

Örebro universitet
Akademin för naturvetenskap och teknik
[Thomas Padron-McCarthy \(thomas.padron-mccarthy@oru.se\)](mailto:thomas.padron-mccarthy@oru.se)

Tentamen i Databasteknik II för D3 m fl

lördag 21 maj 2011

Gäller som tentamen för:
DT3001 Datateknik C, Databasteknik II, provkod 0100

Hjälpmedel:	Miniräknare.
Poängkrav:	Maximal poäng är 32. För betyget 3 krävs 16 poäng.
Resultat och lösningar:	Meddelas via e-post eller på kursens hemsida, http://basen.oru.se/kurser/db2/2010-2011-p34/ , senast lördag 11 juni 2011.
Återlämning av tentor:	Efter att resultatet meddelats kan tentorna hämtas i den centrala tentamensutlämningen i Studentcentrum.
Examinator och jourhavande:	Thomas Padron-McCarthy, telefon 070-73 47 013.

- Skriv tydligt och klart. Lösningar som inte går att läsa kan naturligtvis inte ge några poäng. Oklara och tvetydiga formuleringar kommer att misstolkas.
 - Skriv den personliga tentamenskoden på varje inlämnat blad. Skriv *inte* namn eller personnummer på bladen.
 - Skriv bara på en sida av papperet. Använd inte röd skrift.
 - Antaganden utöver de som står i uppgifterna måste anges. Gjorda antaganden får inte förändra den givna uppgiften.
 - Skriv gärna förklaringar om hur du tänkt! Även ett svar som är fel kan ge poäng, om det finns med en förklaring som visar att huvudtankarna var rätt.
-

LYCKA TILL!

Scenario till uppgifterna

Polisen registrerar alla hastighetsbegränsningar och fortkörningsböter i en databas:

Tabellen **Personer**
10 miljoner rader

<u>Pid</u>	Personnummer	Namn
1	621010-2219	Olle Olsson
2	170114-1440	Lotta Lottén
3	081213-1658	Kasper Svensson
...

Tabellen **Vägsträckor**
10 miljoner rader

<u>Sid</u>	Namn	Hastighetsbegränsning
1	Strutsvägen	50
2	Halkvägen 1-4	30
3	Halkvägen 5-21	50
4	E4 avsnitt 76	120
...

Tabellen **Böter**
1 miljon rader

<u>Bid</u>	Person	Vägsträcka	Datum	Uppmätt	Böter
1	3	2	2010-09-22	44	2000
2	1	2	2011-03-12	48	2000
3	3	4	2011-04-20	196	10000
...

Pid, **Sid** och **Bid** är primärnycklar. **Person** och **Vägsträcka** är referensattribut, och refererar till **Pid** respektive **Sid**.

Uppgift 1 (8p)

Vi vill veta vilka som kört mer än 10 kilometer för fort på Strutsvägen, och skriver därför följande SQL-fråga:

```
select Personer.Personnummer
from Personer, Vägsträckor, Böter
where Personer.Pid = Böter.Person
and Böter.Vägsträcka = Vägsträckor.Sid
and Vägsträckor.Namn = 'Strutsvägen'
and Böter.Uppmätt > 40;
```

- a) (1p) Ange den systematiska översättningen av SQL-frågan till relationsalgebra, även kallad kanonisk form. (Det är alltså den o-optimerade formen.) Skriv upp den som ett relationsalgebrauttryck.
- b) (1p) Rita upp samma relationsalgebrauttryck som ett frågeträd.
- c) (3p) Visa hur en heuristisk frågeoptimerare optimerar frågeträdet från deluppgiften ovan. Visa både vilka olika optimeringar som görs, och vad slutresultatet blir.
- d) (1p) Hur fungerar (kort) en kostnadsbaserad frågeoptimerare?
- e) (2p) En kostnadsbaserad frågeoptimerare kan ta hänsyn till en del saker som en heuristisk frågeoptimerare normalt inte bryr sig om. Vilka? Förklara, med exempel från databasen och frågan i exemplet!

Uppgift 2 (7 p)

Databasen lagras med en vanlig, diskbaserad relationsdatabashanterare på en vanlig, modern dator. Gör rimliga antaganden om blockstorlek mm (och ange dem!), och svara sedan på följande frågor. Förklara i samtliga fall hur du räknat.

a) Hur lång tid tar det i genomsnitt att köra följande fråga, om databasen saknar särskilda index:

```
select Datum
from Böter
where Bid = 786777
```

b) Hur lång tid tar det i genomsnitt att köra samma fråga, om databasen har ett primärindex i form av ett B⁺-träd på **Bid**?

c) Hur lång tid tar det i genomsnitt att köra samma fråga, om databasen har ett sekundärindex i form av ett B⁺-träd på **Bid**

d) Hur lång tid tar det i genomsnitt att köra följande fråga, om databasen saknar särskilda index:

```
select sum(Böter)
from Böter
```

e) Hur lång tid tar det i genomsnitt att köra samma fråga, om databasen har ett primärindex i form av ett B⁺-träd på **Bid**?

f) Hur lång tid tar det i genomsnitt att köra samma fråga, om databasen har ett sekundärindex i form av ett B⁺-träd på **Bid**

Uppgift 3 (3 p)

I arbetet med att öka trafiksäkerheten, ska vi försöka hitta samband och trender i materialet. Hur väl lämpar sig den här databasen för data mining? Borde den vara annorlunda, och i så fall hur?

Uppgift 4 (5 p)

Alla bötesbelopp ska höjas med tusen kronor, så vi kör följande SQL-fråga:

```
update Böter set Böter = Böter + 1000;
```

- a) Mitt i uppdateringen går strömmen! Förklara, med ändringar i databasen, noteringar i loggfilen och allt, varför det skulle kunna vara ett problem, och varför det inte är det.
- b) Mitt i återställningen ("recovery") går strömmen en gång till! Förklara, med ändringar i databasen, noteringar i loggfilen och allt, varför det skulle kunna vara ett problem, och varför det inte är det.

Uppgift 5 (4 p)

Förklara kort vad följande är:

- a) (1p) ODBC
- b) (1p) JDBC
- c) (1p) ESQLE
- d) (1p) ADO.NET

Uppgift 6 (5 p)

- a) Hastighetsbegränsningen på en viss väg ändras ibland. Därför vill vi lägga till tidsdimensionen **giltighetstid** i tabellen **Vägsträckor**. Visa hur det kan göras.
 - b) Visa med exempel vilka problem med referensintegritet som man kan få, när man lägger till den tidsdimensionen.
 - c) Det kan bli fel i databasen. Om en person slår upp en hastighetsbegränsning genom att titta i tabellen **Vägsträckor**, kan hon bli lurad, och ska då få lägre böter. Därför måste vi spara allt som tidigare stått i den tabellen, genom att lägga till tidsdimensionen **transaktionstid**. Visa hur det kan göras.
-