förlorar (bl.a.) beroendet 2. $CD \rightarrow E$. (Faktiskt även de härledda beroenden $CD \rightarrow AB$.)

Felaktiga data kan matas in som bryter mot (bl.a.) beroendet 2., t.ex.:

r_1 :

A	B	C	E
a_1	b_1	c	e_1
a_2	b_2	c	e_2

 med nycklar A, BC och E . r_2 :

B	D
b_1	d
b_2	d

 med nyckel B .

– 3p.

- (e) Om vi **inte** delar upp relationen r , vilka felaktiga data är då möjliga att mata in i tabellen?

Svar: Felaktiga data kan matas in som bryter mot beroendet 3. $B \rightarrow D$, t.ex.:

r_1 :

A	B	C	D	E
a_1	b	c_1	d_1	e_1
a_2	b	c_2	d_2	e_2

 med nycklar A, BC, CD och E .

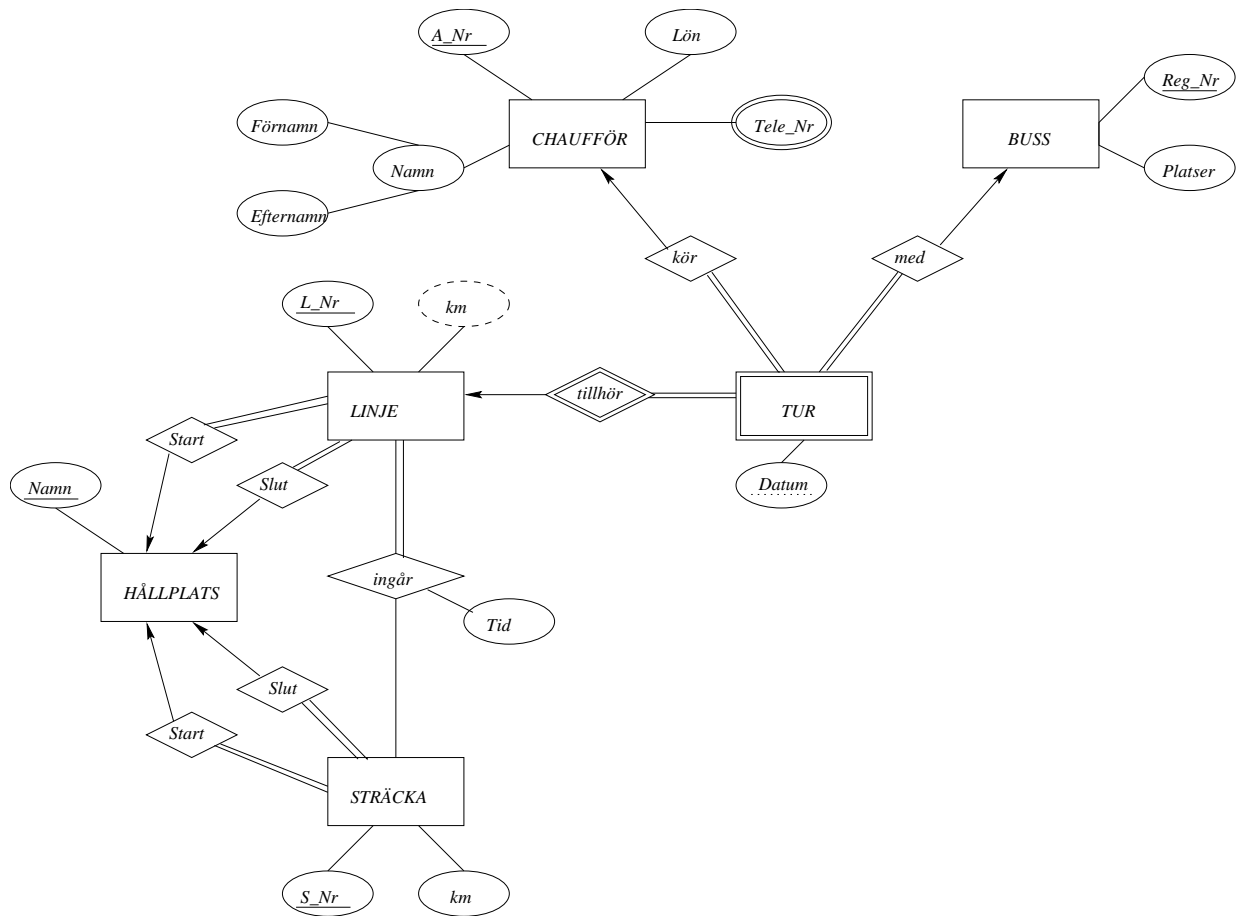
– 1p.

