

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 23 maj 2009 kl 08:15 - 12:15

Gäller som tentamen för:
DT3001 Datateknik C, Databasteknik II, provkod 0100
TDD119 Databaser, fortsättningskurs, provkod 0100

Hjälpmedel:	Miniräknare.
Poängkrav:	Maximal poäng är 40. För betyget 3 krävs 20 poäng.
Resultat och lösningar:	Meddelas via e-post eller på kursens hemsida, http://basen.oru.se/kurser/db2/2008-2009-p3/ , senast lördag 13 juni 2009.
Återlämning av tentor:	Efter att resultatet meddelats kan tentorna hämtas på institutionen. Man kan också få sin rättade tenta hemskickad.
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

Rymdpatrullen har till uppgift att patrullera rymden. De åker runt i sina rymdskepp mellan de olika stjärnorna i galaxen. De har förstås en databas för att hålla reda på sina rymdskepp, och de stjärnor som de besöker.



Databasen består av två tabeller: en med stjärnor, och en med rymdskepp.

Tabellen **Stjärnor**
100 miljarder rader (10^{11})

<u>Sid</u>	x	y	z	Typ
1	35456249	6518015	797411	Röd jätte
2	46162024	77253767	59001	Vit dvärg
3	26939336	37832582	44488	Röd jätte
...

Tabellen **Rymdskepp**
1 miljon rader (10^6)

<u>Rid</u>	Namn	Position
1	Enterprise	2
2	Invincible	null
3	Doomhammer of Doom	2
...

Kolumnen **Position** i tabellen **Rymdskepp** anger vid vilken stjärna ett rymdskepp just nu befinner sig. (Rymdskeppet **Invincible** befinner sig just nu på resa mellan två stjärnor.)

Rymdpatrullen har inte råd med nya datorer, så databasen körs på en riktigt gammal maskin, från år 2009, med en databashanterare som inte uppdaterats sen dess.

Uppgift 1 (6 p)

En vanlig SQL-fråga som Rymdpatrullen ofta behöver köra, är att få reda på positionen där ett visst rymdskepp befinner sig. Exempelvis var rymdskepp nummer 3 finns just nu:

```
select z, y, z
from "Stjärnor", Rymdskepp
where Sid = Position
and Rid = 3;
```

Hur lång tid tar det, i genomsnitt, att köra en sån fråga?

Gör rimliga antaganden (och ange dem!) om blockstorlek, vilka index som finns, och så vidare. Du behöver beskriva hur frågan troligen kommer att köras, och varför SQL-frågan översätts och optimeras till just den exekveringsplanen. Visa hur du räknat!

Uppgift 2 (5 p)

Vad innebär giltighetstid och transaktionstid? Visa och förklara (a) vad det är, och (b) hur man kan göra för att få med dem i databasen, med rympatrullens databas som exempel.

Uppgift 3 (5 p)

Vi vill göra om rympatrullens databas till en distribuerad databas.

a) Visa hur rympatrullens databas kan fragmenteras med (a) primär horisontell fragmentering, (b) härledd horisontell fragmentering, och (c) vertikal fragmentering.

b) Visa hur SQL-frågan i uppgift 1 (översatt till relationsalgebra) lokaliseras i en fragmenterad databas, och hur deluttryck (eventuellt) kan reduceras. (Du behöver inte ha samma fragmenteringsschema som i a-frågan.)

Uppgift 4 (5 p)

I en databashanterare körs fyra olika transaktioner (**T1**, **T2**, **T3** och **T4**) enligt följande tidsschema. Transaktionerna arbetar med tre olika dataobjekt (**X**, **Y** och **Z**).

Tid	T1	T2	T3	T4
1	Start			
2		Start		
3		Läs(X)		
4		Läs(Y)		
5		Skriv(X)		
6		Commit		
7	Läs(X)			
8	Läs(Y)			
9	Skriv(X)			
10			Start	
11			Läs(Z)	
12			Skriv(Z)	
13				Start
14				Läs(X)
15				Läs(Y)
16			Commit	
17				Läs(Z)
18	Skriv(Y)			
19				Skriv(X)
20	Commit			
21				Commit

Precis före **Skriv**-operationen på rad 18 går plötsligt strömmen.

a) (2p) Hur ser loggfilen ut då?

b) (3p) Vilka operationer kommer att utföras under recovery-processen, när databasen återstartas? (Ignorera eventuella problem med isolering mellan transaktioner!)

Uppgift 5 (5 p)

a) (1p) Tidsschemat i uppgiften ovan är inte serialiserbart. Förklara varför!

b) (4p) Kursen tar upp flera olika metoder för isolering av transaktioner. Välj *två* av följande metoder, och visa hur tidsschemat ovan skulle förändras om respektive metod användes. (Observera: Välj alltså *två* av dessa metoder!)

- Tvåfaslåsning med enkla binära lås
- Tvåfaslåsning med läs- och skrivlås
- Tidsstämpelmetoden, antingen med eller utan Thomas skrivregel (Thomas's Write Rule)
- En optimistisk metod, till exempel som i Mimer

Uppgift 6 (7 p)

a) (3p) Vad är data mining? Vad är OLAP och OLTP? Vad är ett datalager (på engelska: data warehouse)? Vad har alla dessa saker med varandra att göra?

b) (2p) Vad är en datakub? Vad är ett stjärnschema? Vad har dessa saker med varandra att göra?

c) (2p) Vad är en dimensionstabell, och varför uppfyller den ofta bara andra normalformen?

Uppgift 7 (7 p)

Här är några olika tekniker för att kommunicera med en relationsdatabas från ett vanligt program:

- ODBC
- ESQL
- JDBC
- ADO.NET
- databasegna ("native") gränssnitt
- objekt-relationella översättningsramverk som till exempel Hibernate

I följande tre scenarion ska du bygga in databasåtkomst i ett program. Vilken eller vilka av dessa tekniker skulle du rekommendera? Varför? (Det är förmodligen motiveringarna som är den viktigaste delen i svaret.)

a) Ett gammalt system, som är skrivet i C++ och som kör på Unix-servrar. Belastningen på systemet är ganska låg.

b) Ett datainsamlingssystem som körs på en ganska vanlig Windows-dator, med anslutna sensorer, och som har mycket höga krav på att snabbt kunna samla in och lagra stora datamängder.

c) Ett nyutvecklat program som ska köras på klientdatorer av olika typer (Windows, Mac och Linux). Det kommer att ha en användare per installation.