9 poäng.

Uppgift 7. Ett bussbolag vill införa en databas över sina bussar, chaufförer och linjer. Följande informationer om bolaget ska ingå i databasen:

- Bolaget har ett antal **chaufförer**. Varje chaufför har ett unikt anställd-nummer A_Nr , ett $Namn$, som består av $Förnamn$ och $Efternamn$, ett eller flera $Telefon_Nr$ och en $Lön$.
- Bolaget har också ett antal **bussar**, vilka har sitt registreringsnummer Reg_Nr (unikt) och ett antal sittplatser.
- Det finns ett antal **hållplatser**, vilka identifieras med sitt namn.
- Bolaget kör på flera **linjer**, vilka består av flera **sträckor** (minst en per linje). Varje sträcka har ett unikt $Sträcka_Nr$, en starthållplats och en sluthållplats, samt en längd i kilometer. Varje linje har ett unikt $Linje_Nr$, en starthållplats och en sluthållplats. Varje linje har också en längd i kilometer, vilken inte lagras direkt utan beräknas utifrån de ingående sträckorna.
En sträcka kan ingå i noll till flera linjer. För varje sträcka som ingår i en linje anges den planerade tiden när den sträckan ska köras i denna linje.
- Varje **tur** identifieras med numret av den linje den tillhör **och** det datum den genomförs. Varje tur körs av en viss chaufför i en viss buss.

Databasen beskrivs med följande *Entity-Relationship*-diagram:



(a) Översätt ER-diagrammet till relationsmodellen. (Du får ignorera det härledda attributet *Km* på *Linje* för denna deluppgift.)

Svar: Översättning i 7 steg:

- i. 5 vanliga entiter ger 5 relationer, med nycklar och attribut: *chaufför*, *buss*, *hållplats*, *linje*, *sträcka* – **4p.**
- ii. svag entitet ger 1 relation: *tur*, + 1 referens: *tur* → *linje* – **2p.**
- iii. inga 1:1-samband här
- iv. 6 stycken 1:N-samband ger 6 referenser – **2p.**
- v. 1 M:N-samband *ingår* ger 1 ny relation med 2 referenser – **2p.**
- vi. 1 flervärdesattribut ger en ny relation *telefon* med 1 ref – **2p.**
- vii. inga ternära eller högre samband här

Resultatet i relationsmodellen: 8 relationer, 10 referenser

telefon(*A_Nr*, *Tele_Nr*)

A_Nr → *chaufför*

chaufför(*A_Nr*, *Förenamn*, *Efternamn*, *Lön*)

buss(*Reg_Nr*, *Platser*)

tur(*L_Nr*, *Datum*, *Chaufför*, *Buss*)

L_Nr → *linje*

Chaufför → *chaufför*

Buss → *buss*

hållplats(*Namn*)

linje(L_Nr, Start, Slut)

Start → *hållplats*

Slut → *hållplats*

sträcka(S_Nr, Start, Slut, Km)

Start → *hållplats*

Slut → *hållplats*

ingår(L_Nr, S_Nr, Tid)

L_Nr → *linje*

S_Nr → *sträcka*

12p.

12 poäng.

- (b) Baserad på din lösning i (a), skapa en vy *linje_km(L_Nr, Km)* i SQL som listar alla linjer och deras längd i km, beräknad som summa över alla ingående delsträckor.

Svar:

```
create view LINJE_KM (L_NR, KM)
as
select I.L_NR, sum (S.KM)
  from STRÄCKA S, INGÅR I
  where S.S_NR = I.S_NR
group by I.LINJE;
```

3p.

3 poäng.